

IADIS MULTI CONFERENCE
ON COMPUTER SCIENCE
AND INFORMATION SYSTEMS

Rome, ITALY
20 -26 JULY



Proceedings of the
IADIS International Conference
Computer Graphics, Visualization,
Computer Vision and Image Processing 2011

EDITED BY
Yingcai Xiao

IADIS INTERNATIONAL CONFERENCE

**COMPUTER GRAPHICS,
VISUALIZATION, COMPUTER
VISION AND IMAGE
PROCESSING 2011**

part of the

**IADIS MULTI CONFERENCE ON COMPUTER SCIENCE AND
INFORMATION SYSTEMS 2011**

**PROCEEDINGS OF THE
IADIS INTERNATIONAL CONFERENCE
COMPUTER GRAPHICS,
VISUALIZATION, COMPUTER
VISION AND IMAGE
PROCESSING 2011**

part of the

**IADIS MULTI CONFERENCE ON COMPUTER SCIENCE AND
INFORMATION SYSTEMS 2011**

Rome, Italy

JULY 24 - 26, 2011

**Organised by
IADIS**

International Association for Development of the Information Society

Copyright 2011

IADIS Press

All rights reserved

This work is subject to copyright. All rights are reserved, whether the whole or part of the material is concerned, specifically the rights of translation, reprinting, re-use of illustrations, recitation, broadcasting, reproduction on microfilms or in any other way, and storage in data banks.

Permission for use must always be obtained from IADIS Press. Please contact secretariat@iadis.org

Informatics Volume Editors:

Yingcai Xiao

Computer Science and Information Systems Series Editors:

Piet Kommers and Pedro Isaías

Associate Editors: Luís Rodrigues

ISBN: 978-972-8939-48-9

TABLE OF CONTENTS

FOREWORD	xi
PROGRAM COMMITTEE	xiii
KEYNOTE LECTURES	xvii

FULL PAPERS

TONGUE SEGMENTATION FROM MRI IMAGES USING ITK-SNAP: PRELIMINARY EVALUATION <i>Paula Martins, Catarina Oliveira, Samuel Silva, Augusto Silva and António Teixeira</i>	3
SWITCHING PAIRWISE MARKOV CHAINS FOR NON STATIONARY TEXTURED IMAGES SEGMENTATION <i>Mohamed El Yazid Boudaren, Emmanuel Monfrini and Wojciech Pieczynski</i>	11
TEXTURE COMPRESSION IN MEMORY AND PERFORMANCE-CONSTRAINED EMBEDDED SYSTEMS <i>Jens Ogniewski, Andréas Karlsson and Ingemar Ragnemalm</i>	19
ACCURATE MULTI-MODAL IMAGE REGISTRATION USING COMPRESSION <i>Pere-Pau Vázquez and Jordi Marco</i>	27
A HEURISTIC APPROACH TO MULTI-FOCUS IMAGE FUSION <i>Po-Whei Huang, Cheng-I Chen and Phen-Lan Lin</i>	35
BRAIN MEDICAL IMAGE FUSION BASED ON IHS AND LOG-GABOR WITH SUITABLE DECOMPOSITION SCALE AND ORIENTATION FOR DIFFERENT REGIONS <i>Phen-Lan Lin, Po-whei Huang, Cheng-I Chen, Tsung-Ta Tsai and Chin-han Chan</i>	41
SUPER-RESOLUTION IMAGING: THE USE CASE OF OPTICAL ASTRONOMY <i>Thilo Bauer</i>	49
RECOGNITION OF FIRE/SMOKE BASED ON COMPUTER-VISION IMAGE SEGMENTATION <i>Marie Providence Umugwaneza, Jean Paul Dukuzumuremyi, Bei-Ji Zou, Lei Wang and Yixiong Liang</i>	60
AN IMAGE FEATURE DESCRIPTORS-BASED RECOVERY ACTIVATION METRIC FOR FLIR TARGET TRACKING <i>Gianluca Paravati, Andrea Sanna and Fabrizio Lamberti</i>	67

A FAST AND SPACE-EFFICIENT DISCRETE FOCAL STACK TRANSFORM <i>Fernando Pérez Nava and Sergio Fumero</i>	75
3D RECONSTRUCTION OF SMALL SOLAR SYSTEM BODIES USING PHOTOCLINOMETRY BY DEFORMATION <i>Claire Capanna, Laurent Jorda, Philippe Lamy and Gilles Gesquiere</i>	83
POINT-BASED SIMILARITY ESTIMATION BETWEEN 2.5D VISUAL HULLS AND 3D OBJECTS <i>Konstantinos Moustakas, Georgios Stavropoulos and Dimitrios Tzovaras</i>	91
COMPUTATION AND VISUALIZATION OF SURFACE OF PROTEINS AND THEIR CHANNELS <i>Barbora Kozlíková, Igor Aleksandrowicz and Jiří Sochor</i>	99
SURFACE FEATURE SHARPENING: OBSERVATION MINUS CALCULATION (O-C) APPROACH <i>V. Savchenko and M. Savchenko</i>	107
ADAPTIVE CHARACTER MOTION SYNTHESIS BY QUALITATIVE APPROACH <i>Fangde Liu, Xiaosong Yang and Jianjun Zhang</i>	115
ANIBOT - ANIMATION ROBOT FOR A LIFE <i>Masashi Furukawa, Michiko Watanabe, Ikuo Suzuki and Masahito Yamamoto</i>	123
SIMFOR: TOWARDS A COLLABORATIVE SOFTWARE PLATFORM FOR URBAN CRISIS MANAGEMENT <i>Jean-Christophe Chambelland, Romain Raffin, Brett Desbenoit and Gilles Gesquière</i>	133
A FRAMEWORK FOR IDENTIFYING USER SUITABILITY TO VISUALISATION APPROACHES <i>Christopher Moir and G. Stewart Von Itzstein</i>	141
MOVEMENT EPENTHESIS WITH PROSODY CONTROL FOR CHINESE SIGN LANGUAGE VIDEO SYNTHESIS <i>Ru Wang, Lichun Wang, Dehui Kong, Baocai Yin and Qingming Huang</i>	153
ZAPP – A MANAGEMENT FRAMEWORK FOR DISTRIBUTED VISUALIZATION SYSTEMS <i>Georg Tamm, Alexander Schiewe and Jens Krüger</i>	160
DATA VISUALISATION AND VALIDATION FOR HYDROLOGICAL MODELS <i>Karsten Rink, Thomas Fischer and Olaf Kolditz</i>	169
A STUDY ABOUT INFLUENCEABLE PARAMETERS IN THE CHOICE OF VISUALIZATION TECHNIQUES BASED ON GROUNDED THEORY <i>Juliana Keiko Yamaguchi and Maria Madalena Dias</i>	177

SHORT PAPERS

INDICATION OF METHANE GAS IN IR-IMAGERY <i>Julia Áhlén and Stefan Seipel</i>	187
---	-----

MICRO MACRO IMAGE SEGMENTATION <i>Asmar A. Khan, Costas Xydeas and Hassan Ahmed</i>	192
PHOTOMOSAIC USING QUAD-TREE STRUCTURE ON GPU <i>Jinsuk Yang, Choongjae Joo and Kyoungsu Oh</i>	197
AN ENLARGEMENT METHOD BASED ON UNDECIMATED WAVELET TRANSFORM AND ENHANCEMENT FUNCTION <i>Wen-Chien Yen, Ying-Wen Chang, Zhong-Xin Yu and Yi-In Lee</i>	202
SEGMENTATION AND DISCRIMINATION OF BREAST TUMORS IN ULTRASONIC IMAGES USING AN ENSEMBLE CLASSIFIER AND APPLICATION TO A DIAGNOSTIC SUPPORT SYSTEM <i>Atsushi Takemura</i>	207
INTEGRATION DATA HIDING WITH JP3D COMPRESSION ON VOLUMETRIC MEDICAL IMAGES <i>Marouen Toueti and Azza Ouled Zaid</i>	212
GAUSSIAN MIXTURE BACKGROUND MODEL WITH SHADOW INFORMATION <i>Jung-Ming Wang, Sei-Wang Chen and Chiou-Shann Fuh</i>	217
A SIMILARITY MEASURE BASED ON GEOMETRIC CONSISTENCY FOR IMAGE RETRIEVAL USING LOCAL DESCRIPTORS <i>A. Najjar and E. Zagrouba</i>	223
COMPUTING THE EUCLIDEAN DISTANCE TRANSFORM ON THE EPUMA PARALLEL HARDWARE <i>Ingemar Ragnemalm and Andréas Karlsson</i>	228
AUTOMATIC FUZZY SEGMENTATION OF A COLOR IMAGE – AN APPLICATION FOR ROBOCUP VISION SYSTEM <i>Juan M. Calderón, Andrea K. Pérez, Andres Gómez and J. A. López</i>	233
DEFORMATION EXCHANGE BETWEEN LARGE MESHES <i>Benjamin Duplex, Gilles Gesquiere, Marc Daniel and Fabien Perdu</i>	239
DEPTH VALUE DEDUCTION USING OPTICAL FLOW FOR REVERSE ENGINEERING <i>Rahmita Wirza Rahmat and Suhail Azmi</i>	245
BEZIER CLIPPING METHOD IMPROVEMENTS FOR DIRECT RENDERING OF BEZIER PATCHES <i>Sergey Fedorov, Mikhail Letavin and Maxim Shevtsov</i>	249
3D DOCUMENTATION OF NATURAL HERITAGE FOR VIRTUAL ENVIRONMENTS AND WEB — CASE STUDY: VALLEY OF GEYSERS, KAMCHATKA <i>Andrey Leonov, Alexander Aleynikov, Dmitriy Belosokhov, Alexander Bobkov, Evgeny Eremchenko, Pavel Frolov, Andrey Klimenko and Stanislav Klimenko</i>	255
MEASURE OF INDIVIDUAL EYE-HEAD COORDINATION COMPARED IN REAL AND VIRTUAL ENVIRONMENT <i>Gildas Marin, Marine Faure, Marc le Renard and Damien Paillé</i>	260
BUBBLING IN LIQUIDS <i>H. Sato and Y. Kawaguchi</i>	265

MEDICAL IMAGE 3D VISUALIZATION BY VECTOR BASED METHODS <i>Katrina Bolochko, Mihails Kovalovs and Aleksandrs Glazs</i>	271
NODE COLOR ASSIGNMENT FOR DECISION TREE STRUCTURE <i>Jeong Joong Lee, Seung Gyu Kim, Beom Seok Lee, Naveed Ejaz, Joon Yeon Choeh, Cheol Whan Lee, Myung Ho Lee and Sung Wook Baik</i>	276
HUMAN ACTION ANALYSIS VIA VIDEO STREAMS AND WORKFLOW SCHEDULING <i>Anastastios Doulamis, Nikolaos Matsatsinis and Pavlos Delias</i>	281
POLAR MOTION STATION-WEBCAM CALIBRATION FOR SOLAR TRACKING <i>Miguel Romero, Rafael Lemuz, Carlos Guillén, Cesar Bautista, Irene Ayaquica and Daniel Valdés</i>	287
UNSUPERVISED OPTIMAL DISCRIMINANT PLANE WITH UNCORRELATED FEATURES <i>Su-Qun Cao, Yun-Feng Bu, Xiao-Ming Zuo and Quan-Yin Zhu</i>	292
AN OFF-LINE PERSIAN HANDWRITTEN FORGERY DETECTION METHOD <i>Behzad Helli and Mohsen Ebrahimi Moghaddam</i>	297
PARTITION ITERATED FUNCTION RADIOGRAPHS IMAGE ENHANCEMENT FOR FISH BONE DETECTION <i>Noor Elaiza Abdul Khalid, Saadiyah Yahya and Hidayah Shariff</i>	303
EVALUATION OF TWO RECONSTRUCTION SYSTEMS BASED ON STEREOVISION <i>Amira Soudani and Ezzeddine Zagrouba</i>	308
EYE CENTRE LOCATION WITH HOUGH TRANSFORM <i>Ivan A. Matveev</i>	313
APPROXIMATION OF 3D DATA WITH PIECEWISE SERIES EXPANSIONS FOR SURFACE INSPECTION <i>Sebastian von Enzberg and Bernd Michaelis</i>	319

REFLECTION PAPERS

IMPLEMENTING COMPOSITING MANAGERS IN EMBEDDED SYSTEMS <i>Mirto Musci</i>	327
A COMPARISON OF OPTICAL FLOW TECHNIQUES FOR A BLIND SPOT CAMERA <i>Lorenz Van den Heuvel, Kristof Van Beeck and Toon Goedemé</i>	331

POSTERS

A FLEXIBLE FRAMEWORK FOR MULTI-VOLUME DATA VISUALIZATION <i>Nikolay Gavrilov, Alexandra Belokamenskaya and Vadim Turlapov</i>	337
EDGE-DIRECTED IMAGE UPSAMPLING METHOD BASED ON DIRECTIONAL KERNEL INTERPOLATION <i>Jungho Yoon and Yeon Ju Lee</i>	340
HOMOGRAPHY USING TOPOLOGICAL INFORMATION FROM SCATTERED FEATURE POINTS <i>Rishu Gupta and Lee Byung Gook</i>	343
ADEQUATE 3D VISUALIZATION OF REMOTELY SENSED ICE SHELF AND SEA SURFACE TEMPERATURE DATA <i>Guido Staub</i>	346
A VISUAL ANALYSIS TOOL FOR AMPLIFYING STORY GENERATION CYCLE <i>Mina Akaishi, Makoto Sato and Koichi Hori</i>	349

AUTHOR INDEX

FOREWORD

These proceedings contain the papers of the IADIS International Conference Computer Graphics, Visualization, Computer Vision and Image Processing 2011, which were organised by the International Association for Development of the Information Society Rome, Italy 24 – 26 July, 2011. This conference is part of the Multi Conference on Computer Science and Information Systems 2011, 20 - 26 July 2011, and which had a total of 1402 submissions.

The IADIS Computer Graphics, Visualization, Computer Vision and Image Processing (CGVVCVIP) 2011 conference aims to address the research issues in the closely related areas of Computer Graphics, Visualization, Computer Vision and Image Processing. The conference encourages the interdisciplinary research and applications of these areas.

This event received 98 submissions from more than 30 countries. Each submission has been anonymously reviewed by an average of four independent reviewers, to ensure that accepted submissions were of a high standard. Consequently only 21 full papers were published which corresponds to an acceptance rate below 16 %. A few more papers were accepted as short papers, reflection papers and posters. An extended version of the best papers will be published in the International Journal on Computer Science and Information Systems (ISSN: 1646-3692) and/or in the IADIS International Journal on WWW/Internet (ISSN: 1645-7641) and also in other selected journals, including journals from Inderscience.

Besides the presentation of full papers, short papers, reflection papers and posters, the conference also included two keynote presentations from internationally distinguished researchers. We would therefore like to express our gratitude to Professor Anthony Steed, Head of Virtual Environments and Computer Graphics, University College London, United Kingdom and Professor Patrick S. P. Wang, Zijiang Visiting Chair Professor, ECNU, Shanghai, NTUST, Taipei & Founding President and CEO, Wang Teknowledge Lab, for accepting our invitation as keynote speakers.

As we all know, organising these conferences requires the effort of many individuals. We would like to thank all members of the Program Committees, for their hard work in reviewing and selecting the papers that appear in the proceedings.

This volume has taken shape as a result of the contributions from a number of individuals. We are grateful to all authors who have submitted their papers to enrich the conferences' proceedings. We wish to thank all members of the organizing committee, delegates, invitees and guests whose contribution and involvement are crucial for the success of the conferences.

Last but not the least, we hope that everybody will have a good time in Rome, and we invite all participants for the next edition of CGVCVIP that will be held in Lisbon, Portugal.

Yingcai Xiao

The University of Akron

USA

Computer Graphics, Visualization, Computer Vision and Image Processing 2011

Conference Program Chair

Piet Kommers, University of Twente, The Netherlands

Pedro Isaías, Universidade Aberta (Portuguese Open University), Portugal

MCCSIS 2011 General Conference Co-Chairs

Rome, Italy

July 2011

PROGRAM COMMITTEE

COMPUTER GRAPHICS, VISUALIZATION, COMPUTER VISION AND IMAGE PROCESSING 2011 CONFERENCE PROGRAM CHAIR

Yingcai Xiao, The University of Akron, USA

MCCSIS GENERAL CONFERENCE CO-CHAIRS

Piet Kommers, University of Twente, The Netherlands
Pedro Isaías, Universidade Aberta (Portuguese Open University), Portugal

COMPUTER GRAPHICS, VISUALIZATION, COMPUTER VISION AND IMAGE PROCESSING 2011 COMMITTEE MEMBERS

Adrian Jarabo, University Of Zaragoza, Spain
Alberto Raposo, Puc-rio, Brazil
Alessandro Artusi, CaSToRC Cyprus Institute, Cyprus
Alexander Pasko, Bournemouth University, United Kingdom
Alexander Wiebel, Zuse Institute Berlin (ZIB), Germany
Alexei Sourin, Nanyang Technological University, Singapore
Anderson Maciel, UFRGS, Brazil
Andreas Gerndt, German Aerospace Center (DLR), Germany
Andreas Kerren, Linnaeus University, Sweden
Andrzej Lukaszewski, University Of Wroclaw, Poland
Angele Giuliano, Acrosslimits, Malta
Angelica De Antonio, Universidad Politecnica De Madrid, Spain
Anna Tomaszewska, Technical University Of Szczecin, Poland
Arcadio Reyes Lecuona, Universidad De Málaga, Spain
Arturo S. Garcia, University of Castilla-la Mancha, Spain
Beatriz Rey, Universidad Politecnica De Valencia, Spain
Belen Masia, Universidad de Zaragoza, Spain
Branka Medved Rogina, Institute Ruđer Bošković, Croatia
Brian D'auriol, Kyung Hee University, Korea, Republic Of
Bruce Campbell, Rhode Island School Of Design, USA
Carla Binucci, Università Degli Studi Di Perugia, Italy
Carlo Nati, Working Group for development of Scientific and Te, Italy
Carlos Gonzalez , University Of Castilla-la Mancha, Spain
Chang Ha Lee, Chung-ang University, Republic of Korea

Charalambos Poullis, University Of Cyprus, Cyprus
 Charalampos Georgiadis, The Aristotle University, Greece
 Chiara Catalano, CNR IMATI-ge, Italy
 Chih-Cheng Hung, Southern Polytechnic State University, USA
 Christos Gatzidis, Bournemouth University, United Kingdom
 Creto Vidal, Federal University Of Ceará, Brazil
 Dalton Lin, National Taipei University, Taiwan
 Daniel Acevedo Feliz, King Abdullah University Of Science And Technology, Saudi Arabia
 Daniel Steffen, German Research Center For Artificial Intelligence, Germany
 Daniel Thalmann, Nanyang Technological University, Singapore
 Dariusz Frejlichowski, West Pomeranian University Of Technology, Poland
 Demetri Terzopoulos, University of California, Los Angeles, USA
 Dimitri Plemenos, Université De Limoges, France
 Dongfeng Han, University Of Iowa, USA
 Emre Celebi, Louisiana State University In Shreveport, USA
 Eugene Zhang, Oregon State University, USA
 Ezendu Ariwa, London Metropolitan University, United Kingdom
 Fatima Nunes, University Of Art, Science And Humanities School, Brazil
 Flavio Prieto, Universidad Nacional de Colombia, Colombia
 Fotis Liarokapis, University Of Coventry, United Kingdom
 Francisco Gonzalez, University Of Girona, Spain
 Frank Michel, German Research Center For Artificial Intelligence, Germany
 Gaetano Impoco, Co.R.Fi.La.C., Italy
 Galina Pasko, British Institute of Technology & E-commerce, United Kingdom
 Georges Grinstein, University Of Massachusetts, USA
 Georgios Sakas, Fraunhofer Institute For Computer Graphics, Germany
 Gerd Reis, German Research Center for Artificial Intelligence, Germany
 Gerik Scheuermann, University of Leipzig, Germany
 Giovanni Farinella, University of Catania, Italy
 Giovanni Gallo, Università Di Catania,, Italy
 Giuseppe Patané, CNR-IMATI, Italy
 Gleb Nosovskiy, Moscow State University, Russia
 Gordon Erlebacher, Florida State University, USA
 Gustavo Patow, Universitat De Girona, Spain
 Heidrun Schumann, University of Rostock, Germany
 Hong Yi, University Of North Carolina, Usa
 Hongchuan Yu, Bournemouth University, United Kingdom
 Huub Van De Wetering, Technische Universiteit Eindhoven, Netherlands
 Igor Sevastianov, Mental Images Gmbh, Germany
 Ingemar Ragnemalm, Linköping University, Sweden
 Issac Rudomin, Itesm-cem, Mexico
 Jens Krüger, Interactive Visualization and Data Analysis Grou, Germany
 Jian Chang, Bournemouth University, United Kingdom
 Jian Zhang, Bournemouth University, United Kingdom
 Jinyuan Jia, Tongji University, China

Jorge López, Universidad De Zaragoza, Spain
 José Izkara, TECNALIA, Spain
 Jose Ignacio Echevarria, University Of Zaragoza, Spain
 José Pascual Molina Massó, Universidad de Castilla-la Mancha, Spain
 Josip Almasi, Vrspace, Croatia
 Juan Manuel Gonzalez Calleros, University Of Puebla, Mexico
 Jyoti Singhai, Maulana Azad National Institute Of Technology Bhop, India
 Kadi Bouatouch, Irisa, University Of Rennes 1, France
 Karolj Skala, Rudjer Boskovic Institute, Croatia
 Krzysztof Okarma, West Pomeranian University Of Technology, Poland
 Krzysztof Walczak, Poznan University of Economics, Poland
 Kurt Debattista, The University Of Warwick, United Kingdom
 Kyoungsu Oh, Soongsil University, Korea, Republic Of
 Laura Monroe, Los Alamos National Lab, USA
 Laura Papaleo, University Of Genova, Italy
 Lihua You, Bournemouth University, United Kingdom
 Linda Giannini, Teacher, scholar of several subjects about Educati, Italy
 Luciano Soares, Pontifical Catholic University Of Rio De Janeiro, Brazil
 Luis Molina-tanco, Universidad De Malaga, Spain
 Marc Daniel, Ecole Superieure D ingenieurs De Luminy, France
 Marcelo Guimaraes, IFSP/FACCAMP, Brazil
 Marco Attene, Cnr Imati-ge, Italy
 Marla Schweppe, Rochester Institute Of Technology, USA
 Marta Fairen, Universitat Politècnica De Catalunya, Spain
 Matteo Cantamesse, Catholic University Of Milan, Italy
 Michela Mortara, CNR-IMATI Ge, Italy
 Mu-Chun Su, National Central University, Taiwan
 Nadia Ambrosetti, Universita Degli Studi Di Milano, Italy
 Nathan Carr, Adobe Systems Inc., USA
 Nicoletta Adamo-villani, Purdue University, USA
 Pablo Figueroa, Universidad De Los Andes, Colombia
 Paolo Pingi, CNR - ISTI, Italy
 Patrick Gioia, Orange Labs, France
 Pavel Slavik, Czech Technical University In Prague, Czech Republic
 Pawel Forczmanski, West Pomeranian University Of Technology, Poland
 Pedro Cano, Universidad de Granada, Spain
 Pere-pau Vázquez, Upc, Spain
 Peter Dannenmann, RheinMain University of Applied Sciences, Germany
 Petr Felkel, Czech Technical University In Prague, Czech Republic
 Quan Wen, University Of Electronic Science And Technology Of, China
 Richard Kulpa, M2s University Of Rennes 2, France
 Robert Laramee, Swansea University, United Kingdom
 Roman Durikovic, Comenius University in Bratislava, Slovakia

Sabine Coquillart, Inria, France
Sae Hwang, University Of Illinois At Springfield, USA
Shaun Bangay, Deakin University, Australia
Silvester Czanner, The University Of Warwick, United Kingdom
Simon Richir, Arts Et Metiers ParisTech, France
Sougata Mukherjea, Ibm India Research Lab, India
Stella Sylaiou, Aristotle University Of Thessaloniki, Greece
Stratos Stylianidis, Aristotle University Of Thessaloniki, Greece
Tassos Mikropoulos, The University Of Ioannina, Greece
Theresa-marie Rhyne, Computer Graphics Consultant, USA
Thomas Wischgoll, Wright State University, USA
Valery Adzhiev, Bournemouth University, United Kingdom
Wichian Sittiprapaporn, Mahasarakham University, Thailand
Wojciech Wiza, Poznan University Of Economics, Poland
Xiaogang Jin, Zhejiang University, China
Xiaosong Yang, Bournemouth University, United Kingdom
Xuejun Hao, Columbia University, USA
Zhong-hui Duan, University Of Akron, USA
Zongyi Liu, Amazon.com, USA
Zoran Ivanovski, Faculty Of Electrical Engineering and Information , Macedonia

KEYNOTE LECTURES

THE IMPACT OF IMMERSION ON HUMAN-COMPUTER INTERACTION

Professor Anthony Steed
Head of Virtual Environments and Computer
Graphics, University College London, United Kingdom

ABSTRACT

In the 1990s, the vision of virtual reality was that in the next generation of human-computer interaction, users would see, hear and touch computer-generated sensory information. It was argued that key to virtual reality was that users were immersed into the virtual world, and that the virtual world surround them. The vision required the use of head-mounted displays or CAVE systems, surround audio and a range of user sensing systems.

Needless to say, this vision didn't radically change most users' experience of HCI: the technology was too expensive, too cumbersome and just not easy to access. The use of virtual reality has been confined to a small number of niche high-value applications.

Having said that, over the past couple of years, all the component parts for a virtual reality system have become available in consumer form: position tracking technologies, fast stereo projectors, multi-channel video systems, etc. They are being assembled in various configurations but in variants of the virtual reality paradigm where computer generated stimuli are mixed into the real world setting. Such systems are often known as mixed-reality systems.

In this talk I will present an argument for how understanding the impact of immersion can inform the design and critique of these emerging mixed-reality systems. Key will be the way in which we can use proprioceptive information in HCI. I will illustrate this argument with a series of virtual reality and mixed-reality systems that have attempted to support simple face to face communication tasks.

INTELLIGENT PATTERN RECOGNITION AND APPLICATIONS --- MODELING AND SIMULATION IN INTERACTIVE E-LEARNING

Prof. Patrick S.P. Wang, Ph.D.
Fellow, IAPR , ISIBM and WASE Northeastern University Boston, ECNU
Shanghai, NTUST, Taipei

ABSTRACT

This talk deals with fundamental aspects of Intelligent Pattern Recognition (IPR) and applications. It basically includes the following: Overview of 3D Biometric Technology and Applications, Importance of Security: A Scenario of Terrorists Attack,, What are Biometric Technologies? Biometrics: Analysis vs Synthesis, Analysis: Interactive Pattern Recognition Concept, Importance of Measurement, How it works: Fingerprint Extraction and Matching, Iris, and Facial Analysis, Authentication Applications, Thermal Imaging: Emotion Recognition. Synthesis in Biometrics, Modeling and Simulation, and more Examples and Applications of 3D Biomedical Imaging in Interactive Web/Video Networking Fuzzy e-Learning Environment. Finally, some future research directions are discussed.

Full Papers

TONGUE SEGMENTATION FROM MRI IMAGES USING ITK-SNAP: PRELIMINARY EVALUATION

Paula Martins¹, Catarina Oliveira¹, Samuel Silva², Augusto Silva² and António Teixeira²

¹*Health School/IEETA, University of Aveiro*

Campus Universitário de Santiago 3810-193 Aveiro, Portugal

²*Dep. of Electronics, Telecommunications and Informatics/IEETA, University of Aveiro*

Campus Universitário de Santiago 3810-193 Aveiro, Portugal

ABSTRACT

The purpose of this study was to evaluate and compare the efficiency, reliability and accuracy of manual and semi-automatic segmentation techniques to segment tongue images. This work is included in a vast framework (HERON II) that aims to improve an articulatory-based speech synthesizer (SAP-Windows), for European Portuguese (EP). Volumetric data from Magnetic resonance images (MRI) were used to extract tongue configurations from several speakers uttering different EP sounds, or the same sound produced in different contexts or syllabic positions, in a speech production study. Segmentations were performed manually and using a semi-automatic approach implemented in ITK-SNAP (Region Competition Snakes). Results from similarity metrics (Jaccard coefficient and voxelwise comparison) revealed that the semi-automatic (SA) method presents good agreement with the manual segmentation method (Jaccard=0.9002 and 10.495 voxel error). Furthermore, the semi-automatic method is more reproducible (Jaccard=0.9382 and 6.388 voxel error) than manual segmentation method (Jaccard= 0.9170 and 8.662 voxel error). The semi-automatic approach provides an efficient and reliable method to segment tongue images providing 3D visualizations that allows description and comparison of tongue configurations during the production of different sounds. This information is of great relevance in speech production field contributing to a better understanding of speech production mechanisms.

KEYWORDS

Speech Production, 3D MRI, Tongue segmentation, European Portuguese, Validation

1. INTRODUCTION

In the speech synthesis field, articulatory synthesis, e.g., (Teixeira et al. 2005) is one of the most promising techniques but this type of anthropomorphic synthesizers demands large amounts of detailed anatomic-physiological information (e.g., tongue, vocal tract, velum), if possible in 3D (Birkholz and Kröger 2006). The tongue is a complex muscular structure and one of the most important articulators involved in the production of the different sounds. During speech production the tongue may assume very different tongue postures and shapes but as an incompressible structure maintains its volume constant (Stone and Lundberg 1996; Takemoto 2001; Perrier et al. 2003; Badin and Serrurier 2006). Although 2D data (in the midsagittal plane) could provide important and easy to obtain information regarding the tongue and vocal tract configurations, 3D data allow a more realistic and detailed description of the various speech organs involved in speech production. This information is particularly relevant when the goal is modeling lateral or nasal sounds (Badin and Serrurier 2006). However, to obtain accurate and reliable 3D articulatory information from the different vocal tract articulators involved in speech production is quite challenging (Narayanan et al. 1997).

From the several imaging techniques available, MRI is one of the most promising and used techniques in the field of speech sciences to obtain articulatory data. The advantages and limitations of the method in the field are well known and reported in the literature (e.g., (Narayanan et al. 1997; Engwall and Badin 1999)). Briefly, MRI main advantages are related with its multiplanar capacity; the possibility to allow 3D modeling of the different articulators, its excellent soft tissue contrast and no use of ionizing radiation being considered a safe imaging technique. The most frequent approach, in speech production studies, to gathered 3D data is to perform a stack of 2D images, acquired in different planes (coronal, axial, and sagittal) with slice thickness

varying from 3 to 4 mm (Narayanan et al. 1997; Badin et al. 1998; Engwall and Badin 1999). However, due to several technological improvements in the field of MR it is now possible, to use 3D sequences allowing to obtain a volume from all the vocal tract in few seconds (e.g., (Martins et al. 2008; Kim et al. 2009)). Several studies in the field still use manual segmentation to extract vocal tract and tongue 3D configurations (Perrier et al. 2003; Badin and Serrurier 2006; Ventura et al. 2008). Our broad interests are related with the study of EP sounds (particularly lateral sounds) and to evaluate lingual coarticulatory effects (e.g., the influence of one sound into another). In this scenario, there is a need to quantify the extent of this influence. Because the influence of a vowel in the production of e.g., an /l/ sound could be very subtle, the segmentations and meshes obtained, in which we will base our assumptions, should be accurate. Moreover, because we are dealing with an extensive database with more than 20 tongue and vocal tract configurations, *per* speaker, we aim to have an efficient but reliable segmentation technique that allows a correct articulatory description of the different EP sounds. To the best of our knowledge validation studies in this area are scarce. The main goal of the present paper is to assess, through the use of different metrics, if a more time-efficient (semi-automatic) method to segment tongue images provides valuable results as compared to a more thorough (manual) time consuming method.

Following a brief overview on the topic, section 2 describes image acquisition, segmentation techniques used and validation procedures/metrics considered to assess the quality of the segmentations provided semi-automatically and manually; the results obtained are presented in section 3. Finally, a discussion of the obtained results is presented along with the main conclusions that can be drawn from this preliminary evaluation and future ideas for future work.

2. METHODS

2.1 Image Acquisition

Volumetric MRI data were acquired with a 3T MRI scanner (Magnetom Trio) with high performance gradients ($G_{max} = 45\text{mT/m}$, rise time= 0.2s, slew Rate= 200 T/m/s; and FOV =50 cm). A standard 12-channel Head and Neck phased-array coils and parallel imaging (GRAPPA) were used in all data acquisition sessions. The Imaging protocol used in the present study was based in a previous MRI study (Martins et al. 2008). The subjects were positioned comfortably in a supine position using headphones. A volume covering the entire vocal tract was obtained in the sagittal plane with a T1W 3D Spoiled GE sequence (VIBE), resulting in an acquisition time of 19 seconds; matrix (224x256); voxel size (1.055, 1.055, 2). MRI data was acquired from seven EP speakers: three female and four male, ages ranging from 21 to 39, with no history of hearing or speech disorders. An MRI screening form and informed consent were obtained before their participation in the study. European Portuguese cardinal vowels [a, i, u], lateral (e.g., l as in *sal*, salt and L as in *palha*, straw) and rhotic (e.g /r/ as in *caro*, expansive) sounds were included in the corpus. The lateral sound /l/ was acquired in the context of the three EP cardinal vowels and in different positions in the word (e.g., word initially, intervocalic and word final), conducting to a large MRI database.

2.2 Image Segmentation

Segmentation and its accuracy in representing an object are fundamental to obtain precise qualitative or quantitative measures, and a crucial step in a pipeline to obtain biomechanical models (Ribeiro et al. 2009). Although our immediate goals do not include the construction of a biomechanical model of the tongue, we need accurate data that allows the description and comparison of tongue posture, shape, and features, such as, groove or tongue dorsum configuration in the production of different sounds.

Although manual segmentation is frequently considered the gold standard, this process is very time consuming, depends on the operator prior knowledge regarding the structures to be segmented, and reliability is a vulnerable issue (Pham et al.,2000). Because we are in the presence of a rather extensive database (with seven speakers and several contexts), it is desirable to use, whenever possible, semi-automatic tools as they represent a good trade-off between efficiency and accuracy (Pham et al. 2000). In this paper our focus is on tongue segmentation but vocal tract segmentation (to obtain vocal tract 3D visualizations, length, lateral channels and area functions) has also been performed. For the vocal tract other software application has been

used (MevisLab). To obtain 3D qualitative and quantitative data regarding the tongue, we used (ITK-SNAP 2010). This software application is based on active contours and it was developed with the main goal of segmenting anatomical structures, in clinical settings, as a reliable alternative to manual segmentation (Yushkevich et al. 2006). ITK-SNAP has mainly been used in the field of neuroimaging to perform segmentations of several intracranial structures from high resolution CT or MRI images (e.g., caudate nucleus, lateral ventricles, hippocampus, brain tumors) with excellent reliability and efficiency (Ho et al. 2002; Ho et al. 2003; Yushkevich et al. 2006). Other examples include segmentation of bone structures for geometrical modeling e.g. (Ribeiro et al. 2009). ITK-SNAP implements two different automatic methods of segmentation, both based on active contours: Geodesic Active Contours and Region Competition Snakes. Manual editing is also possible allowing fully manual segmentations or post processing corrections after semi-automatic segmentations (Ho et al. 2003; Yushkevich et al. 2006).

2.2.1 Manual Segmentation

Manual segmentation of the tongue is performed using ITK SNAP manual editing tool. SNAP display options, allow simultaneous visualization of the three orthogonal planes. This is very important and helpful to better delineate the structure of interest. The segmentation starts with delineation of landmarks in the axial and coronal images (see figure 1) facilitating the identification of the tongue margins, in the most lateral and inferior sides of the tongue, where the contrast between the tongue and surrounding soft tissues is poor. After that segmentation proceeds in the sagittal plane, slice by slice through all the tongue. Finally, a careful inspection and correction of the segmentation must be performed. The process takes 30 minutes (Mean), ranging from 22 to 41 minutes depending on the tongue dimension, which varies among speakers.

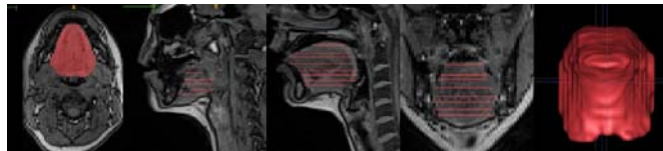


Figure 1. Reference tracings, in orthogonal planes, to guide manual segmentation. Sample of a tongue mesh (at right).

2.2.2 Semi-automatic Segmentation (SNAP-assisted)

Semi-automatic (SA) segmentation of the tongue is performed using the implementation of ITK-SNAP based on Region Competition Snakes. In this case the contour evolution is commanded by intensity differences between the object (the tongue) and the background. As shown in figure 2, the steps to segment the tongue images, using SNAP semi-automatic tool, includes: 1) selecting the region of interest (tongue), 2) image resampling to obtain an isotropic voxel, 3) defining a threshold, 4) placing seeds or bubbles inside the tongue, choosing the parameters that control contour evolution and finally running the algorithm. The first 4 steps take about one minute. However, some time was spent previously in the optimization of all the parameters (used, thereafter, for all segmentations). The computational time is very small, the snake converges in about 180 iterations (few seconds). After that, manual correction can be performed to obtain a more accurate segmentation (particularly at the lateral and inferior sides of the tongue), which can takes more 4 to 5 minutes (revise the segmentation provided semi-automatically to remove or add unwanted or missing voxels). Total time of the SA's segmentations ranges between 6 to 7 minutes. A mesh of the tongue was obtained from the segmented volume allowing a qualitative 3D evaluation (e.g., position, overall tongue configuration, presence and location of grooving).

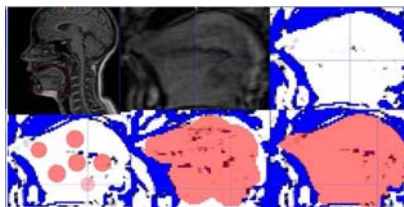


Figure 2. Segmentation steps using ITK SNAP semi-automatic segmentation based on Intensity Regions. From left to right, top: definition of the region of interest, resampled region, and thresholded image. From left to right, bottom: 3D bubbles placed inside of the tongue, evolution after 121 iterations, and after 182 iterations when the evolution was finished.

In this preliminary evaluation of tongue segmentation performed with ITK-SNAP, several productions from three of the speakers were used to assess efficiency, intra-operator reliability and accuracy. At this time, all the images were segmented by the same operator (PM), one of the authors of the paper. The operator has a large experience with imaging techniques and anatomical background due to its academic formation (radiology technician) and has also some knowledge at the level of Speech Sciences. All the procedures described were performed in a computer running Windows 7 with 6 GB RAM, 1.6 GHz Intel Core Duo processor and NVIDIA GeForce GPU. Environmental conditions were kept constant (e.g ambient light, display size). For validation purposes, to assess the reliability with both methods, the segmentations performed for the same sound were carried out in different days.

2.2.3 Validation Tools and Metrics

Validation of the segmentations obtained from a particular structure is of special relevance. According to Liu et al., 2003 when assessing the utility of any image segmentation method one should considered precision, accuracy and efficiency. Accuracy is most difficult to evaluate when a real “ground truth” does not exist. Frequently in these situations (e.g. clinical settings), manual tracings performed by an expert are considered the surrogate to which other methods of segmentation are compared (Liu et al. 2003). However, this approach is not fully consensual as manual tracings are vulnerable to reliability issues (Pham et al. 2000).

To evaluate reliability, the total volume of the structure (based on the number of voxels) is one of the metrics that can be used, being the simplest to compute. However, total volume based measures does not account for regional differences and usually conducts to high levels of agreement that are not necessarily correct (Gerig et al., 2001). In the literature there are several similarity metrics that allow comparison of areas or volumes provided by different raters, the same rater at different times or different segmentations methods accounting for differences in spatial properties (Silva et al. 2011).

In our study, after a preliminary qualitative evaluation, we used total volume of the segmentations as the first approach to assess reliability of both methods: fully manual segmentations (M) and semi-automatic segmentations (SA) followed by manual adjustments. Following (Gerig et al. 2001; Yushkevich et al. 2006) we computed Intraclass Correlation Coefficient to obtain SA vs SA, SA vs M and M vs M reliability measures. In the absence of a gold standard manual segmentations will be considered the reference method.

From the different possible metrics we choose the Jaccard coefficient and Voxel-wise difference. Figure 3 presents notable regions of interest when analyzing similarity between two regions and includes the Jaccard and Voxel wise formulation. Jaccard coefficient is equal to 1 if two regions overlap completely and it is zero if the regions do not overlap; Voxel-wise difference computes the ratio between the amount of false positive (FP) and false negative (FN) voxels and the mean segmented volume (considering the segmentations being compared). Statistical analysis presented in this work was performed with SPSS 15.0 (SPSS Inc., Chicago, IL, USA).

X is the region compared with Y (considered the reference).

True positives (TP)	a	$ X \cap Y $
False positives (FP)	$b1 + b2$	$ X - X \cap Y $
False negatives (FN)	$c1 + c2$	$ Y - X \cap Y $
True negatives (TN)	d	$ X \cup Y $

$$Jac = \frac{|X \cap Y|}{|X \cup Y|} \quad V = \frac{FP + FN}{MeanVolume} = \frac{|X| - |X \cap Y| + |Y| - |X \cap Y|}{(|X| + |Y|)/2}$$

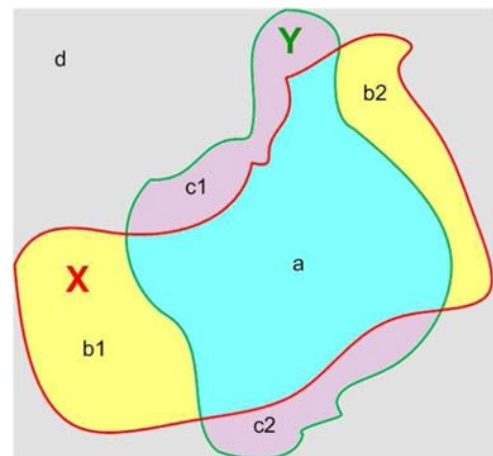


Figure 3. Notable regions of interest when analyzing the similarity between two regions X and Y. Jaccard coefficient and Voxel wise formulations are included. Adapted from Silva et al.,2011.

3. RESULTS

In this section, we present the main results obtained from both qualitative and quantitative analysis when comparing manual and semi-automatic segmentations of the tongue.

3.1 Qualitative Evaluation

The first approach to evaluate the segmentations was a qualitative visual evaluation of the overall quality of the segmentations, provided by both methods (manual and semi-automatic). The meshes obtained with ITK SNAP, for three manual segmentations and five semi-automated segmentations (2 different sounds) were randomized and presented (frontal, lateral, and oblique views of the mesh) to three raters (AS, CO, LJ). All of the raters have some phonetic and linguistic knowledge and were blinded to the segmentation process used to obtain the meshes. The raters are asked to observe all the meshes presented and evaluate if there are differences between meshes of the same sound that can preclude a correct articulatory description of the tongue (e.g tongue tip/ dorsum position/ presence and extension of midline groove, tongue dorsum configuration-convex/concave and lateral compression towards midline). Moreover, because smoothing process can introduce errors, meshes for the same sound with different degrees of smoothing were presented to the raters.

None of the three raters found relevant differences between meshes produced by both methods. When asked to identify the “best” segmentation all of them considered it to be a difficult choice. Two of the raters (CO, AS) chose 2 of the semi-automatic segmentations and the other chose a manual tracing (LJ) as the best segmentation. With the different degrees of smoothing, all the raters considered that a high degree of smoothing resulted in the loss of important features, such as grooving. Though meshes with less degree of smoothing presented some typical stair artifacts.

Furthermore, complementarily to this evaluation a mesh comparison tool: PolyMeco (Silva et al. 2009), was used to compare the different instances of the segmentations performed (i.e. same speaker and sound), providing important qualitative data (e.g., it is possible to highlight where the main differences occur and the amount of those differences based on a color representation). This is relevant because we can find if the main differences occur in regions of the tongue that are directly involved in the articulation of the different sounds. For the performed comparisons we observe that most differences occur in the most inferior-lateral areas of the tongue, where the contrast between tongue and surrounding soft tissues is poor. These areas are not involved in the articulation.

3.2 Quantitative Evaluation

3.2.1 Volume Analysis

Before comparing and assessing intra rater reliability based on total volume with both methods, we performed segmentations from tongue images acquired when producing different sounds, or the same sound in different contexts (e.g /l/ as in *litro*, *liter* or /l/ as in *sal*, *salt*), using SA technique for all the segmentations. Based on the knowledge that the tongue is an incompressible structure that can assume different postures and shapes but maintaining its volume constant (see Introduction for references) we may assume that, if the method is reliable, similar volumes will be obtained for all the segmentations. In fact, the volumes obtained from the 14 segmentations for speaker CO present a mean value of 79210 mm³ (min=77013mm³, max=81879 mm³ and SD =1688 mm³). These results corroborate the theory of tongue volume conservation and, though not representing a robust metric, the small standard deviation (SD) obtained can be interpreted as a good indicator of the level of the reliability of the semi-automatic algorithm in segmenting the tongue.

To assess intra-operator reliability with both methods, seven volumes (cases) consisting of different tongue configurations (see details on speakers and sounds on Table 1) were segmented 2 times manually, and 3 times semi-automatically, in a total of 35 volumetric segmentations. Table 1, presents the volumes obtained for each of the semi-automatic (SA1, SA2, SA3) and manual (M1, M2) segmentations obtained for each sound, mean and SD for both methods. As can be observed tongue volumes varies with speaker; the female speaker presenting the lowest volumes, not surprisingly, due to differences in height and weight. The values of SD obtained for the different segmentations for the same sound are lower than the SD obtained

when the volume was estimated from 14 different tongue shapes. Perhaps, the shape is a factor that introduces more variability than the variability introduced by the observer when segmenting the structure. Notice that the volumes obtained from manual segmentations, for the same sound and speaker, are very close to those obtained with the semi-automated method with a trend to being slightly smaller than the latter.

Table 1. Tongue volumes (in mm³), Mean and SD values obtained from semi-automatic (SA) and manual segmentations (M).

Speaker	Sound	Sa1	Sa2	Sa3	Mean (Sa)	SD	Man1	Man2	Mean	SD
co	sal	79836	80245	80110	80064	208	76816	77418	77117	426
co	sul	81567	82862	79947	81459	1461	80026	77988	79007	1441
co	vowel_a	80143	80003	80161	80102	86	80428	79550	79989	621
jpm	laca	140966	143417	14219	142177	1226	137366	139271	138319	1347
jpm	vowel_a	143346	143077	14162	142682	928	132852	133775	133314	653
lcr	laca	130006	129372	124700	128026	2898	121866	119931	120899	1366
lcr	palha	126972	125490	126583	126348	768	121025	120950	120988	53

The values obtained for Intraclass Correlation Coefficient (ICC) for each class are: SA vs SA (0.99), M vs M was (0.99) and 0.98 when considering SA vs M. These values pointed to an equivalent high reliability of both methods. The lower reliability values obtained when comparing SA and M is probably due to the differences in the volume reported above.

3.2.2 Overlapping Metrics

Although volumetric analyses could provide valuable information concerning the reliability of the method, it is also true that one can have exactly the same volume but a poor spatial overlap. To further evaluate how well two segmentations agree, other metrics were used. Comparisons include all M vs M, SA vs M and SA vs SA comparison pairs in a total of 70 comparisons. Table 2 presents minimum, maximum, mean and standard deviation values obtained with the Jaccard coefficient and voxel wise difference for all the comparison pairs. Figure 4 shows box plots (minimum, median, 25% and 75% quartiles and maximum) of overlaps statistics for all the comparison pairs.

Table 2. Minimum, Maximum, Mean and SD of the two validation metrics from all the segmentation pairs. Jaccard, range: [0, 1] and Voxelwise is presented in percentage (%).

Metric	Pairs	min	max	Mean	SD
Jaccard	MvsM	0.9066	0.9281	0.9170	0.0075
Voxwi	MvsM	7.4546	9.7986	8.6624	0.8101
Jaccard	MvsSA	0.8822	0.9162	0.9002	0.0079
Voxwi	MvsSA	8.7477	12.5098	10.4945	0.8730
Jac	SAvsSA	0.9150	0.9576	0.9382	0.0123
Voxwi	SAvsSA	4.3371	8.8769	6.3876	1.31113

A one-way analysis of variance (ANOVA) was carried out, with the method of segmentation as a factor: type of segmentation (SA, Manual). The ANOVA assumptions of residual normality (Kolmogorov-Smirnov's test) and homogeneity of variance (Levene's test) were verified. Table 3 summarizes ANOVA results.

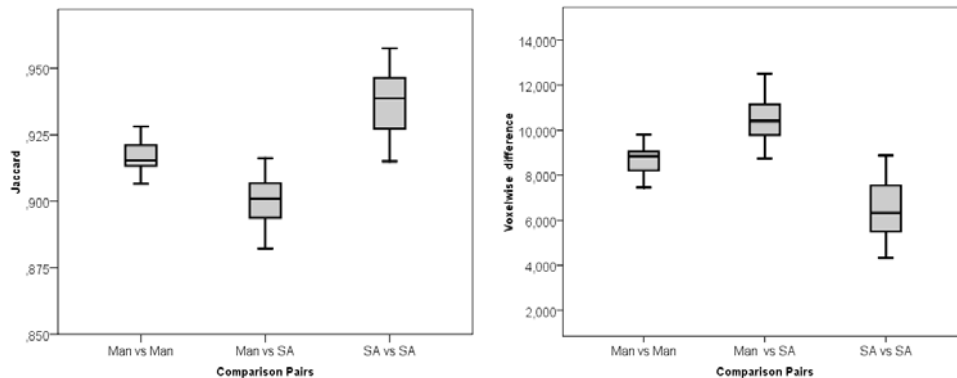


Figure 4. Box plots (minimum, median, 25% and 75% quartiles and maximum values) of overlaps statistics for all the comparison pairs. At left, Jaccard coefficient [0 1]; at right voxel wise difference (%).

Table 3. ANOVA results considering the segmentation method as a factor, Jaccard and voxel wise metrics.

Metric	Mean Sq	F	p-value
Jaccard	.002	18.232	< .001
Voxel wise	27.166	18.433	< .001

4. DISCUSSION AND CONCLUSIONS

The semi-automatic method (SNAP based on Region Competition Snakes) was much more efficient than manual segmentation, significantly decrease the time (from 30 to 6 minutes) and segmentation efforts, which is relevant in the presence of an extensive database. A qualitative, visual evaluation of the meshes performed by three observers did not detect relevant differences between manual segmentations and semi-automatic segmentations. The quality of the obtained meshes allows a distinction between the different tongue shapes, in the production of the different sounds, irrespectively of the method used for segmentation. Important features such as presence and location of grooving, shaping of the tongue dorsum (convex/concave), tongue tip position and the presence or absence of inward lateral compression can be perfectly evaluated with the meshes created using both techniques.

A quantitative evaluation based on total volume comparison pointed to an excellent reliability of both methods but this metric does not account for spatial differences. The values obtained for similarity using the Jaccard coefficient and voxelwise difference point to good levels of agreement between the semi-automatic and manual approaches (Jaccard=0.9002 and 10.495 voxel error). The one-way analysis of variance (ANOVA) shows a statistical significant difference between both methods of segmentation (Jaccard $F=18.232$, $p<.001$; voxelwise $F=18.433$, $p<.001$). This is due to the fact that the semi-automatic method is even more reproducible (Jaccard=0.9382 and 6.388 voxel error) than the manual segmentation (Jaccard=0.9170 and 8.662 voxel error). The slightly lower Jaccard coefficient observed for M vs SA is probably due to a more conservative segmentation with manual tracings. The values obtained regarding intra-operator reliability in tongue segmentations are in line with results that have been reported using ITK-SNAP to perform segmentations of other structures (e.g. for caudate nucleus, lateral ventricles). It should be noted that the quality of our images (due to very short acquisition times, 19s) is far from good, with respect to SNR and resolution. These results allow to conclude that the semi-automatic method seems to be a viable alternative to the manual method.

Furthermore, the obtained results are also relevant because we aim to compare different sounds, to quantify differences when the same sound is produced in different contexts (coarticulatory effects) or in different word position (subtle differences in tongue shape are expected in specific situations). To do so we must be quite sure that the segmentations performed are accurate and the technique is robust with respect to reliability. One of the limitations of this study is related with the absence of a second observer, precluding a more robust evaluation. Using the semi-automatic technique it is possible to explore our extensive database, in a reasonable period of time, providing valuable and newest 3D information concerning the production of EP sounds, particularly lateral sounds. At the moment, we are using PolyMeCo to compare 3D meshes from tongue images of the same speaker (different sounds) and also for quantifying differences between speakers, with very interesting results. This line of work can also be applied in more clinical settings to estimate e.g., tongue and soft palate volumes in patients with obstructive sleep apnea, macroglossia and other orthodontic disorders (Ajaj et al. 2005).

These are still preliminary, but encouraging, results and further work is still needed to fully validate the segmentation procedures described in this paper. This should include: one more observer allowing to assess inter observer reliability, increase the number of segmentation pairs and to consider other factors and interactions between factors in a more comprehensive statistical evaluation.

ACKNOWLEDGEMENT

This research was supported by FCT PhD grants of the first and third authors (SFRH/BD/65183/2009) (SFRH/BD/38073/2007) and HERON II Project (PTDC/EEA-PLP/098298/2008). Authors thank IBILI Director and technical staff of the MRI unit, as well as to the speakers involved in the study.

REFERENCES

- Ajaj, W. et al, 2005. Measuring tongue volumes and visualizing the chewing and swallowing process using real-time TrueFISP imaging - initial clinical experience in healthy volunteers and patients with acromegaly. *European Radiology*, Vol. 15, No. 5, pp. 913-918.
- Badin, P. et al, 1998. A Three-Dimensional Linear Articulatory Model based on MRI Data. *Proceedings of International Conference on Spoken Language Processing (ICSLP)*, Sydney, Australia, pp. 14-20.
- Badin, P. and A. Serrurier, 2006. Three-dimensional linear modeling of tongue: articulatory data and models. *Proceedings of 7th International Seminar on Speech Production (ISSP)*, Ubatuba, Brasil, pp. 395-402.
- Birkholz, P. and B. J. Kröger, 2006. Vocal tract model adaptation using magnetic resonance imaging. *Proceedings of 7th International Seminar on Speech Production (ISSP)*, Ubatuba, Brasil, pp. 493-500.
- Engwall, O. and P. Badin, 1999. Collecting and analysing two and three dimensional MRI data for Swedish. *Speech Transmission Laboratory: Quarterly Progress and Status Report (STL-QPSR)*, pp. 11-38.
- Gerig, G. et al, 2001. Valmet: A new validation tool for assessing and improving 3D object segmentation. *Lecture Notes in Computer Science*, Vol. 2208, pp. 516-528.
- Ho, S. et al, 2002. Level-set evolution with Region Competition: Automatic 3-D segmentation of brain Tumors. *Proceedings of 16th International Conference on Pattern Recognition*, Quebec City, Canada, pp. 532-535.
- Ho, S. et al, 2003. SNAP: A software package for User-Guided Geodesic Snake Segmentation. *Technical report*. North Carolina, University of North Carolina.
- ITK-SNAP. 2010. Available from <http://www.itksnap.org/> [Accessed 25 May 2010].
- Kim, Y.-C. et al, 2009. Accelerated three-dimensional upper airway MRI using Compressed Sensing. *Magnetic Resonance in Medicine*, Vol. 61, pp. 1434-1440.
- Liu, J. et al, 2003. System for Upper Airway Segmentation and Measurement with MR Imaging and Fuzzy Connectedness. *Academic Radiology*, Vol. 10, pp. 13-24.
- Martins, P. et al, 2008. European Portuguese MRI based speech production studies. *Speech Communication* Vol. 50, No. 11-12, pp. 925-952.
- MeVis Medical Solutions, 2003-2010. Available from <http://www.mevislab.de> [Accessed 15 March 2010].
- Narayanan, S. et al, 1997. Toward articulatory-acoustic models for liquid approximants based on MRI and EPG data. Part I. The laterals. *Journal of the Acoustical Society of America (JASA)*, Vol. 101, No. 2, pp. 1064-1077.
- Perrier, P. et al, 2003. Influences of tongue biomechanics on velar-stop consonants: A modeling study. *Journal of Acoustical Society of America*, Vol. 103, pp. 1615-1627.
- Pham, D. et al, 2000. Current Methods in Medical Imaging Segmentation. *Annual Review of Biomedical Engineering*, Vol. 2, pp. 315-338.
- Ribeiro, N. et al, 2009. 3-D solid and finite element modeling of biomechanical structures - a software pipeline. *Proceedings of EUROMECH Solid Mechanics Conference*, Lisboa, Portugal
- Silva, S. et al, 2009. PolyMeCo - An Integrated Environment for Polygonal Mesh Analysis and Comparison. *Computers & Graphics*, Vol. 33, pp. 181-191.
- Silva, S. et al, 2011. A preparatory study to choose similarity metrics for left-ventricle segmentations comparison. *Proceedings of SPIE Medical Imaging*, Florida, USA.
- Stone, M. and A. J. Lundberg, 1996. Three-dimensional tongue surface shapes of English consonants and vowels. *Journal of the Acoustical Society of America (JASA)*, Vol. 99, No. 6, pp. 3728-3737.
- Takemoto, H., 2001. Morphological Analyses of the Human Tongue Musculature for Three-Dimensional Modeling. *Journal of Speech, Language, and Hearing Research*, Vol. 44, pp. 95-107.
- Teixeira, A. et al, 2005. Simulation of Human Speech Production Applied to the Study and Synthesis of European Portuguese. *EURASIP Journal on Applied Signal Processing*, Vol. 9, pp. 1435-1448.
- Ventura, S. et al, 2008. Three-dimensional modeling of the tongue during speech using MRI data. *Proceedings of 8th International Symposium on Computer Methods in Biomechanics and Biomedical Engineering*, Porto, Portugal.
- Yushkevich, P. A. et al, 2006. User-guided 3D active contour segmentation of anatomical structures: Significantly improved efficiency and reliability. *Neuroimage*, Vol. 31, No. 3, pp. 1116-1128.

SWITCHING PAIRWISE MARKOV CHAINS FOR NON STATIONARY TEXTURED IMAGES SEGMENTATION

Mohamed El Yazid Boudaren¹, Emmanuel Monfrini² and Wojciech Pieczynski²

¹*Ecole Militaire Polytechnique, Laboratoire Mathématiques Appliquées
PO Box 17, Algiers 16111, Algeria*

²*Institut Telecom, Telecom SudParis, CITI Dpt., CNRS UMR 5157,
9 rue Charles Fourier, Evry 91000, France*

ABSTRACT

Hidden Markov chains (HMCs) have been extensively used to solve a wide range of problems related to computer vision, signal processing (Cappé, O., et al 2005) or bioinformatics (Koski, T., 2001). Such notoriety is due to their ability to recover the hidden data of interest using the entire observable signal thanks to some Bayesian techniques like MPM and MAP. HMCs have then been generalized to pairwise Markov chains (PMCs), which offer similar processing advantages and superior modeling possibilities. However, when applied to nonstationary data like multi-textures images, both HMCs and PMCs fail to produce tolerable results given the mismatch between the estimated model and the data under concern. The recent triplet Markov chains (TMCs) have offered undeniable means to solve such challenging difficulty through the introduction of a third underlying process that may model, for instance, the switches of the model along the signal. In this paper, we propose a new TMC that incorporates a switching PMC to model non stationary images. To validate our model, experiments are carried out on synthetic and real multitextured images in an unsupervised manner.

KEYWORDS

Hidden Markov chains, switching pairwise Markov chains, textured image segmentation.

1. INTRODUCTION

The hidden Markov chains have widely been applied to various image processing problems including supervised and unsupervised image segmentation. When the image under consideration is stationary, these models are quite proficient and their corresponding unsupervised Bayesian segmentation techniques using EM (Expectation- Maximization) yield suitable results. However, when the image is nonstationary, they fail to establish a satisfactory association with the image to be modeled and the segmentation results can be relatively poor. To overcome this limitation, authors in (Lanchantin, P. et al, 2011) propose a particular TMC called switching hidden Markov chains (S-HMC) to take the switches of the model into account. In such a model, each stationary part of the image is modeled via a classical HMC and the switches between these different parts are assumed to be Markovian. The same thing happens when we consider a multi-textured image; it is intuitive to consider a HMC per each texture. Furthermore, the gain in using S-HMC is that the switches between different textures are governed by a Markov chain rather than independent on each other, which may serve as a regularization tool to prevent the “pepper and salt” aspect of the unobserved image which is being determined.

In this paper, we deal with the problem of non stationary image modeling with application to multi-textured image segmentation. For this purpose, we propose an original approach, based on a switching PMC rather than a switching HMC to model textured images. Supremacy of PMCs over HMCs has already been shown in previous works (Derrode, S. and Pieczynski, W., 2004). It hinges essentially on their ability to take complex noises into account. In particular, in PMCs, one can easily model the fact that pixels near region boundaries inside the image may have different visual aspect than those located inside region.

The remainder of this paper is organized as follows: the next section summarizes the hidden, pairwise and triplet Markov chains. The third section presents the proposed switching pairwise Markov chains.

Experimental results are provided in the fourth section. Concluding remarks and future improvements end the paper.

2. HIDDEN, PAIRWISE AND TRIPLET MARKOV CHAINS

In this section, we briefly describe three families of Markov chains with strictly increasing degrees of generality: HMCs, PMCs and TMCs. Let us notice that HMCs, which are the most basic ones, were extended in different directions. However, to our knowledge, in all these extensions, the hidden process remains Markovian and the resulting models are still hidden Markov models (Pieczynski, W., 2010). On the other hand, PMCs in which the hidden process is not Markovian exist and are therefore firmly more general. Similarly, TMCs form a family which is strictly more general than PMCs since TMCs that are not PMCs exist and were used to deal with several data irregularities that neither HMCs nor PMCs can handle (Boudaren, M. E. Y., et al, 2011).

All along this section, we consider an observable signal $Y = (Y_n)_{n=1}^N$ that is to be indexed into $X = (X_n)_{n=1}^N$ where Y_n take their values in \mathbb{R} and X_n take their values from a finite set of classes $\Omega = \{\omega_1, \dots, \omega_K\}$. Realizations of the processes will be denoted by lowercase letters. To simplify the notations, we will write $p(x_n)$ instead of $p(X = x_n)$. Accordingly, we recall the formalisms of HMCs, PMCs and TMCs.

2.1 Hidden Markov Chains

A HMC is a pairwise process $Z = (X, Y) = (X_n, Y_n)_{n=1}^N$ that considers X as a Markov chain which is to be recovered from its noisy version Y . Moreover, when the classical noise assumptions hold, the joint probability of Z is given by the simple formula:

$$p(z) = p(x_1)p(y_1|x_1) \prod_{n=2}^N p(x_n|x_{n-1}) p(y_n|x_n) \quad (1)$$

Accordingly, X may be recovered from Y by means of some Bayesian decision rules such as marginal posterior mode (MPM) or maximum a posteriori (MAP) (Rabiner, L. R., 1989). Throughout this paper, MPM will be adopted. Its corresponding formula is the following:

$$[\hat{x} = \hat{x}_{MPM}(y)] \Leftrightarrow [\hat{x}_n = \operatorname{argmax}_i p(x_n = \omega_i | y)] \quad (2)$$

When the model parameters are known, the posterior distributions $p(x_n|y)$ required to perform MPM estimation are computed thanks to forward functions $\alpha_n(x_n) = p(y_1, \dots, y_n, x_n)$ and backward functions $\beta_n(x_n) = p(y_{n+1}, \dots, y_N | x_n)$ that can be computed in the following iterative way:

$$\alpha_1(x_1) = p(x_1)p(y_1|x_1);$$

$$\alpha_{n+1}(x_{n+1}) = \sum_{x_n \in \Omega} \alpha_n(x_n)p(x_{n+1}|x_n)p(y_{n+1}|x_{n+1}) \quad (3)$$

$$\beta_N(x_N) = 1;$$

$$\beta_n(x_n) = \sum_{x_{n+1} \in \Omega} \beta_{n+1}(x_{n+1})p(x_{n+1}|x_n)p(y_{n+1}|x_{n+1}) \quad (4)$$

The posterior margins can then be computed as follows:

$$p(x_n|y) \propto \alpha_n(x_n)\beta_n(x_n) \quad (5)$$

The indexing is then derived according to equation 2.

On the other hand, when the model parameters are unknown, several relatively quick algorithms can be used to find out these latter. We can cite for instance expectation- maximization algorithm (EM), its stochastic version (SEM) or iterative conditional estimation (ICE) (Pieczynski, W., 2010).

2.2 Pairwise Markov Chains

Z is referred to as a PMC if Z is itself Markovian. Hence, Z is said to be a PMC if and only if its joint distribution is given by

$$p(z) = p(z_1) \prod_{n=2}^N p(z_n | z_{n-1}) \quad (6)$$

An HMC defined by (1) can then be seen as a particular PMC where $p(z_n | z_{n-1}) = p(x_n | x_{n-1})p(y_n | x_n)$ whereas in more general PMC we have $p(z_n | z_{n-1}) = p(x_n | x_{n-1}, y_{n-1})p(y_n | x_{n-1}, y_{n-1}, x_n)$. This shows the greater generality of PMC over HMC at the local level. At the global level, the noise distribution $p(y|x)$ is of Markovian form in PMC whereas it is given by the simple formula $p(y|x) = \prod_{n=1}^N p(y_n | x_n)$ in HMC. The posterior margins $p(x_n | y)$ required for MPM restoration are computable within linear computational complexity like for HMC thanks to the same forward functions $\alpha_n(x_n) = p(y_1, \dots, y_n, x_n)$ and extended backward functions $\beta_n(x_n) = p(y_{n+1}, \dots, y_N | x_n, y_n)$ that can be computed in the following iterative way:

$$\begin{aligned} \alpha_1(x_1) &= p(x_1, y_1); \\ \alpha_n(x_n) &= \sum_{x_{n-1} \in \Omega} \alpha_{n-1}(x_{n-1}) p(x_n, y_n | x_{n-1}, y_{n-1}) \end{aligned} \quad (7)$$

$$\begin{aligned} \beta_N(x_N) &= 1; \\ \beta_n(x_n) &= \sum_{x_{n+1} \in \Omega} \beta_{n+1}(x_{n+1}) p(x_{n+1}, y_{n+1} | x_n, y_n) \end{aligned} \quad (8)$$

When the model parameters are unknown, they can be estimated via adapted variants of the same Bayesian algorithms used for HMCs. For further details, the reader may refer to (Derrode, S. and Pieczynski, W., 2004) where some related theoretical developments and experiments are shown.

2.3 Triplet Markov Chains

Z is said to be a TMC if there exists a third process $U = (U_1, \dots, U_N)$ with each U_n taking its values from a finite set $\Lambda = \{\lambda_1, \dots, \lambda_M\}$ such that the triplet $T = (X, Y, U)$ is a Markov chain. Let $V = (U, X)$. $T = (V, Y)$ is then a pairwise Markov chain (PMC). This makes the computation of the distributions $p(x_n | y)$, required to perform MPM restoration, affordable even when Z is not Markovian. This shows the greater generality of TMC over PMC, which is more general than HMC.

3. SWITCHING PAIRWISE MARKOV CHAINS

3.1 The Model

Let $Y = (Y_n)_{n=1}^N$ be an observable signal that is to be indexed into $X = (X_n)_{n=1}^N$ where Y_n are in \mathbb{R} and X_n belongs to $\Omega = \{\omega_1, \dots, \omega_K\}$. Let us consider now the situation where the data $Z = (X, Y)$ follow a pairwise Markov chain with parameters depending on the realizations of a third Markovian process $U = (U_n)_{n=1}^N$, where each U_n belongs to $\Lambda = \{\lambda_1, \dots, \lambda_M\}$. More precisely we will assume that the transition probability of Z is then given by:

$$p(z_n, u_n | z_{n-1}, u_{n-1}) = p(u_n | u_{n-1}) p(x_n | x_{n-1}, y_{n-1}, u_n) p(y_n | x_{n-1}, y_{n-1}, x_n, u_n) \quad (9)$$

This gives a particular TMC that will be referred to as a switching pairwise Markov chain (S-PMC) that extends the switching hidden Markov chain (S-HMC) (Lanchantin, P. et al, 2011). The plain PMC can be seen as a particular S-PMC where we have a unique regime and thus a unique PMC without any switches. The S-PMC may then be applied to all situations where we have regime switches provided that we can model each one via a PMC.

3.2 Application to Textured Images Modeling

We apply now our model S-PMC to model texture images. Let us consider the most general case where the image contains more than one texture. Let $I = (I_{ab})_{a,b=1}^h$ be such an image where I_{ab} is the pixel with position (a, b) and let $\Lambda = \{\lambda_1, \dots, \lambda_M\}$ be the set of M textured classes present in the image. The problem of image segmentation consists then in assigning each image pixel to one of these classes. To make our model applicable, we need first to convert the two-dimensional image I into one-dimensional signal. For this

purpose we use the Hilbert-Peano scan. This gives a mono-dimensional signal $Y = (Y_n)_{n=1}^N$ that is to be indexed into $U = (U_n)_{n=1}^N$ where Y_n are assumed to be real numbers and U_n are in Λ . According to S-PMC formalism, we have to model each texture class λ_m via a stationary PMC. Since we assume each mono-texture sub-image to be stationary, only stationary PMC in which $p(z_{n-1}, z_n | U_n = \lambda_m)$ does not depend on n will be considered. Let us denote $p_m(i, j) = p(X_{n-1} = \omega_i, X_n = \omega_j | U_n = \lambda_m)$ and $f_m^{i,j}(y_{n-1}, y_n) = p(y_{n-1}, y_n | X_{n-1} = \omega_i, X_n = \omega_j, U_n = \lambda_m)$. The distribution of Z is then given by:

$$p(z_{n-1}, z_n | U_n = \lambda_m) = p_m(i, j) f_m^{i,j}(y_{n-1}, y_n) \quad (10)$$

The distribution of the Markov chain Z can be equivalently determined by the initial probabilities $p(z_1 | U_2 = \lambda_m)$ given by $p(z_1 | U_2 = \lambda_m) = \sum_{\omega_j \in \Omega} p_m(i, j) \int_{\mathbb{R}} f_m^{i,j}(y_1, y_2) dy_2 = \sum_{\omega_j \in \Omega} p_m(i, j) f_m^{i,j}(y_1)$ and the transition matrix given by $p(z_n | z_{n-1}, U_n = \lambda_m) = \frac{p(z_{n-1}, z_n | U_n = \lambda_m)}{p(z_{n-1} | U_n = \lambda_m)} = \frac{p_m(i, j) f_m^{i,j}(y_{n-1}, y_n)}{\sum_{\omega_j \in \Omega} p_m(i, j) f_m^{i,j}(y_{n-1})}$.

In this work, only Gaussian S-PMC will be considered. An S-PMC is called Gaussian if and only if all its densities $f_m^{i,j}$ are Gaussian. Hence, an S-PMC Z can be specified through the transition matrix $A = (a_{\lambda, \lambda'})$ where $a_{\lambda, \lambda'} = p(U_n = \lambda' | U_{n-1} = \lambda)$, the M matrices $\Gamma_m = (\gamma_m^{i,j})$ where $\gamma_m^{i,j} = p_m(i, j)$ and the means $\mu_{m,1}^{i,j}, \mu_{m,2}^{i,j}$, the standard deviations $\sigma_{m,1}^{i,j}, \sigma_{m,2}^{i,j}$ and the correlation coefficient $\rho_m^{i,j}$ of the $M \times K^2$ bi-dimensional densities $f_{i,j}$.

3.3 Bayesian MPM Segmentation of a Textured Image

To accomplish the segmentation \hat{u} of the image I , one has to compute the marginal distributions $p(u_n | y)$. When the parameters of the model are given, the posterior marginal distributions $p(u_n, x_n | y)$ are workable via the forward function $\alpha_n(v_n) = p(y_1, \dots, y_n, v_n)$ and extended backward one $\beta_n(x_n) = p(y_{n+1}, \dots, y_N | v_n, y_n)$ that may be computed in the following iterative way:

$$\alpha_1(v_1) = p(v_1, y_1);$$

$$\alpha_n(v_n) = \sum_{v_{n-1} \in \Lambda \times \Omega} \alpha_{n-1}(v_{n-1}) p(u_n | u_{n-1}) p(z_n | z_{n-1}, u_n) \quad (11)$$

$$\beta_N(v_N) = 1;$$

$$\beta_n(v_n) = \sum_{v_{n+1} \in \Lambda \times \Omega} \beta_{n+1}(v_{n+1}) p(u_n | u_{n-1}) p(z_{n+1} | z_n, u_{n+1}) \quad (12)$$

The posterior distributions of V and U can then be derived as follows:

$$p(v_n | y) \propto \alpha_n(v_n) \beta_n(v_n) \quad (13)$$

$$p(u_n | y) = \sum_{x_n \in \Omega} p(v_n | y) \quad (14)$$

3.4 Parameters Estimation

When the model parameters $\Theta = (A, \Gamma_m, \mu_{m,1}^{i,j}, \mu_{m,2}^{i,j}, \sigma_{m,1}^{i,j}, \sigma_{m,2}^{i,j}, \rho_m^{i,j})$ are unknown, we propose to estimate them using the EM iterative algorithm according to the following steps:

- i) Choose an initial set of parameters $\theta^0 = (A, \Gamma_m, \mu_{m,1}^{i,j}, \mu_{m,2}^{i,j}, \sigma_{m,1}^{i,j}, \sigma_{m,2}^{i,j}, \rho_m^{i,j})^0$.
- ii) For each iteration, we compute $\psi_n(v_n, v_{n+1}) = p(v_n, v_{n+1} | y)$ and $\xi_n(v_n) = p(v_n | y)$ according to θ^q thanks to:

$$\psi_n(v_n, v_{n+1}) \propto \alpha_n(v_n) a_{u_n, u_{n+1}} p(z_{n+1} | z_n, u_{n+1}) \beta_{n+1}(v_{n+1}) \quad (15)$$

$$\xi_n(v_n) = \sum_{v_{n+1}} \psi_n(v_n, v_{n+1}) \quad (16)$$

Then we derive θ^{q+1} as follows

$$(\mu_{m,1}^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) y_n 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}} \quad (17)$$

$$(\mu_{m,2}^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) y_{n+1} 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}} \quad (18)$$

$$(\sigma_{m,1}^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) (y_n - (\mu_{m,1}^{i,j})^{q+1})^2 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}} \quad (19)$$

$$(\sigma_{m,2}^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) (y_{n+1} - (\mu_{m,2}^{i,j})^{q+1})^2 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}} \quad (20)$$

$$(\rho_m^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) (y_n - (\mu_{m,1}^{i,j})^{q+1}) (y_{n+1} - (\mu_{m,2}^{i,j})^{q+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n, v_{n+1}} \psi_n(v_n, v_{n+1}) 1_{[(u_n, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}} \quad (21)$$

$$a_{l,m}^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n} \psi_n(v_n, v_{n+1}) 1_{[(u_n, u_{n+1}) = (\lambda_l, \lambda_m)]}}{\sum_{n=1}^{N-1} \sum_{v_n} \xi_n(v_n) 1_{[u_n = \lambda_l]}} \quad (22)$$

$$(\gamma_m^{i,j})^{q+1} = \frac{\sum_{n=1}^{N-1} \sum_{v_n} \psi_n(v_n, v_{n+1}) 1_{[(u_{n+1}, x_n, x_{n+1}) = (\lambda_m, \omega_i, \omega_j)]}}{\sum_{n=1}^{N-1} \sum_{v_n} \xi_n(v_n) 1_{[u_{n+1} = \lambda_m]}} \quad (23)$$

iii) We repeat the previous step until an end criterion is reached.

4. EXPERIMENTS

In this section, we present three series of experiments. In the first one, synthetic textured images are considered and MPM segmentation is achieved in an unsupervised manner via EM algorithm according to S-HMC and S-PMC proposed here. In the second set of experiments, we judge our model via experiments conducted on one Brodatz texture image (Brodatz, P., 1966). In the last set of experiments, we consider an assorted image that we assembled by combining some mono-class texture images from CGT database. All along this section, one-dimensional chains are converted to and from 2D images using the Hilbert-Peano scan.

4.1 Unsupervised Segmentation of Synthetic Textured Images

For this set of experiments, we will consider two real classes $\Omega = \{\omega_1, \omega_2\}$ and three texture classes $\Lambda = \{\lambda_1, \lambda_2, \lambda_3\}$. Textures images of size 128×128 are generated according to S-PMC formalism as follows: $U = u$ realization sequence is fixed in the following manner: pixels of the image upper left quarter are assigned to λ_1 (black), pixels of the second quarter are assigned to λ_2 (grey) and pixels of the last half are assigned to λ_3 (white). The realizations $X = x$ and $Y = y$ are simulated from $\Omega = \{\omega_1, \omega_2\}$ (where ω_1 is depicted in black and ω_2 in white) and \mathbb{R} according to the matrices $(\Gamma_m)_{m=1}^3$ and the noise parameters as described in section 3. The parameters used for experiments 1, 2 and 3 are given in Tables 1, 2 and 3 respectively. The corresponding segmentation results are provided in Table 4.

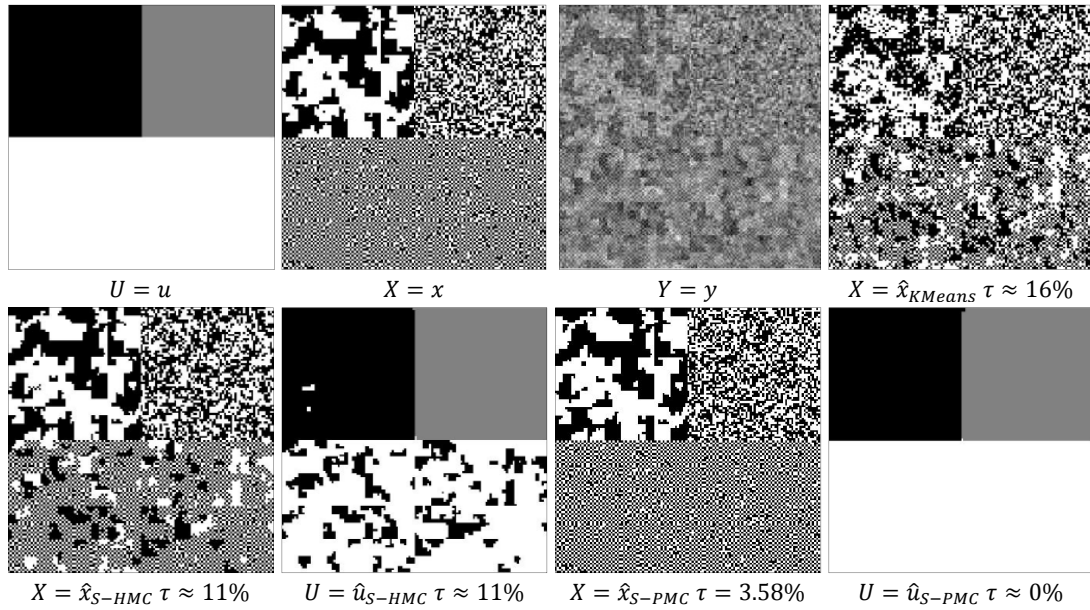


Figure 1. Unsupervised segmentation of a synthetic image from experiment 2 according to different models.

Table 1. Parameters of experiment 1

Texture	Γ	μ_1	μ_2	σ_1	σ_2	ρ
λ_1	[0.49 0.01; 0.01 0.49]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0 0; 0 0]
λ_2	[0.25 0.25; 0.25 0.25]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0 0; 0 0]
λ_3	[0.01 0.49; 0.49 0.01]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0 0; 0 0]

Table 2. Parameters of experiment 2

Texture	Γ	μ_1	μ_2	σ_1	σ_2	ρ
λ_1	[0.49 0.01; 0.01 0.49]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0.1 0.1; 0.1 0.1]
λ_2	[0.25 0.25; 0.25 0.25]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0.5 0.5; 0.5 0.5]
λ_3	[0.01 0.49; 0.49 0.01]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0.9 0.9; 0.9 0.9]

Table 3. Parameters of experiment 3

Texture	Γ	μ_1	μ_2	σ_1	σ_2	ρ
λ_1	[0.49 0.01; 0.01 0.49]	[1 1; 3 3]	[1 3; 1 3]	[1 1; 1 1]	[1 1; 1 1]	[0.5 0.5; 0.5 0.5]
λ_2	[0.25 0.25; 0.25 0.25]	[2 2; 4 4]	[2 4; 2 4]	[1 1; 1 1]	[1 1; 1 1]	[0.5 0.5; 0.5 0.5]
λ_3	[0.01 0.49; 0.49 0.01]	[0 0; 2 2]	[0 2; 0 2]	[1 1; 1 1]	[1 1; 1 1]	[0.5 0.5; 0.5 0.5]

The interest of the first experiment is to check whether the proposed S-PMC performs well when the data are generated according to an S-HMC. As shown in Table 4, the answer is affirmative; this permits to confirm that S-PMC generalizes S-HMC. The aim of the next experiments is to provide examples of situations where the former S-HMC is not suitable. In experiment 2, synthetic texture classes are governed by different transition matrices and noise correlation coefficients. In experiment 3, textures have different transition matrices and means values but the same covariance matrix. As illustrated in Table 4 and Fig. 1, the S-PMC yields significant gain in segmentation accuracy.

Table 4. Misclassification rates of synthetic textured images using different Markov chains models

Experiment		KMeans	S - HMC	S - PMC
1	τ_X	15.26	4.58	4.61
	τ_U	-	0.00	0.00
2	τ_X	15.93	10.95	3.58
	τ_U	-	10.96	0.00
3	τ_X	22.75	17.03	13.25
	τ_U	-	3.83	0.00

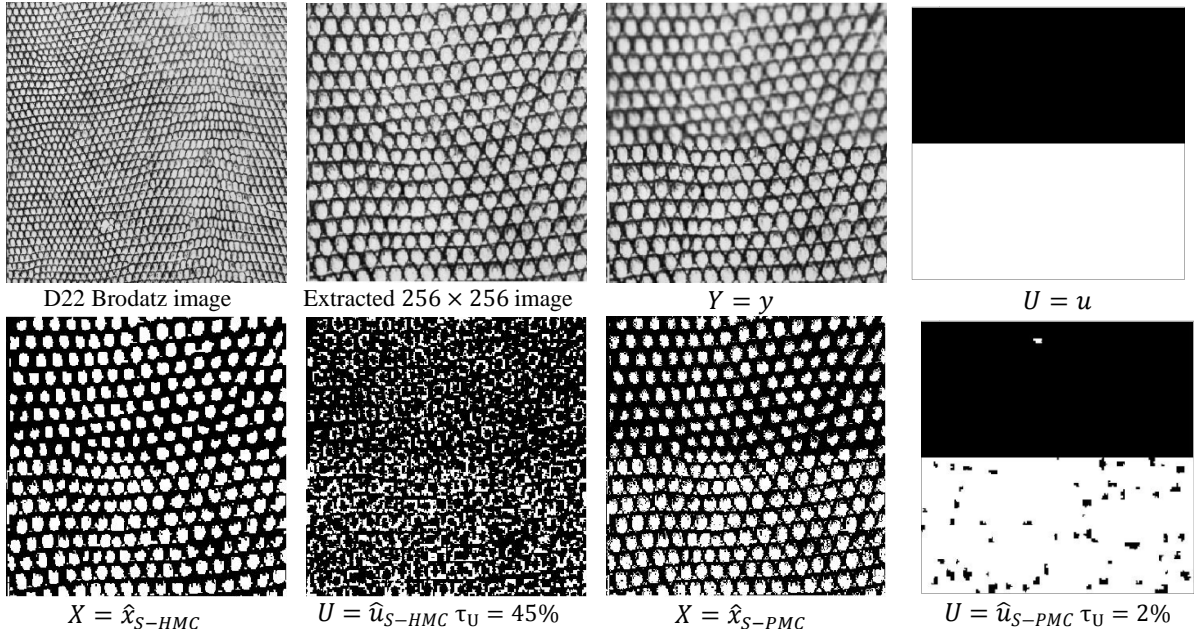


Figure 2. EM-MPM segmentation of D22 texture image according to S-HMC and S-PMC models.

4.2 Unsupervised Segmentation of Textured Images hidden with Average Filter

For this experiment, we consider the D22 texture image of Brodatz album from which we extract a sub-image of size 256×256 . Then, we apply an average filter on the upper half of the image, which gives two stationary parts (Fig. 2). We perform then MPM-EM unsupervised segmentation according to both S-HMC and S-PMC formalisms. For both models, we consider two real classes $\Omega = \{\omega_1, \omega_2\}$ and two texture classes $\Lambda = \{\lambda_1, \lambda_2\}$. The obtained results are shown in Fig.2.

The segmentation results confirm the ability of S-PMC to take more complex noises into account. This latter property is of crucial importance since it permits to distinguish between two stationary parts of the same texture and having almost the same visual aspect.

4.3 Unsupervised Segmentation of Mosaic Textured Images

For this experiment, we consider an assorted textured image that we assembled using four different mono-textured images from CGT database (Fig. 3). Segmentation is achieved through MPM-EM according to both S-PMC and S-HMC models where $\Omega = \{\omega_1, \omega_2, \omega_3\}$ and four texture classes $\Lambda = \{\lambda_1, \lambda_2, \lambda_3, \lambda_4\}$.

As shown in Fig. 3, the segmentation results demonstrate the supremacy of S-PMC over S-HMC when applied to images presenting different texture classes. In fact, as shown in Fig. 3, S- PMC allows to satisfactorily recover the auxiliary process realization $U = u$ ($\tau_U = 2\%$) whereas S-HMC confounds textures λ_2 (dark grey) and λ_4 (white). This is due to the fact that these latter present analogous visual aspects. Hence, noise correlation represents an important complementary feature that permits to discriminate similar texture classes that classical models fail to differentiate.

5. CONCLUSION

In this work, we have proposed an original model designed for textured images segmentation. Our main contribution was to extend the recent switching- HMC by substituting the PMC model for the classical HMC. This extension offers more modeling potential while the parameters estimation and MPM restoration remain workable via some Bayesian methods. The efficiency of our new model was assessed against the former one

via some experiments conducted on synthetic and real texture images. It turned out that the S-PMC yields better segmentation results. As perspectives, let us cite two promising directions for further improvements. We may mention the use of the Markov fields instead of the Markov chains, which seems promising according to some preliminary results given in (Benboudjema, D., and Pieczynski, W., 2007). Another direction may concern the family of Markov trees, which can be applied to image segmentation, in particular when multiresolution images are concerned (Bouman, C., and Shapiro, M., 1994).

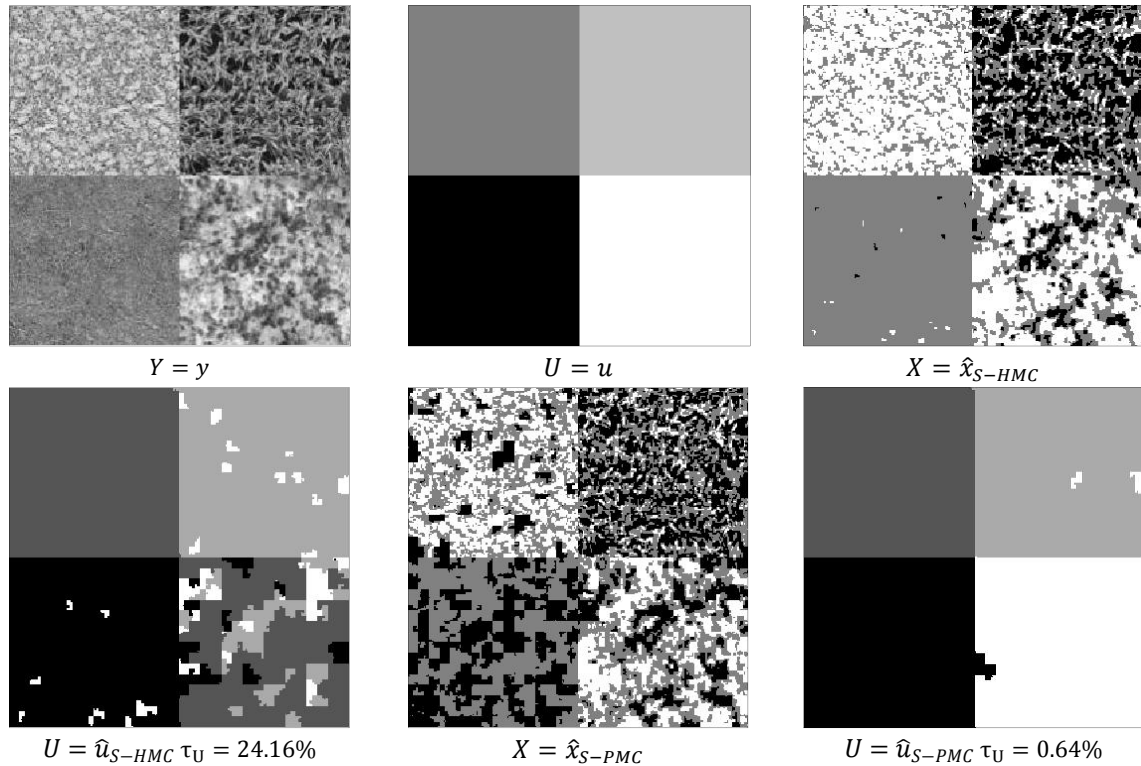


Figure 3. EM-MPM segmentation of assembled textured image according to S-HMC and S-PMC models.

REFERENCES

- Benboudjema, D. and Pieczynski, W., 2007. Unsupervised statistical segmentation of non stationary images using triplet Markov fields, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol. 29, No. 8, pp. 1367-1378.
- Boudaren, M. E. Y., et al, 2011. Unsupervised Segmentation of Non Stationary Data Hidden with Non Stationary Noise, *Proceedings of IEEE Workshop on Systems, Signal Processing and their Applications*, Tipaza (to appear).
- Bouman, C., and Shapiro, M., 1994. A Multiscale Random Field Model for Bayesian Image Segmentation. *In IEEE Transactions on Image Processing*, Vol. 3, No. 2, pp. 162- 177.
- Brodatz, P., 1966. A Photographic Album for Artists and Designers. *Textures*, Dover Publications, New York.
- Cappé, O., et al, 2005. Inference in hidden Markov models, Springer, New York.
- Derrode, S. and Pieczynski, W., 2004. Signal and image segmentation using Pairwise Markov chains. *In IEEE Transactions on Signal Processing*, Vol. 52, No. 9, pp. 2477-2489.
- Koski, T., 2001. Hidden Markov Models for Bioinformatics, Kluwer Academic Publishers.
- Lanchantin, P. et al, 2011. Unsupervised segmentation of randomly switching data hidden with non-Gaussian correlated noise, *Signal Processing*, Vol. 91, No. 2, pp. 163-175.
- Pieczynski, W., 2010. Triplet Markov chains and image segmentation. *In Inverse problems in Vision and 3D Tomography*, Chapter 4, A. Mohammed-Djafari ed., Wiley.
- Rabiner, L. R., 1989. A tutorial on hidden Markov models and selected applications in speech recognition. *In Proceedings of IEEE*, vol. 77, no. 2, pp. 257-286.

TEXTURE COMPRESSION IN MEMORY AND PERFORMANCE-CONSTRAINED EMBEDDED SYSTEMS

Jens Ogniewski¹, Andréas Karlsson² and Ingemar Ragnemalm¹

¹*Information Coding Group*

²*Computer Engineering*

Linköping University

581 83 LINKÖPING, Sweden

ABSTRACT

More embedded systems gain increasing multimedia capabilities, including computer graphics. Although this is mainly due to their increasing computational capability, optimizations of algorithms and data structures are important as well, since these systems have to fulfill a variety of constraints and cannot be geared solely towards performance. In this paper, the two most popular texture compression methods (DXT1 and PVRTC) are compared in both image quality and decoding performance aspects. For this, both have been ported to the ePUMA platform which is used as an example of energy consumption optimized embedded systems. Furthermore, a new DXT1 encoder has been developed which reaches higher image quality than existing encoders.

KEYWORDS

Embedded systems, texture compression

1. INTRODUCTION

Many modern embedded systems were not primarily designed with complex computer graphics in mind, yet are supposed to handle graphics as well, e.g. for user interfaces or simple computer games. Since most of these systems are also constrained by a comparably slow bus, which is furthermore shared between different computing units, an efficient image and texture compression method with low complexity is in dire need.

Already in (Beers et al., 1996) the requirements for such a compression method were pointed out:

- Low complexity of the decoder and high decoding speed
- Ability to access each pixel in the texture randomly (since objects may be obscured and oriented arbitrarily)
- Lossy compression is tolerable, since a screen image is made up of a multitude of different textures, so that a visual loss in one of them is hardly noticeable. Furthermore, lossy compression leads to a higher compression factor.
- A great majority of the used textures are or can be generated beforehand, so the encoding speed is only of a minor concern.

For embedded systems with limited performance and memory, low complexity is by far the most important factor, and vector quantization is a natural choice for this task.

Vector quantization has been used in image compression for several decades. The main idea is that instead of using the whole color space, a few colors will be selected which closely represent the colors contained in the image. Each pixel is then represented by an index into the list of these colors (the so called codebook). An overview of early algorithms can be found in (Nasrabadi and King, 1988). Later examples include (Buhmann et al., 1998) and the more recent (Horng et al., 2011).

Adoptions of vector quantization codecs include block truncation coding (Delp and Mitchell, 1979) and color cell compression (Campbell et al., 1986), on which most texture compression approaches are based upon. Here, the image is divided in non-overlapping blocks of equal size. Each block has an own list of a few colors, and further colors are computed by combination of these (see further section 2). The color of each pixel in the block is then encoded by an index determining which of these colors should be used. The first

commercially used texture compression was based on these ideas and was called S3TC at first, but became later the DXT family of texture compression standards. Note that the different DXT codecs only differ in the way how they handle transparency, which will not be considered in this paper. The way they encode the color information is however the same (with the exception of DXT1 if it is used with an alpha channel).

PVRTC (PowerVR Texture Compression), another often used texture compression method, however follows a slightly different, more complicated approach. It was introduced by Fenney (2003) and is aimed mainly towards embedded systems with a comparably high graphical performance, which normally includes a complete (though often relatively small) GPU. It offers one of the highest image qualities as well.

PVRTC includes an upsampling step: instead of encoding color vectors, two color maps are built which are of the size of the original image divided by the blocksize (e.g. if the blocksize is 4x4, the width and the height of the color map will be 1/4th of the original width and height). During decoding, the color maps will be upsampled by linear interpolation so that their sizes match the size of the original image. Each decoded pixel can then have the color found at its position in either color map, or a combination of these. This has two drawbacks:

- PVRTC is based on the assumption that the system includes a GPU with a special texture memory that offers hardware support for the upsampling step. However, that is not the case for embedded systems without or with a too simple dedicated GPU, which increases the runtime of the decoder heavily.

- For the decoding of each block 8 different base colors need to be loaded (4 for each color map) instead of only 2, thus decreasing the performance of random access and increasing usage of the bus.

Another texture compression method is ETC (Ericsson Texture Compression, former PACKMAN) which was introduced in (Ström and Akenine-Möller, 2005) and refined in (Ström and Petterson, 2007). Here, instead of calculating colors based on interpolation a color is given, which is modified based on modifiers contained in a codebook. This codebook only contains a luminance component and is fixed, but different ones are available which can be chosen on a block-by-block basis. Furthermore, instead of using blocks of size 4x4, 2x4 blocks are used in ETC.

Other works on texture encoding specializes on different applications, like (Roimela, 2008) on hdr textures (which uses YUV rather than RGB color space) and (Rasmusson et al., 2010) on textures with smooth edges for lightmaps.

Although modern graphic cards in desktop computers contain enough memory to make texture compression unnecessary, it is at least still very important in embedded systems (like e.g. mobile phones, portable multimedia players, settop boxes) due to the memory limit and comparably slow buses, which are often furthermore shared between many different components. An example for these small embedded systems is the energy consumption optimized ePUMA (embedded Parallel DSP with Unique Memory Architecture) platform (Ragnemalm and Liu, 2010), which consists mainly of a master CPU with 8 tightly coupled SIMD (Single Instruction Multiple Data) units which each has access to its own scratchpad memory.

The remainder of this paper is organized as follows: 2. introduces the ePUMA platform in more detail, 3. describes the decoding process for DXT1 and PVRTC, 4. gives an overview of a new encoder for DXT1, 5. shows an evaluation and 6. finishes with a discussion and ideas for future work.

In the following, whenever PVRTC is mentioned, it is referring to its 4bpp mode. The 2bpp mode will not be discussed further in this paper. Furthermore, transparency will not be discussed, and therefore both PVRTC and DXT1 are only considered in their modes without transparency.

2. THE EPUMA PLATFORM

The ePUMA platform is a processor for embedded systems optimized for low energy consumption, which is currently in development and up until now only exists in a simulator. It consists of a master CPU which act as controller for 8 SIMD processors which each has 8 parallel 16-bit datapaths. These can alternatively be used as 4 datapaths of 32-bit width. 8-bit datatypes were not supported yet during the writing of this paper, but are to be included in future implementations. An overview of the platform is given in figure 1. In it, N1 to N8 are the network nodes which handle communication between the different SIMDs, the master CPU and the main memory. PM is the program memory, CM a constant memory and PT are permutation tables designed to assist fast, parallel and conflict-free accesses to the LVMs (local vector memories) even with irregular addressing.

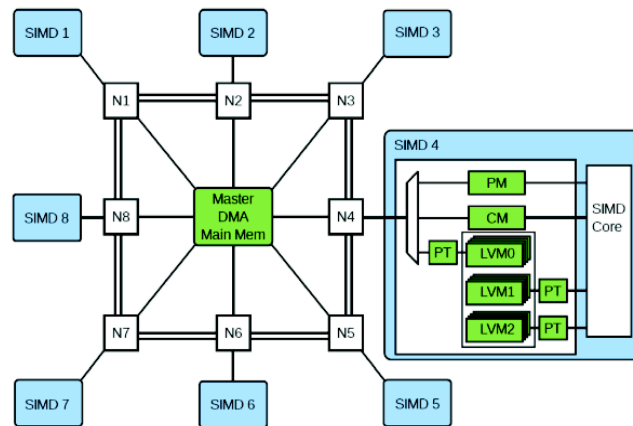


Figure 1. Overview of the ePUMA platform.

The SIMD units build the DSP subsystem of the platform and serve as its primary computing unit. Apart from the main memory, which is globally shared, each SIMD core possesses its own local memory, which can be used as a programmer controlled scratchpad. This memory is organized in three so called local vector memories (LVM), which is used to hide the main memory latency. A possibility would be to use one LVM for input data, one for output data, and load the input data for the next task into the third memory. Another possibility would be to use one LVM as input/output for the current task, one as persistent memory, and one to write back the output of the former task / load the input for the next task.

Since the processor is not primarily designed to serve as a GPU, it lacks dedicated memories for textures or the framebuffer. Thus, these have to be placed in the main and local memories.

Access to the main memory is quite slow compared to the local memory and can therefore become a bottleneck. Furthermore, the size of the local vector memories is limited, 80kB in the current implementation. Considering RGB only and that the pixeldata does not need to be aligned any further, the local memory can only hold 6 64x64 textures or 26 32x32 textures. If transparency is to be used, or mipmapping, the number of textures the local memory can hold will be further reduced. Considering 2 levels of mipmapping (which should be enough for most purposes), this is further reduced to only 5 64x64 textures resp. 21 32x32 textures. Additionally, in most applications the local memory cannot be used solely as a texture memory, thus decreasing the number of textures it can hold even further.

This induces the need for a good texture compression scheme which minimizes the access to the main memory without introducing much runtime for the decoding. Using DXT1 or PVRTC decreases the size of the texture by a factor of 6, thus relaxing memory usage significantly. Ideally, it should be possible to decode the textures “on the fly”, directly before they are used, thus eliminating the need of storing the decoded textures.

More information about the ePUMA platform can be found in (Liu et al., 2010) and (Ragnemalm and Liu, 2010).

3. DECODING

As already mentioned, modern texture codecs are vector quantization based, and use combinations of the encoded color vector to derive additional colors. In most cases 2 colors (c_1 and c_2) are encoded and 2 other ones (c_3 and c_4) are calculated through their combination, thus giving 4 different colors in total. Intuitively, evenly spaced combinations come to mind, i.e.:

$$\begin{aligned} c_3 &= \frac{1}{3} * c_1 + \frac{2}{3} * c_2 \\ c_4 &= \frac{1}{3} * c_2 + \frac{2}{3} * c_1 \end{aligned}$$

In fact, these are exactly the weights which are used in many different texture compression standards, like DXT1. PVRTC on the other hand uses weights of $\frac{3}{8}$ and $\frac{5}{8}$ instead. This is thought to be more advantageous since it more closely resembles the normal distribution:

$$\begin{aligned}c3 &= \frac{3}{8} * c1 + \frac{5}{8} * c2 \\c4 &= \frac{3}{8} * c2 + \frac{5}{8} * c1\end{aligned}$$

Both PVRTC and DXT1 use 16 bits to encode one color. In PVRTC, the first bit is used to switch the alpha channel on or off (i.e. switch between RGB and RGBA). The remaining 15 bits encode the color channels. In RGB mode, each of the three channels are described by 5 bits, thus decreasing the resolution by 3 bits per color channel from the more common 8 bit. DXT1 on the other hand uses the whole 16 bits for color information, giving the green channel a higher precision of 6 bits since the human eye has a higher sensitivity for green than for red and blue. The switch between RGB and RGBA is a little more complicated in the DXT coding standards, but since the alpha channel is neglected in this report it is not further explained here.

During decoding, the scaling of the compressed colors to convert them back to the 24-bit RGB colorspace can be combined with the multiplication of the weights as described above.

Since 4 different colors are used, 2 bits are needed for the indexing. The blocksize is set to 4x4, and therefore 32 bits are needed for the indexing of each block. Adding the 32 bits for the 2 color vector gives 64 bits for each block, or 4 bits per pixel (corresponding to a compression factor of 6). Here the disadvantage of PVRTC in terms of random access becomes obvious. Although it uses 4 bits per pixel as well, altogether 160 bits (8 color vector plus the indexes) need to be loaded to decode one block instead of only 64, due to the upsampling step.

4. TEXTURE ENCODING

We have developed and evaluated our own encoder for DXT1 texture compression, called *line matching*. In the following, the encoder will be described, with an evaluation in the next section.

If omitting their spatial position (which is already encoded in the order of the color indices), the pixels of a block can be described as points in a three dimensional space, where each of the axes represents one of the color channels. The 2 encoded color vectors then span a line in the color space, which includes the calculated color vectors as well. A simple (but effective) encoding scheme can then be constructed by first determining the best fitting line (i.e. the line which minimizes the sum of its distance to the different colors), and then searching along the line to find the points on the line which minimizes the distance to the color values of the current block.

For the first step a total least square approach was chosen, which is known to deliver an overall optimal line (up to a certain accuracy), and is used in many other different applications as well. It determines a line of form $l = s * x + a$, with a , l and s a vector, x a scalar. The intercept a is set to the mean color of the block. A matrix M is formed which consists of the different colors, each subtracted by the mean. Then a matrix $N = 1/n * M * M^T$ is built, and decomposed using singular value decomposition (SVD) to the form $U \Sigma V^*$, where U is a 3x3 matrix, and Σ a 3x3 diagonal matrix. The slope s of the line l is set to the column vector of U which corresponds to the largest absolute value in Σ . During this work, the standard SVD decomposition of the GNU Scientific Library was used.

Next, x -values are computed for every color of the block, which lead to the point on the line which is nearest to the current color. Both the total highest and the total lowest x -values are kept, and all possible values between (and including) these two x -values are then used to create possible candidates for $c1$ and $c2$. For each candidate pair, the distance between the block after encoding and the original block is computed.

A special case occurs if all values of s are zero, (e.g. when all pixels in the block have the same color). In this case $c1$ and $c2$ are set to the average, and the second step is left out.

To increase the performance of the encoding, every possible combination of $c1$ and $c2$ should only be checked once. Since $c1$ and $c2$ have a fixed resolution, the candidates have to be quantized during the search, and candidates should be rejected immediately, if they are leading to the same value after quantization as another configuration that has already been checked. Furthermore, combinations in which the values for $c1$ and $c2$ are simply switched should only be evaluated once to avoid unnecessary computations.

Initially, a third step was included in the encoder, which consisted of a fullsearch around the final values $c1$ and $c2$ to find a local optimum. However, this lead to an increase of less than 0.03% in point of the PSNR and less than 0.005% in terms of SSIM, and was therefore omitted in the final version. Intuitively, the optimal candidates (in Terms of PSNR) for $c1$ and $c2$ should lay on the best fitting line, but no proof to this

effect is known to the authors. The fact that a fullsearch around the final candidates lead to slightly better results is however not a contradiction of this assumption either. They might also be introduced by a low accuracy of the SVD algorithm, or due to inaccurate rounding and quantization during the search along the best fitting line; in the experiments, the encoder proved to be very sensitive to how this was exactly implemented. Furthermore, in the case were all pixels contain the same color, setting c_1 and c_2 to this color is not an optimal solution either, since the calculated values of c_3 and c_4 may actually reach a higher resolution than 6 bits, due to their multiplication with non-integer weights.

Table 1. Cycle counts for the decoding of one block with DXT1 and PVRTC

Task	DXT1	PVRTC
Unpacking	3	3
Calculation of color the vectors	3	99
Waiting for pipeline to finish	8	8
total	14	110

5. EVALUATION

Decoders for both DXT1 and PVRTC have been implemented for the ePUMA platform. The cycles needed to decode one 4x4 block are given in table 1. As suspected, PVRTC needed much longer time for the task, 33 times if only the time needed to compute the color vectors is taken into account, and nearly 8 times as long for the whole decoding. Since the last step (waiting for the pipeline to finish) needs only to be done once, independent on the number of blocks that will be decoded, the performance gain if using DXT1 will probably be more than 8 in practice.

Note that it is not necessary to copy the different colors to each pixel, since this can be quite efficiently combined with the next step in the graphics processing pipeline.

Although decoding speed is the most important criterion for a texture compression scheme used in embedded systems, it is not the only one. To test the image quality of both codecs, several test sequences have been encoded and compared. These sequences consist of all sequences in the MIT vision texture testbench (MIT) which contains color images with a resolution of 512x512 resolution. An additional test sequence was added containing several standard images (simply called images in the following) which are used in image and texture compression, namely the standard Lena picture, the lorikeet picture which is made available by Simon Fenney and the Lichtenstein picture which is published under the wikicommons license. All in all, 170 different images have been evaluated.

For PVRTC, the reference encoder provided by Imagination Technologies (PVRTC) was used. For DXT1, we used both our own encoder, as described in section 3, and an open source implementation called Squish (Squish). Squish is considered to reach high image quality and is close to the implementation used by NVIDIA in their texture encoding solution.

Apart from signal-to-noise ratio, the mean structural integrity (SSIM) (Wang et al., 2004) has also been computed for all images. SSIM is designed as a difference index for luminance, contrast and structure between two images. Its calculated value varies between -1 (completely different) to +1 (perfect match). A standard window size of 8 was chosen for SSIM, and all possible image patches have been tested.

The mean results for the different sequences for both SNR and SSIM are presented in table 2. The standard deviation for each is given in parentheses. As can be seen, while neither algorithm performs best in all sequences, line matching lead to the best overall result. A further subjective evaluation of the image quality is left to the reader. For that, encoded versions of an example image is give in figure 2 along with the difference images in figure 3. The difference images contain the squared difference between the original image and each encoded image.

According to our tests, PVRTC does not give a noticeably better image quality than the DXT1. In some extreme cases it even introduces visible artifacts, as can be seen in the sky of figure 2. Although the used PVRTC encoder could be optimized for better image quality, it is doubtful if the quality will ever compensate for the much higher decoding complexity. Therefore, we conclude that DXT1 is the preferable choice for texture encoding.

Table 2. PSNR and SSIM for the used encoders – PVRTC reference encoder, line matching and squish. Higher values are better, standard deviations are given in parentheses. The respective best results are marked in red.

Testsequence	PSNR			SSIM		
	PVRTC ref.	Line matching	Squish	PVRTC ref.	Line matching	Squish
Bark	27.86 (3.07)	28.04 (3.25)	27.86 (3.07)	0.980 (0.006)	0.981 (0.003)	0.980 (0.003)
Brick	30.73 (2.41)	31.21 (2.38)	30.94 (2.27)	0.978 (0.004)	0.981 (0.002)	0.980 (0.002)
Buildings	29.78 (2.19)	30.59 (2.28)	30.73 (2.01)	0.978 (0.003)	0.983 (0.002)	0.983 (0.002)
Clouds	34.45 (0.02)	35.51 (0.15)	34.60 (0.11)	0.967 (0.002)	0.974 (0.000)	0.971 (0.000)
Fabric	24.82 (2.51)	25.20 (2.41)	25.21 (2.35)	0.974 (0.005)	0.980 (0.004)	0.978 (0.004)
Flowers	29.98 (3.36)	29.86 (2.95)	29.62 (2.86)	0.982 (0.004)	0.983 (0.002)	0.982 (0.002)
Food	27.18 (4.82)	26.61 (4.80)	26.50 (4.48)	0.982 (0.005)	0.981 (0.004)	0.981 (0.004)
Grass	23.62 (4.67)	23.95 (4.48)	23.96 (4.31)	0.980 (0.003)	0.981 (0.000)	0.981 (0.000)
Images	30.47 (0.34)	30.71 (0.81)	30.47 (0.77)	0.976 (0.008)	0.975 (0.005)	0.973 (0.001)
Leaves	26.81 (3.73)	26.70 (3.67)	26.46 (3.55)	0.982 (0.004)	0.983 (0.003)	0.981 (0.003)
Metal	21.90 (1.13)	22.54 (1.00)	22.63 (1.03)	0.972 (0.003)	0.977 (0.002)	0.977 (0.002)
Misc	28.90 (2.97)	29.26 (3.21)	29.05 (3.17)	0.980 (0.003)	0.981 (0.002)	0.980 (0.001)
Paintings	27.70 (2.19)	27.16 (2.23)	26.81 (2.21)	0.977 (0.007)	0.976 (0.007)	0.974 (0.007)
Sand	29.46 (1.65)	29.21 (1.69)	29.07 (1.66)	0.981 (0.004)	0.981 (0.002)	0.979 (0.002)
Stone	28.48 (2.55)	28.64 (2.61)	28.36 (2.50)	0.973 (0.009)	0.978 (0.007)	0.974 (0.007)
Terrain	34.20 (0.72)	34.59 (0.83)	33.98 (0.76)	0.985 (0.001)	0.984 (0.001)	0.981 (0.001)
Tile	30.25 (2.40)	30.66 (2.66)	30.42 (2.55)	0.977 (0.006)	0.981 (0.005)	0.980 (0.004)
Water	31.65 (2.87)	32.23 (3.17)	31.81 (2.98)	0.979 (0.003)	0.982 (0.002)	0.980 (0.003)
WheresWaldo	26.30 (1.15)	25.81 (0.79)	25.59 (0.86)	0.984 (0.004)	0.983 (0.002)	0.981 (0.004)
Wood	31.77 (3.38)	31.77 (4.41)	31.32 (4.26)	0.984 (0.005)	0.984 (0.002)	0.982 (0.002)
average	28.45 (4.05)	28.60 (4.12)	28.39 (3.93)	0.979 (0.006)	0.980 (0.005)	0.979 (0.005)



Figure 2. Original (top left), encoded with PVRTC (up right), encoded with squish (bottom left) and with line matching (bottom right).

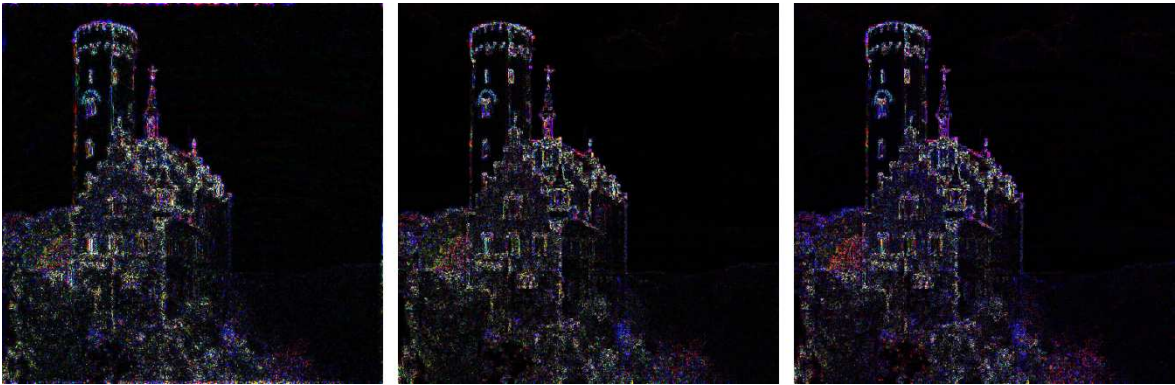


Figure 3. Sum of squared differences between original and encoded image for PVRTC (left), line matching (middle) and squish (right).

6. CONCLUSION AND FUTURE WORK

Two different popular texture decoders were implemented for the ePUMA platform, namely DXT1 and PVRTC. Both have been compared in decoding time and image quality. As suspected, the more complicated PVRTC leads to a much higher decoding time, nearly 8 times higher, which might in reality be even higher still since only the decoding of one block at a time was considered, thus rendering an efficient use of the pipeline impossible in the case of DXT1. In fact, the DXT1 decoder is fast enough to enable decoding “on-the-fly”, i.e. to decode the texture directly before it is used and therefore eliminating the need of storing the decoded textures in the local memory.

A comparison of quality aspects of the encoded images shows that PVRTC does not produce an overall better image quality over DXT1. Rather, our new encoder for DXT1, introduced in this paper, leads to the best results. Furthermore, the results were very close and each encoder produced high quality images. Therefore, the much higher decoding complexity of PVRTC is hard to justify. And even an optimized PVRTC encoder with higher image quality is not likely to change that. Therefore, the authors recommend the use of DXT1 as texture codec for embedded systems. Note that the different encoding schemes of the DXT family use the same encoding of the color channel, thus the results of this paper can be directly applied to its other members as well.

The line matching encoder was not much optimized for a fast encoding speed. Although it should be fast enough for all possible non-realtime application (on average the encoding of one image took 21.4s during the experiments), further optimization is possible and desirable. Especially the search along the best fitting line could be improved, which might even heighten the quality slightly. A SVD algorithm which is more optimized to this application (e.g. which omits the calculation of unnecessary values) should lead to better results as well.

Furthermore, although the overall graphical quality is high, blocking artifacts can occur. Note that this problem does not occur if using PVRTC, since neighboring blocks share color values due to the upsampling process. However, another encoding approach for DXT1, which encodes several neighboring blocks at once might solve this problem, although this might come with the price of reduced signal-to-noise ratio.

For simplicity, this paper concentrated on pure color images and did not include transparency. Another analysis should be done with this in mind to find out which of the different DXT1 transparency modes that delivers the best results. This could also be compared to the way PVRTC handles transparency. A comparison with the ETC texture compression could be done as well.

REFERENCES

- Beers, A. C. et al., 1996. Rendering from Compressed Textures. *Proc. SIGGRAPH 7*. New York, USA, pp 373-378
- Buhmann, J. M. et al., 1998. Dithered Color Quantization. *Proceedings of Computer Graphics Forum 17*. Oxford, UK, pp C219–C231.
- Campbell, G. et al., 1986. Two Bit/Pixel Full Color Encoding. *ACM SIGGRAPH Computer Graphics*, Vol. 20, No. 4, pp 215-223.
- Delp, E. and Mitchell, O., 1979. Image Compression Using Block Truncation Coding. *In IEEE Transactions on Communications*, Vol. COM-2, No. 9, pp 1335-1342.
- Fenney, S., 2003. Texture Compression Using Low-Frequency Signal Modulation. *Proceedings of ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*. Aire-la-Ville, Switzerland, pp 84 - 91.
- Hornig, M.-H. et al., 2011. Image Vector Quantization Algorithm via Honey Bee Mating Optimization. *In Expert Systems with Applications*, Vol. 38, No. 3, pp 1382-1392.
- Liu, D. et al., 2010. Parallel Programming and Its Architectures Based on Data Access Separated Algorithm Kernels. *In International Journal of Embedded and Real-Time Communication Systems*, Vol. 1, No. 1, pp 64-85.
- MIT Vision Texture Database, <http://vismod.media.mit.edu/vismod/imagery/VisionTexture/>
- Nasrabadi, N. M. and King, R. A., 1988. Image Coding Using Vector Quantization: A Review. *In IEEE Transactions on Communications*, Vol. 36, No. 8, pp 957-971.
- PVRTC, <http://www.imgtec.com/powervr/insider/powervr-pvrtexlib.asp>
- Ragnemalm, I. and Liu, D., 2010. Towards Using the ePUMA Architecture for Hand-Held Video Games. *Proceedings of IADIS International Conferences on CGVCVIP, VC and WEB3DW*. Freiburg, Germany.
- Rasmusson, J. et al., 2010. Texture compression of light maps using smooth profile functions. *Proceedings of the Conference on High Performance Graphics*. Aire-la-Ville, Switzerland, pp 84-91.
- Roimela, K. et al., 2008. Efficient High Dynamic Range Texture Compression. *Proceedings of the 2008 symposium on Interactive 3D graphics and games*. New York, USA, pp 207-214.
- Ström, J., and Akenine-Möller, T., 2005. iPACKMAN: *High-Quality, Low-Complexity Texture Compression for MobilePhones*. Proceedings of the ACM SIGGRAPH/EUROGRAPHICS conference on Graphics hardware, New York, USA, pp 63 - 70.
- Ström, J. and Pettersson, M., 2007. ETC2: Texture Compression using Invalid Combinations. *Proceedings of ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware*. Aire-la-Ville, Switzerland, pp 143-152.
- Squish, <http://code.google.com/p/libsquish/>
- Wang, Z. et al., 2004. Image quality assessment: from error visibility to structural similarity. *In IEEE Transactions on Image Processing*, Vol. 13, No. 4.

ACCURATE MULTI-MODAL IMAGE REGISTRATION USING COMPRESSION

Pere-Pau Vázquez¹ and Jordi Marco²

¹*MOVING Group, Universitat Politècnica de Catalunya - C/ Jordi Girona, 1-3, E-08034, Barcelona, Spain*

²*Dep. LSI, Universitat Politècnica de Catalunya - C/ Jordi Girona, 1-3, E-08034, Barcelona, Spain*

ABSTRACT

Image registration is an important task in medicine, especially when images have been acquired by different scanner/sensor types, since they provide information on different body structures (bones, muscles, vessels...). Several techniques have been proposed in the past, and among those, Normalized Mutual Information has been proven as successful in many cases. Normalized Compression Distance has been proposed as a simple yet effective technique for image registration. It is especially suitable for the case of CT-MRI registration. However, other image modalities such as PET pose some problems and do not achieve accurate registration. In this paper we analyse and propose a valid approach for image registration using compression that works properly for different combinations of CT, MRI and PET images.

KEYWORDS

Medical Image Registration, Image Compression, Visualization.

1. INTRODUCTION

Image registration is the process of overlaying two or more images that represent the same information taken at different times, from different viewpoints, and/or by different sensors. Aligning medical images has interest for the analysis of temporal patient evolution, the fusion of multimodal images, inter-patients comparison, and so on. We concentrate on multimodal images: images acquired with different processes, such as Computerized Tomography (CT), Positron Emission Tomography (PET), or Magnetic Resonance Imaging (MRI). Moreover, we will also consider only rigid transformations of the images. Our objective is to develop a method for robust multi modal image registration that achieves Mutual Information level results (one of the most prominent approaches in literature is Viola & Wells, 1997), requires little user intervention, and, if possible, is faster than previous methods. We use Normalized Compression Distance to solve the image registration problem. This measure has been previously used to address CT-MRI registration Bardera et al., 2010 based on the use of *bzip2*. Unfortunately, other image modalities, such as PET, yield poor results: registration point does not necessarily correspond to the correct one and, more importantly, the distance function has a high number of local minima, which will easily trap any optimization process. We improve this technique and achieve robust CT-MRI-PET registration by fundamental changes:

- Identification of the most suitable real-world compressor for image registration
- A new approach for image scan that provides better results than previous methods.
- Data quantization: Quantizing the input images allows us to improve registration quality and speed.

2. BACKGROUND

Normalized Compression Distance is a universal metric of distance between sequences. The **Kolmogorov complexity** ($K(x)$) of a string x is the length of the shortest binary program to compute x on a universal computer (such as a universal Turing Machine). Thus, $K(x)$ denotes the number of bits of information from which x can be computationally retrieved. As a consequence, strings presenting recurring patterns have low complexity, while random strings have a complexity that almost equals their own length. Hence, $K(x)$ is the

lower-bound of what a real-world compressor can possibly achieve. The **conditional Kolmogorov complexity** $K(x|y)$ of x relative to y is the length of a shortest program to compute x if y is provided as an auxiliary input. Both Kolmogorov complexity and conditional Kolmogorov complexity are machine independent up to an additive constant. Bennett et al. (1998) define the **information distance** between two, not necessarily equal length binary strings as the length of the shortest program that, with input x computes y , and with input y computes x . The information distance is a metric, up to negligible violations of the metric inequalities. Li et al., 2004 present a normalized version of information distance, the *similarity metric*, defined as: $d(x,y) = \max(K(y/x), K(x/y)) / \max(K(x), K(y))$. The authors also prove that it is also a metric, and that this metric is universal: two files of whatever type similar with respect to a certain metric are also similar with respect to the similarity metric. Being Kolmogorov complexity not computable, it may be approximated with the use of a real-world compressor, leading to the Normalized Compression Distance (NCD): $NCD(x,y) = (C(xy) - \min(C(x), C(y))) / \max(C(x), C(y))$, where function $C(F)$ is the size of the compression of a certain file F , and xy is the concatenation of files x and y . Although the similarity metric has values in $[0..1]$, NCD values are usually in the range of $[0..1.1]$, due to compressor imperfections. NCD has been used for applications such as language classification and handwriting recognition in Cilibrasi & Vitanyi, 2005. They also analyse the conditions that compressors must fulfil in order to be used for computing the Normalized Compression Distance: i) **Idempotency**: For a repetition of a string, the compressor should be able to detect the repetitions and thus compress the file to a similar size than the original string compression $C(xx)=C(x)$, and $C(\lambda)=0$ where λ is the empty string. ii) **Monotonicity**: The concatenation of two strings should yield to a less compressible file than taking a single string alone, up to a certain precision: $C(xy) \geq C(x)$. iii) **Symmetry**: $C(xy)=C(yx)$. Compression should be symmetric, that is, changing the order of the concatenated strings should not affect the length of the compression. iv) **Distributivity**: $C(xy)+C(z) \leq C(xz)+C(yz)$. Real-world compressors seem to satisfy this property. Compressors with these properties are named *normal compressors*. Most real-world compressors fulfil those properties, at least to a point where they are usable for NCD computation. NCD has been used for music clustering (Cilibrasi et al., 2004), automatic construction of the phylogeny tree based on whole mitochondrial genomes (Li et al., 2001), the automatic construction of a language trees (Li et al., 2004). However, its use is not trivial, using NCD on the raw data may not yield good results, (Tran, 2007, Rocha et al., 2006). NCD has also been used for image classification (Cilibrasi & Vitanyi, 2005) with grey-scale images. Lan & Harvey, 2005 show that the measure performs better than histogram-based approaches in object recognition using PPM-based compression. A further work by Li & Zhu, 2006 improves the optimization task by Lempel-Ziv encoding and using either the dictionary, or the compressed patterns for measuring image similarity. Bardera et al., 2006 use Normalized Compression Distance for image registration. They select a *window* of pixels in one image and another one in the other reference image. Then, pixels are interleaved forming a new image where the red channel holds the pixels of reference image 1 and green channel the ones of reference image 2. These images are then compressed using JPEG 2000 and the compressed size is used as $C(xy)$ in NCD equation. They also present a second approach where the gray-scale values are treated as elements of a string, and *bzip2* is used to compress the resulting string. Again, the values of both images are interleaved. Although this approach works for the CT-MRI registration, it has problems for the PET-MRI image pair. The existing bibliography does not perform an in-depth study on the suitability of the different compressor families for any of the image comparison problems addressed. Image scan directions and concatenation building have also not been analysed, further than pointing out different possibilities (Bardera et al. (2006), Macedonas et al. (2008)). This paper intends to provide some answers to these questions for the concrete problem of medical image registration.

3. MULTIMODAL REGISTRATION USING COMPRESSION

3.1 Compression Algorithms

In general, a data compression algorithm focuses in identifying and extracting data redundancy. There are several ways to do this. In short, some of the main ideas under the standard compressors:

- **Huffman coding**: The core idea is to assign a fixed-length code to each symbol. The most frequent the symbol, the lower number of bits it is assigned.

- **Arithmetic coding:** The general idea is to replace a stream of input symbols with a single floating-point number. Compared to symbol coding, this method wastes a lower amount of bits.
- **Dictionary-based compression:** These methods encode variable-length strings of symbols as single tokens. These tokens are indices to a phrase dictionary. Thus, the compressor looks for redundant substrings.
- **Block-sorting compression:** It is based on a transformation that permutes the order of the characters. After the transformation, repeated characters are grouped together. This allows for other techniques that work on repeated characters (i.e. run-length encoding) to be applied to the transformed string.
- **Prediction by Partial Matching:** Adaptive statistical data compression technique that uses a set of previous symbols in the uncompressed stream to predict the next symbol in the stream (Cleary et al., 1995).

The main issue with multimodal image registration is that we are not looking for similarities but for *correlation*. This is due to the fact that images from different acquisition methods do not show the same information; on the contrary, it is somewhat complementary (bones versus soft tissues...). Therefore, the same grey level may indicate different information and be placed at different positions. That is why Mutual Information achieves so good results. Our objective is to identify the best compression scheme (if any) for image registration, and to determine which image sources combinations are feasible.

The Normalized Compression Distance works on strings. However, images contain 2D information that is not trivially changed to a 1D string (Macedonas et al., 2008). Thus, some possibilities arise, such as building a 1D sequence from an image by taking pixels row-by-row, or column-by-column. Macedonas et al., 2008 have tested both cases and stated that the row-by-row approach yields slightly better results. This is the approach also followed by Li & Zhu, 2006. Bardera et al., 2006 use *zig-zag* scan, and point as a future work the analysis of other 2D to 1D sequence conversion such as the use of space filling curves. In this paper we also analyse if the 2D to 1D transformation has any impact when using compressors for image registration. We will test different 2D to 1D sequence transformation methods: i) row-by-row, ii) column-by-column, iii) Space Filling Curves, and iv) random sampling.

We will see that random sampling improves image registration. Moreover, the *concatenated* file can be built by several different ways. The original formulation uses concatenation. However, Bardera et al., 2006 showed that pixel by pixel interleaving yielded good results. To the best of our knowledge, no previous study has addressed the effect of different ways to construct the *concatenated string*. Thus, we have evaluated several methods: i) Image concatenation, ii) Pixel interleaving, and iii) N-Pixel interleaving: the concatenated image is created by building groups of n pixels from each image. We used values of $n=2$ and $n=4$.

3.2 Experimental Setup

In this section we analyse the behaviour of different real world compressors for Normalized Compression Distance-based image registration using CT-MRI-PET source data. The experiments are performed by taking a pair of images that, otherwise indicated, are almost aligned. From those images, one is taken as the source, and the other one as the destination. In order to make the plots more intuitive, we only show transformations involving translations in both X and Y directions. For the tests, we move the source image over the destination one by applying translations from $(-10, -10)$ to $(10, 10)$. Of course, the destination image has an extra frame of 10 pixels width (in background colour) around the original image, and we clip the destination image according to the position of the moving one. The results are shown as a 3D chart where we plot the Normalized Compression Distance using different compressors and parameterizations. Since the important point is the minimum value, and the function shape, due to the lack of space, we clipped the plots in order to show only the informative parts of the distance function. The minimum distance is the registration point. In order to compare the results, we will analyse the same image pair using Normalized Mutual Information. The desired result is a function that decreases as we approach the matching point and this one coincides with the one found with Normalized Mutual Information. Note that, for several reasons, the images coming from two different capture devices do not exactly sample the same regions in space. As a consequence, some small translation from the matching point given by NMI may be visually acceptable. A very important key issue is that the function should not be plagued with local minima, as this may challenge the optimization process.

Although we analysed a high number of compressors, we will only show the results from the ones that show better behaviour: a block-based compressor: *bzip2*, a compressor based on the Prediction by Partial Matching scheme: *paq8px*, one of the most effective data compressors, in terms of compression rate, according to Bergmans, 2011, but time consuming. Since some of the approaches are orthogonal, such as

image scan versus image combination, throughout the paper we will analyse different configurations and incrementally incorporate them in the following sections, for the sake of the reading experience.

3.3 Image Concatenation versus Pixel-by-Pixel Interleaving

Bardera et al., 2006, 2010 showed that pixel-by-pixel interleaving was a good means to achieve CT-MRI registration with a block-based compressor. Despite that, the image combination possibilities were not deeply discussed. As a consequence, we first analyse the performance of different schemes using the same compressor. In contrast to the original approach, which analysed the image using *zig-zag*, we perform the image scan row-by-row, as we obtain equivalent results. An analysis on different image scans is presented next. The first experiment is CT-MRI registration. The images used are shown in Figure 1-left.

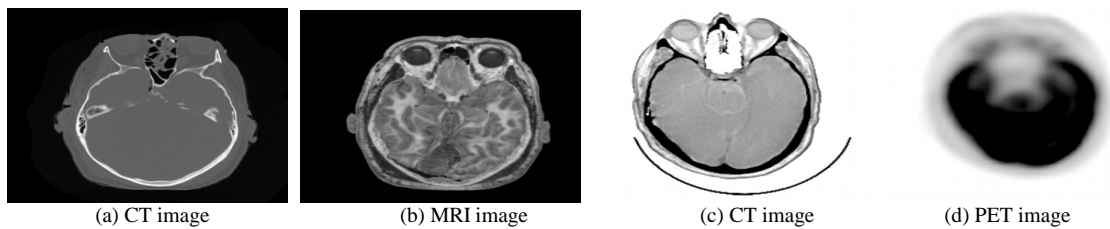


Figure 1. Left: CT and MRI pair to register. Right: CT-PET image pair.

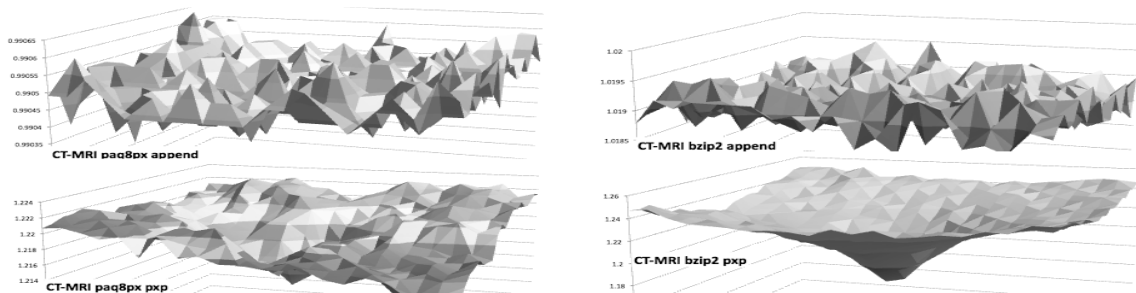


Figure 2. CT-MRI registration using different image combination methods: Regular file append (top for *paq8px* and *bzip2*, respectively) and pixel-by-pixel interleaving (bottom for *bzip2* and *paq8px*, respectively).

In Figure 2 we compare the different behaviour in registration when using image appending and image interleaving, respectively. Note that, independently on the quality of the matching point, pure image concatenation leads to a high number of local minima. From now on, the experiments shown will incorporate pixel-by-pixel interleaving.

3.4 2D to 1D Image Transformation

An image is a 2D data structure. For file writing, this information is transformed to a 1D array. This transformation can be done in different ways. So far, we have scanned the input images row-by-row. However, some other alternatives have been pointed out in literature. Since no experimental evidence on which method would yield better results for image registration, we have tested several approaches: row-by-row, column-by-column, space filling curves, and (pseudo-)random sampling.

Row-by-row and column-by-column have already been used for image comparison, with little advantage for row-by-row according to Macedonas et al., 2008. Bardera et al., 2006 suggested using Space Filling Curves. The rationale behind this is the fact that such curves take advantage of spatial coherence.

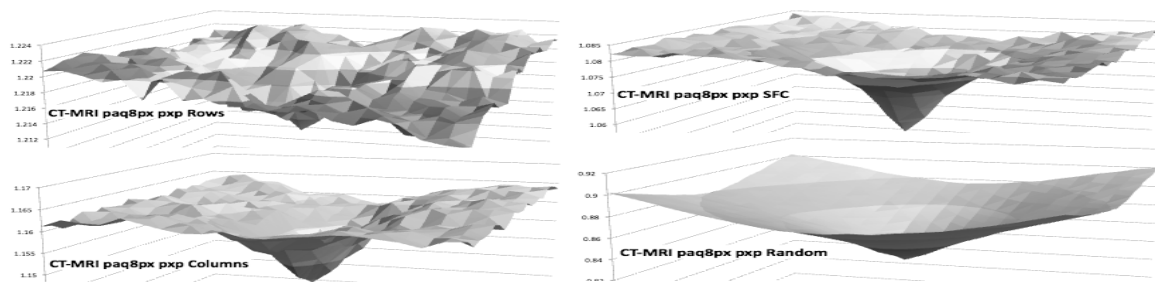


Figure 3. CT-MRI registration using different image scan strategies and *paq8px*. Column-by-column yields better results than row-by-row. However, the best matching is always found using random scan (bottom right): the matching point is good, and the function shape is soft.

From our experiments, we found that none of the previous approaches was optimal for image registration, as in most cases the function still contains a high number of local minima (Figure 3). Then, we came up with a totally different solution: random sampling the input images. We build the combined image by pixels selected from pseudo-random positions in the input images (the randomly selected position is the same for both input images). The reasoning behind is, when we are addressing image registration, we are not looking for local coherence, but pixel correlation between the two input images. Space Filling Curves may worsen the results because the information contents may vary greatly from one image to the other, and therefore, we are, somehow, counteracting the compressor task. For dictionary-based compressors, the dictionary construction might benefit from having a larger set of smaller words, than a smaller set of larger words. This is what may be induced by random image sampling. Actually, this technique improves image registration with dictionary-based compressors such as *paq8px* and *gzip*, and even block-based compressors such as *bzip2*.

We show the results of these different configurations in Figure 3 for the CT-MRI pair in Figure 1-left. Note that random scan (bottom) produces higher quality results than with other techniques, both in terms of function shape, and a more accurate matching of the registration point. Row-by-row and column-by-column often obtain different, contrary results, but none is always better than the other. Sometimes row-by-row scan generates a softer shape, and sometimes column-by-column generates a better one. The results are unpredictable. We have not been able to find a consistent behaviour throughout the different tests. Moreover, except for the random sampling, none of the previous image scan methods improves the registration point.

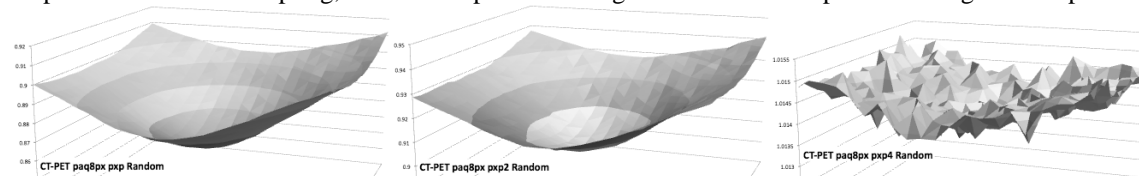


Figure 4. CT-PET image registration using *paq8px* and different ways to combine image pixels interleaving: single pixel interleaving (left), 2-pixel interleaving (center), and 4-pixel interleaving (right). Note how the best results are obtained with single pixel interleaving.

As already commented, we previously experimented with different interleaving strategies with no positive results. In order to assess our intuition supporting the random sampling strategy, we experimented again with N -pixel interleaving. If our intuition is truth, incrementing the number of pixels taken into account in the interleaving process should worsen the registration function. We used the CT-PET image pair in Figure 1-right. This is shown in Figure **Erro! A origem da referência não foi encontrada.** where pixel interleaving sizes are compared: one (pxp), two (pxp2) and four (pxp4) pixels. The method that obtains better registration is always 1 pixel wide image combination. This enforces our idea that correlation is better captured if we randomly sample the input images.

3.5 Improving Registration using Image Quantization

When addressing image registration using Normalized Mutual Information, the number of bins selected for the histograms does influence the registration results. More concretely, the registration may improve if we

select a lower number of bins, say 100 for instance. Furthermore, a lower number of bins also accelerates the registration algorithm because the joint histogram is sensitively simpler. This bin reduction also makes the algorithm more robust to the presence of noise. In our experiments we follow the same idea. We quantize the input images to 16 bins (as shown in Figure 5). This reduces the noise and the amount of information, but the important details are not removed. As a consequence, registration results are improved.

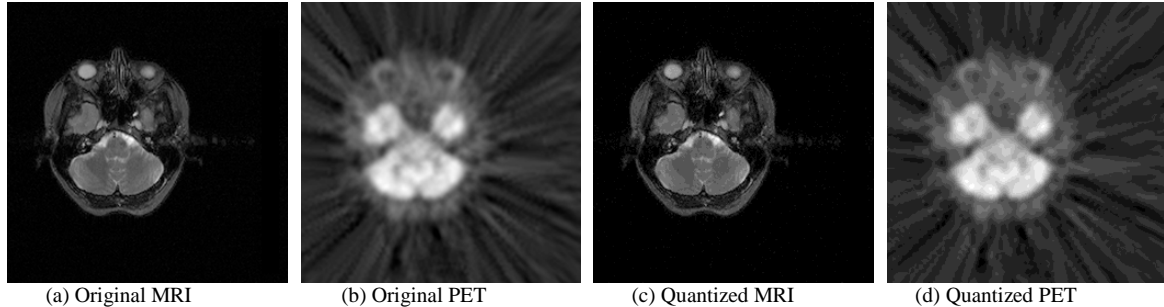


Figure 5. Quantization of the input images to 16 values. Left column shows the original images to register, and the right column shows the same images after the quantization.

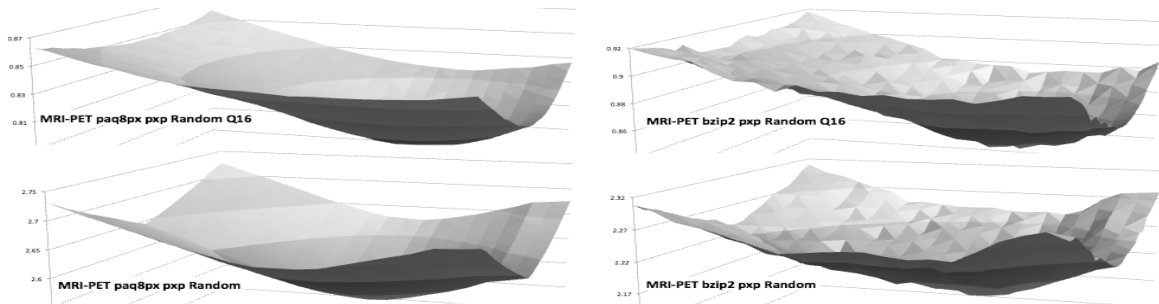


Figure 6. Comparison of MRI-PET registration using *paq8px* and *bzip2* using quantization and with unmodified image sources. Although quantization does not improve the function shape, it improves the matching point (with NCD and *paq8px* we match the result obtained by Normalized Mutual Information).

Once the images are quantized, we compare both using the Normalized Compression Distance. We set one of the two images as the source, and the other one as the destination. Again, we move the source image over the destination one by applying translations from $(-10, -10)$ to $(10, 10)$. The minimum distance is the registration point. As shown in Figure 6, we are able to correctly register the MRI-PET pair, a difficult example, since the information in the PET image is not very detailed.

3.6 Results

We have tested our registration scheme with several multimodal image pairs, and some of them already appeared in this paper. So far, we have only analysed the adequacy of the shape of the distance function. In this section we will further analyse the obtained registration points with a pair of CT-MRI, another one consisting in CT-PET, and finally, we also test an MRI-PET pair.

From now on, the experiments are carried out using all improvements: image quantization, image interleaving, and random scan. For the sake of clarity, we plot the NCD values obtained by *paq8px* compressor, since it is the one that showed most robustness in the experiments. Over the plot, which is shown as a contour chart, we indicate the registration point obtained with this compressor. We also add the registration point obtained by Normalized Mutual Information and the results we obtain with our algorithm and other compressors such as *bzip2* and *7z* (PPM-based, with results similar to *paq8px* but much faster).

Our method performs **CT-MRI registration** very efficiently (with good results even for *gzip*). Figure 7 (left) shows the results for the images in Figure 1. In this case, all the registration points computed by our

method with different compressors (*paq8px*, *bzip2*, and *7z*) correspond to the value computed using Normalized Mutual Information. In this case, *gzip* also correctly finds the registration point.

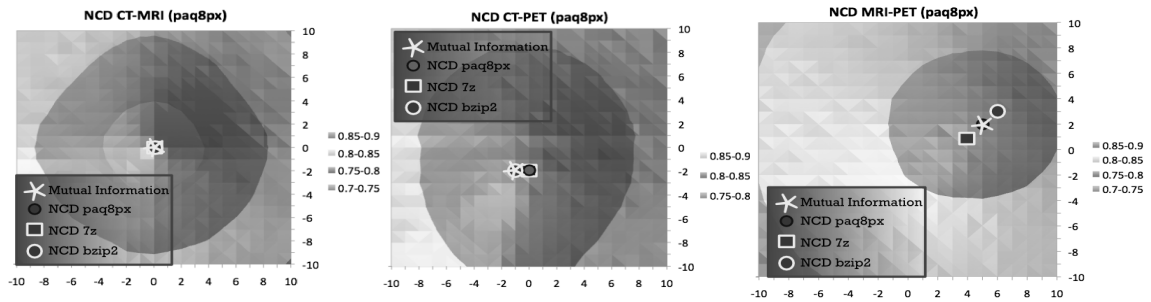


Figure 7. Different registration methods with different image modalities. Left: CT-MRI registration. All registration points coincide with the NMI method. Centre: CT-PET registration. NMI and *bzip2* achieve the same result, while using *paq8px* or *7z* obtains a point shifted only one pixel. Right: MRI-PET registration.

As already said, **CT-PET registration** is usually difficult because of the lack of details of PET images (see Figure 1-right). If we do not use our improvements, *bzip2*, is unable to match the registration point. However, with pixel interleaving and random scan, results are highly improved. Figure 7 (centre) shows the results obtained with our algorithm. Here, block-based compression obtains the same result than NMI, while *paq8px* and *7z* are shifted a pixel in the *X* direction. This is not bad at all, since, as noted previously, the images do not exactly sample the same region of the body, and both matching points are visually acceptable.

The **MRI-PET registration** is also difficult due to the lack of details in PET images. However, our method correctly finds a registration point for these kinds of source data. We show the results in Figure 7 (right). Note that *paq8px* finds the correct point according to NMI, but both *bzip2* and *7z* achieve very good results. In all cases, the matching points are visually acceptable.

4. CONCLUSION

We presented a novel approach to multi-modal image registration using compression that obtains robust results even with images such as PET. Accurate registration is obtained by modifying the *classical* NCD measurement in three ways: a) We quantize the input images, b) we scan the images in a pseudo-random manner, and c) we combine the images pixel by pixel to form the *concatenated* file.

Throughout the process, we also made other interesting findings: a) for image scan, Space Filling Curves and row-by-row or column-by-column did not give good results. b) regular image concatenation is also unsuitable for image registration. c) PPM-based compressors are more robust for image registration than other schemes. Our results are comparable to registration by Normalized Mutual Information. Hence, we believe that PPM-based compressors are a promising tool for further investigation. Though *paq8px* is very costly, another PPM-based compressor, *7z* works very fast, and has also proven useful for image registration. Its results outperform NMI times even when computing NMI with a reduced number of bins such as 100.

We also tested other compressors, such as *jpeg2000*, *rzip* (tailored to find redundancies placed at high distances), *lzma* (based on Lempel-Ziv and Markov Chain coding), *hffzip* (Huffman coding), and *gzip* (based on Lempel-Ziv). Except for *gzip*, that gives good results for CT-MRI registration, the other ones had not satisfactory results in any case. In future we want to deal with image sources such as SPECT.

ACKNOWLEDGEMENT

This project has been supported by TIN2010-20590-C02-01 of the Spanish Ministry of Science and Innovation. We want to thank the images providers, especially the "Vanderbilt Database" and Pubimage sets. This paper was partially developed during a sabbatical period by the authors at the Geometry and Graphics Group from Universitat de Girona.

REFERENCES

- Bardera, A. et al, 2006, Compression-based image registration. *IEEE International Conference on Information Theory*, pp 436–440.
- Bardera, A. et al, 2010, Image registration by compression. *Information Sciences*, 180(7), 1121 – 1133.
- Bennett, C. et al, 1998, Information distance. *IEEE Transactions on Information Theory*, 44.
- Bergmans, W., 2011. Maximum compression. <http://www.maximumcompression.com/>. [Online; accessed Feb 2011].
- Cilibrasi, R. and Vitanyi, P.M.B., 2004. Clustering by compression. *IEEE Trans. Information Theory*, 51(4):1523–1545.
- Cilibrasi, R. et al, 2004. Algorithmic clustering of music based on string compression. *Journal of Computer Music*, 28(4):49–67.
- Cleary, J.G. et al, 1995, Unbounded length contexts for ppm. *DCC '95: Proc. of the Conference on Data Compression*.
- Lan, Y. and Harvey, R., 2005, Image classification using compression distance. *Proceedings of the 2nd International Conference on Vision, Video and Graphics*, pp 173–180.
- Lawder, J.K. and King, P.J.H., 2000, Using space-filling curves for multi-dimensional indexing. *17th British National Conference on Databases*, pp 20–35, London, UK.
- Li, M. et al, 2001. An information-based sequence distance and its application to the whole mitochondrial genome phylogeny. *Bioinformatics*, 17(2):149–154.
- Li, M. et al, 2004, The similarity metric. *IEEE Transactions Informmation Theory*, 50(12):3250–3264, 2004.
- Li, M. and Zhu, Y., 2006, Image classification via lz78 based string kernel: A comparative study. *PAKDD*, pp 704–712.
- Macedonas, A. et al, 2008, Dictionary based color image retrieval. *J. Vis. Comun. Image Representation*, 19(7):464–470.
- Rocha, J. et al, 2006, Compression ratios based on the universal similarity metric still yield protein distances far from cath distances. *CoRR*, abs/q-bio/0603007.
- Tran, N., 2007, The normalized compression distance and image distinguishability. *Proceedings 19th IS&T/SPIE Symposium on Electronic Imaging Science and Technology*, San José, USA.
- Viola, P. and Wells, W.M., 1997, Alignment by maximization of mutual information. *J. Comput. Vision*, 24(2):137–154.

A HEURISTIC APPROACH TO MULTI-FOCUS IMAGE FUSION

Po-Whei Huang¹, Cheng-I Chen¹ and Phen-Lan Lin²

¹*Department of Computer Science and Engineering,*

National Chung Hsing University, Taichung 402, Taiwan

²*Department of Computer Science and Information Engineering,*

Providence University, Shalu, Taichung 433, Taiwan

ABSTRACT

In this paper, we propose a new fusion scheme for multi-focus images based on global decisive locations. A new focus measure is proposed to identify which pixels in the source images can be included in the resultant image. To achieve a better fusion result, our new focus measure evaluates the focus weights for those global decisive locations within a dynamic focus-measuring window. With our new focus measure, the problem of fake edge points caused by blurring effect which may lead to bad fused result can be reduced. Experimental results show that our fusion scheme outperforms other eight existing fusion methods in terms of PSNR.

KEYWORDS

Multi-focus image, image fusion, decisive location, focus weight, focus-measuring window, focus measure.

1. INTRODUCTION

Image fusion is a technique of combining several source images of the same scene captured by multiple sensors or a single sensor under different physical conditions into a fused image. The fused image can provide less uncertain and more abundant information for human visual perception and computerized image-processing tasks such as image segmentation, feature extraction, and target detection. In addition to provide a better interpretation for the image content than any single source image, the fused image can also improve the storage reduction of data, as well as the reliability of image content [9].

There are many image fusion techniques proposed in recent years and their practical implementations can be seen in several application fields. For example, the lower resolution multi-spectral image can be fused with the higher resolution panchromatic image to form a color and higher-resolution image in remote sensing application. In medical diagnosis, MRI (Magnetic Resonance Imaging) can provide high-resolution images with structural and anatomical information while PET (Positron Emission Tomography) can provide lower spatial resolution images with functional information. The fused result obtained from MRI and PET images carries both types of information in one image. Another example of image fusion is to combine visible images with thermal/infrared images which can capture objects with temperature difference for defense or surveillance applications.

One important type of image fusion is to deal with multi-focus visible images. Due to the limitation of depth of field of optical lens in a camera, only those objects within the depth of field of the camera are in focus while other objects are out of focus [1]. Therefore, there is a need to obtain an image that is in focus everywhere by fusing the images taken from the same view point with different focal settings to extend depth of field. Multi-focus image fusion is particularly useful to forensic applications. For example, tool impressions and fingerprints (footprints) are usually recorded by close-up photography. In close-up photography, adjusting focus is necessary and can be done by changing the distance between the specimen and the focal lens. Since the surface of the specimen is usually not smooth and the depth of field of the camera lens is limited, it is very difficult to focus every portion of the specimen at one time. By taking several multi-focus images with close-up photography, we can obtain a composite image containing a better

and complete description for the specimen by image fusion. Similar technique can be applied to AOI (automatic optical inspection) in industrial applications.

2. THE PROPOSED FUSION SCHEME

In this paper, we only concentrate on fusion for visible and multi-focus images. We propose a new fusion scheme based on a set of decisive locations with focus weights contributed from all source images. This scheme uses a new focus measure to identify the right pixels in the source images to be included in the fused image. Two novel ideas are embedded in the design of our fusion scheme: (1) only those decisive locations in the source images are considered as critical for focus-measuring calculation; (2) the size of a focus-measuring window at each individual pixel location is not fixed; instead, it is automatically and dynamically adjusted. Based on the above two ideas, our scheme can achieve a better fusion result by calculating the focus measure more precisely at each pixel location of the source images and choosing the right pixel from the in-focus region of a source image for the fused image. Therefore, the misjudgment of including wrong pixels to the fused result is reduced significantly.

Assume that we have a set of multi-focus images I_1, I_2, \dots, I_N of the same scene. We also assume that any area in the image of this scene has its best focus in some source image I_i ($1 \leq i \leq N$). Therefore, the goal of our fusion scheme is to find the best pixel at each location of the resulting image from the corresponding locations of all source images. To achieve this goal, we need a focus-measuring mechanism to help us judge which pixel of the same corresponding location will be chosen from all source images for the resulting image. Let $M^i(x, y)$ be the focus-measuring mechanism for the pixel at location (x, y) in source image I_i .

The higher the value of $M^i(x, y)$, the better the pixel $I_i(x, y)$ is in-focus. Let the size of the source images be $R \times C$, then the resulting image can be represented by F , where $F = \{I_i(x, y) \mid \forall 1 \leq x \leq R, 1 \leq y \leq C, M^i(x, y) > M^j(x, y), \forall i \neq j \text{ and } i, j = 1, 2, \dots, N\}$.

We have the following two observations for the focus measure: (1) the edge information in the surrounding area of a pixel can help us judge how well that pixel is in-focus because it is extremely difficult to judge whether a pixel is well-focused without looking at its surrounding area; (2) an defocused area may contain fake edge points which lead to misjudgment of focus measure using existing methods reported in the literature.

According to the above observations, we design our fusion scheme based on the concepts of decisive locations, effective focus-measuring window, focus weight, and focus measure to be defined as follows. Given a set of multi-focus gray-level images $I = \{I_1, I_2, \dots, I_N\}$, the set of decisive locations obtained from the first b bit-planes of image I_i is defined as $D_i(b) = \bigcup_{k=1}^b (E_i^k - \bigcap_{i=1}^N E_i^k)$, where E_i^k is the set of edge points detected from the k -th bit-plane of image I_i . The set of global decisive locations with respect to a set of multi-focus images $I = \{I_1, I_2, \dots, I_N\}$ is $D = \bigcup_{i=1}^N D_i(b)$.

Notice that decisive locations play similar roles as edge points did. However, decisive locations are different from edge points in two aspects. First, decisive locations are more stable than edge points because no threshold is needed for the former. Secondly, decisive locations are more capable in terms of discriminating “activity level” which is important to focus measure.

According to our first observation, we need an appropriate surrounding area of a pixel to determine whether that pixel is in focus. This surrounding area is called “effective focus-measuring window” defined as the following. An effective focus-measuring window for a location (x, y) with respect to a set of multi-focus image $I = \{I_1, I_2, \dots, I_N\}$ is the smallest window centered at (x, y) containing at least one global decisive location contributed from image I_i for all i with $1 \leq i \leq N$.

An effective focus-measuring window must contain one or more global decisive locations. Each global decisive location is associated with a focus weight in a source image. The focus weight is defined as below.

The focus weight at a global decisive location (u, v) in image I_i with respect to a window $S_w(u, v)$ is denoted by $g_w^i(u, v)$ and defined as the mean squared intensity difference between point (u, v) and all other points (u', v') in $S_w(u, v)$ whose intensity is different from that of point (u, v) on image I_i . That is,

$$g_w^i(u, v) = |C|^{-1} \sum_{\forall (u', v') \in C} [I_i(u', v') - I_i(u, v)]^2$$

where

$$C = \{(u', v') \mid I_i(u', v') \neq I_i(u, v) \forall (u', v') \in S_w(u, v)\}.$$

Now, we are ready to define the focus measure for a pixel in an image as follows. Given a set of multi-focus images $I = \{I_1, I_2, \dots, I_N\}$ and a set of global decisive locations D with respect to I , the focus measure for a point (x, y) in image I_i based on the effective focus-measuring window $S_z(x, y)$ is defined as

$$M_z^i(x, y) = |R|^{-1} \sum_{\forall (u, v) \in R} g_w^i(u, v)$$

where

$$R = S_z(x, y) \cap \{(u, v) \mid \forall (u, v) \in D \exists g_w^i(u, v) \geq T_i(x, y, z)\}$$

and

$$T_i(x, y, z) = |D \cap S_z(x, y)|^{-1} \sum_{\forall (u, v) \in D \cap S_z(x, y)} g_w^i(u, v)$$

Based on the above concepts and definitions, now we are ready to propose two algorithms: one is for finding an effective focus-measuring window for a given point and the other is the procedure of our fusion scheme.

Algorithm: Finding effective focus-measuring window

Input: Location (x, y) on a set of multi-focus images $I = \{I_1, I_2, \dots, I_N\}$

Output: Size z of the effective focus-measuring window

Step 1: Let R and C be the number of rows and columns of each image I_i , respectively. Let $z_{lower} = 3$ and

$$z_{upper} = \frac{1}{4} \min\{R, C\}.$$

Step 2: While $(z_{upper} - z_{lower}) > 1$

IF $S_z(x, y)$ contains at least one decisive location contributed from image I_i for all i with $1 \leq i \leq N$ THEN

$$z_{upper} = \left\lceil \frac{z_{lower} + z_{upper}}{2} \right\rceil \text{ ELSE } z_{lower} = \left\lfloor \frac{z_{lower} + z_{upper}}{2} \right\rfloor$$

$$\text{Step 3: Return } z = \left\lfloor \frac{z_{lower} + z_{upper}}{2} \right\rfloor$$

Algorithm: Image Fusion

Input: A set of multi-focus images I_1, I_2, \dots, I_N

Output: A fused image I_{fused}

1. Find the set of decisive locations $D_i(b)$ from each image I_i . Let $D = \bigcup_{i=1}^N D_i(b)$.

2. For each location (x, y) in the resulting image I_{fused}

(a) Call ‘‘Finding effective focus-measuring window’’ to get window $S_z(x, y)$.

(b) In each source image I_i , compute the focus measure $M_z^i(x, y)$ for pixel at location (x, y) using the focus-measuring window $S_z(x, y)$.

(c) Assume that the pixel at location (x, y) of source image I_k has the maximum focus measure among all pixels at this same location in all N source images. Then, let $I_{fused}(x, y) = I_k(x, y)$.

3. Return I_{fused} .

3. EXPERIMENTAL RESULTS

In physics, increasing in the magnification of a microscope makes the depth of field of a microscope become narrow. As a result, an observed object or area with an uneven surface under the microscope will be rendered with partial blur in an image. Figure 1-3 show three examples of such a phenomenon. Figure 1(a) and 1(b) respectively focus on the body and the wing of a bee while Figure 2(a) and 2(b) focusing on different parts of eye of a bee. Figure 3(a) and 3(b) shows the images of the same liver tissue with different parts in focus. The fused results of the three examples are shown in Figure 1(c)-3(c), respectively.

We also compared the performance of our fusion scheme with other eight existing fusion methods based on 15 reference images of size 256×256 in term of PSNR. The PSNR is defined as

$$PSNR = 10 \times \log_{10} \frac{255^2}{MSE} \text{ dB}$$

where $MSE = \left(\frac{1}{m \times n}\right) \times \sum_{i=0}^{i < n} \sum_{j=0}^{j < m} (c'_{ij} - c_{ij})^2$, m , n are the width and the height of image, respectively, and c , c' are the pixel intensities of the two images under comparison. Assume that the reference image has intensity c_{ij} and the fused image has intensity c'_{ij} . The reference images are blurred on different parts to emulate the defocused areas. Then, the two blurred images are fused to generate the fused image. A method with better performance should have higher PSNR. The eight methods are abbreviated as PCA [10], LAP [2][3], CON [11], OWD [7][10], NOWD [4], MWF [5], DSCP [6], and LOG [8], respectively. Among these methods, DSCP is block-based while others including ours are pixel-based approach. We tested LAP, CON, OWD and NOWD methods using decomposition level $l = 1, 2$ and 3 , respectively. Then we selected the best performance from these three levels as the performance of each method. MWF was also tested by using decomposition level $l = 1, 2$, and 3 , respectively, with the fusion rule same as the one used by Chanda [5]. The best performance among these levels was regarded as the performance of MWF. The performance of LOG method was obtained by choosing the best PSNR from level-3 decomposition with $3, 4$, and 5 orientations. Note that for those multi-resolution methods including LOG, WMF, NOWD, OWD, CON, and LAP, the averaging strategy was applied as the fusion rule for the approximation image (i.e. the base image), and choosing maximum coefficient or maximum wavelet energy as the fusion rule for detailed images. To grade the performance of DSCP method, we used DCT strategy to obtain the best fusion result as proved by Lin [6].

The result of performance comparison for these nine fusion methods is shown in Table 1. It is very clear that our proposed method outperforms other eight fusion methods in all 15 images. The PSNR value obtained from our fused results is 52.37 on average. The second best is probably the LOG method which has 46.67 as its average PSNR. It is worthwhile to mention that although DSCP is block-based, it is very competitive to LOG, NOWD, CON, and LAP methods. A block-based approach is much faster than a pixel-based approach.

4. CONCLUSIONS

In this paper, we propose a fully automatic new fusion scheme for multi-focus images based on a new focus measure evaluated by a dynamic focus-measuring window. Our proposed focus measure evaluates the focus weights on a subset of global decisive locations within a focus-measuring window so that the fake edge problem caused by blurring effect can be reduced. In our fusion scheme, a dynamic window selection method is also proposed to select feasible focus-measuring windows for focus measure computation. Experimental results showed that our fusion scheme outperforms the other eight well-known fusion methods in terms of PSNR.

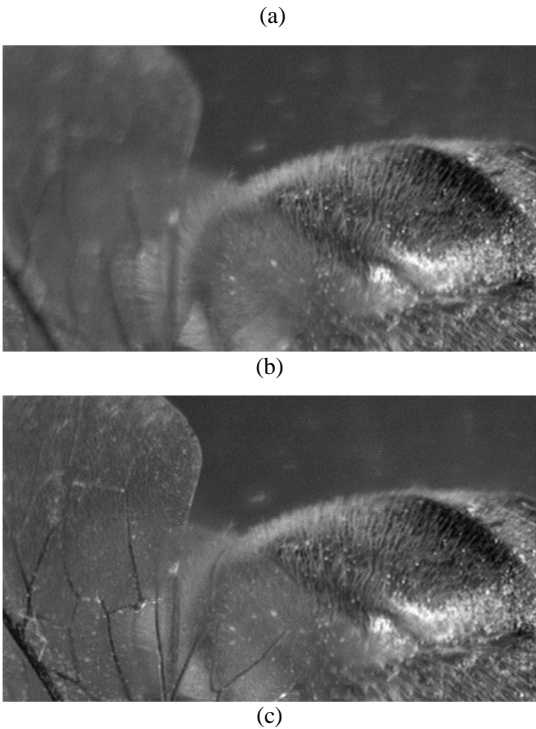


Figure 1. (a) Image focused on wing. (b) Image focused on eye. (c) Our fused result.

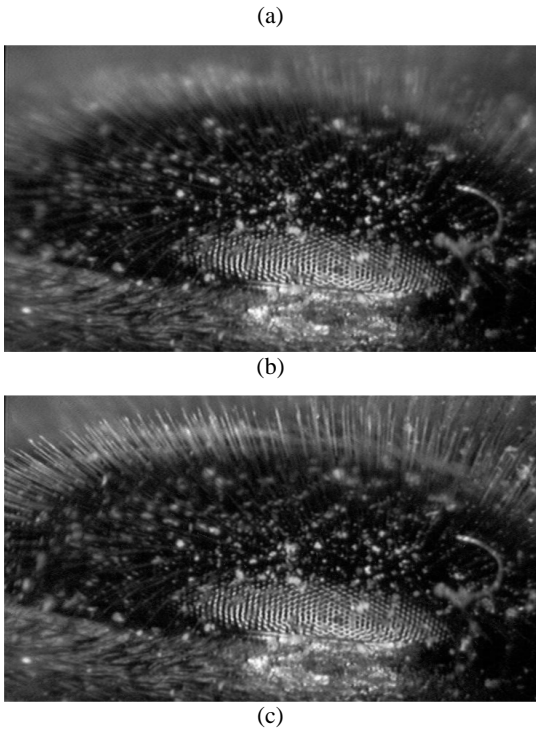


Figure 2. (a) and (b) : Two partially focused images with enlarged eye of a bee. (c) Our fused result.

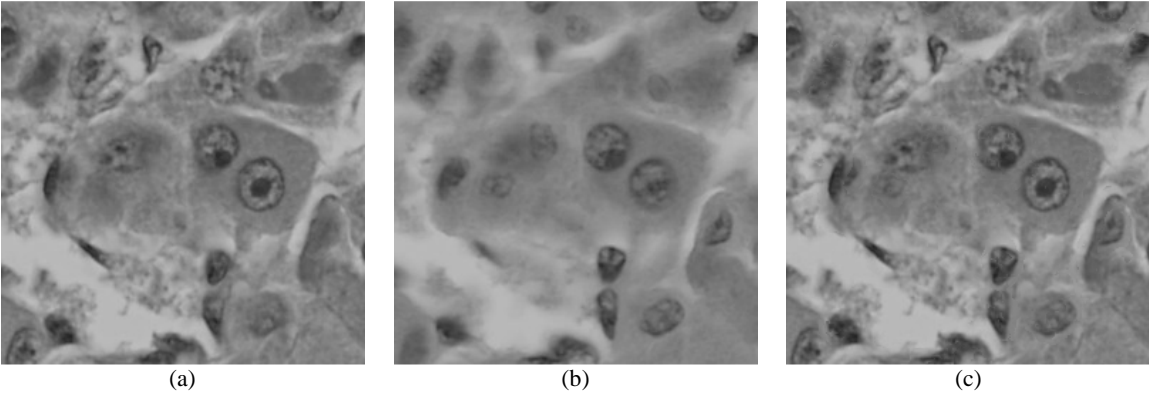


Figure 3. (a) and (b) : Two partially focused images of liver tissue. (c) Our fused result.

Table 1. Performance comparison of our method with other eight methods on 15 reference images in terms of PSNR

Methods	Ours	LOG	DSCP	MWF	NOWD	OWD	CON	LAP	PCA
S-01	49.96	41.01	42.98	36.42	42.93	41.68	41.98	41.96	33.97
S-02	51.74	45.56	40.53	37.30	45.64	43.12	42.17	42.21	34.63
S-03	50.62	42.58	40.75	37.39	45.06	42.44	41.93	41.87	33.95
S-04	52.42	46.89	45.82	38.99	46.10	44.86	44.88	44.81	37.61
S-05	55.25	50.59	46.03	39.75	47.90	45.31	43.68	43.65	36.01
S-06	51.76	42.75	40.08	38.89	46.27	44.58	43.19	43.14	36.25
S-07	49.26	47.95	44.43	39.76	46.31	45.23	45.52	45.47	37.39
S-08	50.33	48.00	42.16	35.78	43.87	41.77	41.52	41.48	33.99
S-09	50.90	50.87	44.65	37.58	44.45	43.11	43.71	43.68	36.20
S-10	51.09	44.20	44.60	37.19	42.47	42.07	44.43	44.36	36.30
S-11	52.94	44.65	44.00	37.63	45.17	44.07	44.39	44.28	37.78
S-12	51.02	48.82	46.67	40.47	47.15	46.31	46.55	46.51	40.60
S-13	53.79	48.94	46.11	41.83	48.41	46.57	46.57	46.55	40.18
S-14	54.23	46.27	49.83	39.83	46.34	45.10	46.58	46.42	39.60
S-15	60.30	50.92	49.81	38.75	44.73	44.34	46.25	46.01	38.77
Average	52.37	46.67	44.56	38.50	45.52	44.04	44.22	44.16	36.88

REFERENCES

- [1] Aslantas, V., 2007. A depth estimation algorithm with a single image. *In Optics Express*, Vol. 15, No. 8, pp. 5024-5029.
- [2] Burt, P.J. and Adelson, E.H., 1983. The Laplacian Pyramid as a Compact Image Code. *In IEEE Trans. Communications*, Vol. com-31, No. 4.
- [3] Burt, P.J. and Adelson, E.H., 1985. Merging images through pattern decomposition. *Applications of Digital Image Processing VIII*, A.G.Teschler, ed., Proc. SPIE 575, pp.173-181.
- [4] Chibani, Y, Houacine, A., 2003. Redundatnt versus orthogonal wavelet decomposition for multisensor image fusion. *In Pattern Recognition*, Vol. 36, pp. 879-887.
- [5] De, I., Chanda, B., 2006. A simple and efficient algorithm for multifocus image fusion using morphological wavelets. *In Signal Processing*, Vol. 86, No. 5, pp. 924-936.
- [6] Lin, P.L., Huang, P.Y., 2008. Fusion methods based on dynamic-segmented morphological wavelet or cut and paste for multifocus images. *In Signal Processing*, Vol. 88, No. 6, pp. 1511-1527.
- [7] Pajares, G., Manuel de la Cruz, J., 2004. A wavelet-based image fusion tutorial. *In Pattern Recognition*, Vol. 37, pp. 1855-1872.
- [8] Redondo, R. et al, 2009. Multifocus image fusion using the log-Gabor transform and a multisize Windows technique. *In Information Fusion 10*, Vol. 2, pp 163-171.
- [9] Rockinger, O., 1996. Pixel - Level Fusion of Image Sequences using Wavelet Frames. *Proceedings of the 16 th Leeds Applied Shape Research workshop*, Leeds University Press.
- [10] Smith, M.I., Heather, J.P., 2005. Review of image fusion technology in 2005. *Proceedings on Defense and Security Symposium*, Orlando, FL.
- [11] Toet, A. et al, 1989. Merging thermal and visual images by a contrast pyramid. *In Optical Engineering*. Vol. 28, No. 7, pp. 789-792.

BRAIN MEDICAL IMAGE FUSION BASED ON IHS AND LOG-GABOR WITH SUITABLE DECOMPOSITION SCALE AND ORIENTATION FOR DIFFERENT REGIONS

Phen-Lan Lin¹, Po-wei Huang², Cheng-I Chen², Tsung-Ta Tsai² and Chin-han Chan²

¹*Department of Computer Science and Information Engineering, Providence University, Shalu, Taichung 433, Taiwan*

²*Department of Computer Science and Engineering, National Chung Hsing University, Taichung 402, Taiwan*

ABSTRACT

We propose two fusion methods, denoted IHS&LG+ and IHS&LG++, based on IHS and log-Gabor wavelet for fusing PET and MRI images by choosing suitable decomposition scale and orientation for different regions of images in the first method, and refining the fused intensity of the first method to further reduce color distortion and enforce the anatomical structure in the second method. Our methods use the hue angle of each pixel in PET image to divide both PET and MRI images into regions of high and low activity. The fused intensity of each region is obtained by inverse log-Gabor transforming of high frequency coefficients of MRI intensity and low frequency coefficients of PET intensity-component. Our experimental results on three sets of normal axial, normal coronal, and Alzheimer's disease brain images demonstrate that all three images fused by IHS&LG+ are with less color distortion and about the same structural information as the images fused by IHS&RIM, and all three images fused by IHS&LG++ are with both color and anatomical structural information closest to PET and MRI images both visually and quantitatively.

KEYWORDS

PET images, MRI images, log-Gabor wavelet, hue angle, IHS transform.

1. INTRODUCTION

Brain medical imaging can be divided into two categories. One provides high-resolution structural and anatomical information in gray scale, such as magnetic resonance imaging (MRI), the other demonstrates the biochemical changes in the body in color, such as positron emission tomography (PET). Although PET images can display metabolism information in colors, its structural resolution is much less than that in MRI images. Due to distinct information contained in these two types of images, doctors often need to compare images from two or more sources to effectively diagnose a disease. Thus, fusion of both PET and MRI brain images with high fidelity not only can help doctors diagnose brain diseases with less effort, but also can reduce the storage cost by saving a single fused image instead of multiple-source images.

Many fusion methods have been presented over the past ten years. IHS substitution method [10] can provide fused images with high structural information, but the colors of the fused images are generally quite distorted. On the other hand, multi-resolution-based methods, such as DWT and its variants [2, 6, 10, 11], can provide fused images with less color distortion; however, the structural resolution of the fused images is unsatisfactory. Fusion based on IHS and RIM (IHS&RIM) [3] is a recent fusion method that produces images with less color distortion from PET images while preserving the high resolution structural information of MRI images. Nevertheless, visible color distortion can still be spotted in some areas of the fused images. Fusion using bilateral filtering [5] has also been presented recently.

In this paper, we propose two fusion methods based on IHS and log-Gabor wavelet by choosing suitable decomposition scale and orientation for different regions of images in the first method, and refining the fused intensity of the first method to further reduce color distortion and enforce the anatomical structure in the second method. Our experimental results show that our methods can indeed produce fused images with less color distortion from PET images and richer structural information in MRI than method IHS&RIM can both visually and quantitatively.

2. IMAGE FUSION BASED ON IHS and LOG-GABOR WAVELET WITH SUITABLE DECOMPOSITION SCALE FOR DIFFERENT REGIONS (IHS&LG+)

The colors in PET images reflect the organ's functional information while the various intensities in MRI images depict the organ's structural information. As shown in Fig. 1(a), regions in colors yellow, red, and white are highly active regions, while regions in blue and cyan are less active. Similarly, higher frequency structure occurs in gray matter (GM) or cerebrospinal fluid (CSF) cells while lower frequency structure occurs in white matter cells (WM), as shown in Fig. 1(b). In general, brain tumors often occur in regions of high cell activity. But, when the tumor is at its early stage, doctors would require the spatial information in MRI images for more effective diagnosis. Thus, a good fusion image of PET and MRI should preserve colors with as little distortion and spatial information with as high resolution as possible.

2.1 Analysis of Fusion Method IHS and Log-Gabor Wavelet

Image fusion using log-Gabor filter has been shown to produce promising results recently both in fusing multi-focus images [8] and in multi-sensor images [4]. The method that fuses MRI and PET images [4] is as follows. At first, the PET image is transformed to intensity, hue, and saturation components. Both the I components of PET and the MRI image are decomposed with a fixed scale and angle to high and low frequency coefficients. The high frequency coefficients of MRI and the low frequency coefficients of PET are inverse log-Gabor wavelet transformed to a fused intensity component. Finally, the fused intensity component, the hue component, and the saturation component of PET are inverse IHS transformed to a fused image of PET and MRI. Figs. 1 (c) and (d) show the result using IHS&LG fusion with a decomposition scale of 4 and 3, respectively. Notice that in Fig. 1(c), which is the result of using scale 4, the structure information is well preserved but the blue shade is noticeably lighter than the original PET, as shown in Fig. 1(a). On the contrary, blue shade in Fig. 1(d) that is the result using scale 3 is very close to that in the original PET, but the structural information in the areas in shades yellow and red is not as significant as that in Fig. 1(c).

Noticing that using higher decomposition scale in log-Gabor wavelet results in a fused image with clearer structure and using lower decomposition scale results in a fused image with less color distortion. Thus, we devise a revision of IHS&LG fusion method by dividing the brain image into regions of high and low activity, respectively, and decomposing each region with different scale to solve the aforementioned problem.

2.2 Image Separation Based on Brain Cells Activity

In IHS model, the angles of colors yellow, red, and magenta are in the range of $[-30^\circ, 90^\circ]$ in the hue plane. Since the yellow-, red-, and white-regions in PET scan images contain highly active brain cells, while the blue- and cyan-regions contain less active brain cells, the hue angle ranges for high and low brain activity pixels are $[0.92, 1]$ and $[0, 0.25]$ after normalization.

The separation of PET and MRI images into high activity region and low activity region can be briefly described as follows.

1. Apply IHS transform to the PET image and obtain the color of each pixel based on its angle value of the hue component.
2. For each pixel in the PET image, assign it to high activity region if its hue angle is within the range of $[0.92, 1]$ or $[0, 0.25]$; otherwise, assign it to low activity region.
3. For each pixel in the MRI image, assign it to high or low activity region if the pixel at the same location in the PET image is in high or low activity region.

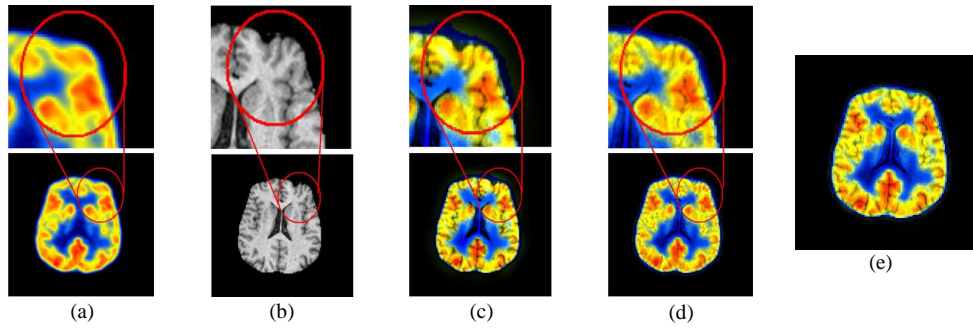


Figure 1. Log-Gabor wavelet fusion analysis with different decomposition scales. (a) PET image (b) MRI image (c) the log-Gabor wavelet fusion result with decomposition scale= 4 (d) the log-Gabor wavelet fusion result with decomposition scale=3 (e) the log-Gabor wavelet fusion result with decomposition scale=4 for high activity region and 3 for low activity region.

2.3 Fusion based on IHS and log-Gabor Wavelet with Suitable Decomposition Scales for Different Regions (IHS&LG+)

Once the input images are separated into two regions of high and low activity, respectively, the intensity of each activity region is fused, respectively, as follows.

1. The intensity component of the PET image and the intensity of the MRI image are respectively decomposed into high and low frequency coefficients by log-Gabor wavelet transform with a suitable scale for each region. As mentioned in Section 3.1, our investigation indicated that decomposition scale 4 is suitable for high activity regions of brain images, and decomposition scale 3 is suitable for low activity regions.

2. The high-frequency coefficients of the intensity of each MRI sub-band and the low-frequency coefficients of the intensity component of each PET sub-band are inverse log-Gabor wavelet transformed to result in a fused intensity component. Note that using high-frequency coefficients of the intensity of MRI and low-frequency coefficients of the intensity of PET can preserve the high spatial resolution information in the MRI image while retaining the cell activity information in the PET scan image.

3. The fused intensity components of both activity regions along with both the hue and the saturation components of the PET image are inverse IHS transformed to result in a PET and MRI fused image.

Fig. 1(e) shows the fusion result of images in 1(a) and 1(b) using decomposition scale 4 for the high activity region and scale 3 for the low activity region and angle scale 6 for both regions. Observing the new fused image, we found that it preserves the spatial structure of the MRI image nearly as well as in 1(c) and has as little color distortion from the PET image as in 1(d). The blocks except the red one in Fig. 2 illustrate the flow of method IHSLG+.

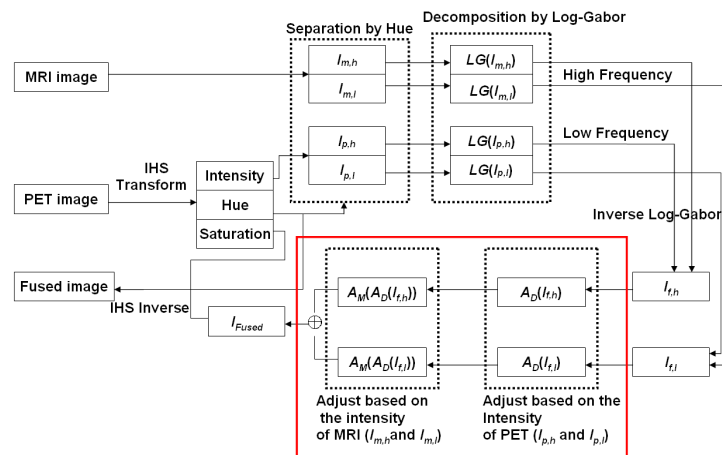


Figure 2. The block diagram of our proposed methods

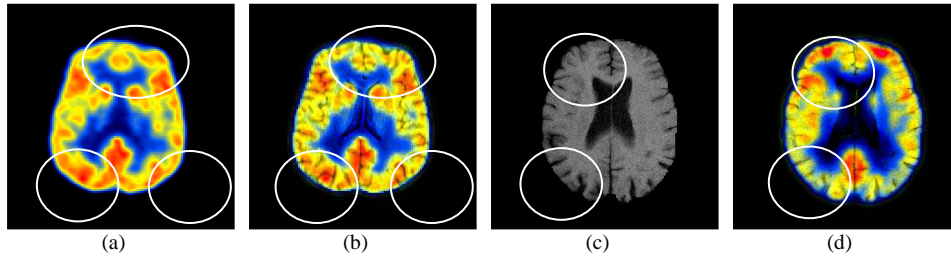


Figure 3. Color distortion and loss of structural information in the fused image of IHS&LG+ (a) PET image (b) fused image using IHS&LG+ (c) MRI image (d) fused image using IHS&LG+

3. FUSION BASED ON IHS&LG+ WITH COLOR ADJUSTMENT AND ANATOMICAL STRUCTURE ENFORCEMENT (IHS&LG++)

Although the fused image of IHS&LG+ shown in Fig. 1(e) preserves the spatial structural information of the MRI image nearly as well as in 1(c) and has as little color distortion from the PET image as in 1(d), we notice that (i) the color of some portions in the new fused image, as shown in the circled regions in Fig. 3(b), is still lighter than the original PET, as shown in 3(a); and (ii) the structure information of some cerebrospinal fluids and gray matters, as shown in the circled regions in Fig. 3 (d), is either not as clear as the original MRI image as shown in 3(c) or disappear entirely.

3.1 Color Distortion Reduction for the Fused Image by IHS&LG+

3.1.1 Color Distortion Rationale

In IHS model, the values of R, G, and B become bigger when the value of I-component is bigger and the values of both H- and S-components unchanged. And, when R, G, and B are all bigger, the color becomes lighter based on RGB color model. In our fusion method of IHS&LG+, we replace the high frequency I-component of PET with the high-frequency I of MRI. Thus, the I-component after inverse IHS transform is no longer the same as that of the original PET. That is, the I-component of some pixels becomes either larger or smaller, which results in color distortion for these pixels. In order to reduce color distortion, we propose the following color fine adjustment.

3.1.2 Color Fine Adjustment

After the inverse log-Gabor transformation, the fused intensity of each high and low activity regions ($I_{f,k}$) is compared with the I-component of the original PET image ($I_{p,k}$, where $k = h, l$). If $I_{f,k}$ is less than $I_{p,k}$ or is greater than $I_{p,k}$ by a large amount, the current pixel location must have MRI structural information. In such case, we leave $I_{f,k}$ unchanged so that the MRI structural information would be preserved after inverse IHS transformation. On the other hand, when $I_{f,k}$ is only greater than $I_{p,k}$ by a small amount, the current location does not have significant MRI structural information; however, the color at this location will become lighter after inverse IHS transformation, as a bigger I-component will lead to bigger R, G, and B, which in turn lead to lighter colors. Thus, we can adjust the fused $I_{f,k}$ back to the value of $I_{p,k}$ so that the I-component after inverse IHS transformation will be identical to PET image. The adjustment equation is as follows.

$$I_{f,k}(x,y) = \begin{cases} I_{p,k}(x,y) & \text{if } I_{f,k}(x,y) > I_{p,k}(x,y) \text{ and } I_{f,k}(x,y) - I_{p,k}(x,y) \leq T \\ I_{f,k}(x,y) & \text{otherwise} \end{cases}$$

where T is the threshold that determines whether the difference between the fused intensity and the original I-component is in fact caused from the MRI structural information at current pixel location. After trial and error, we found that $T = I_{p,k}$ can result in visually optimal fused image.

3.2 Anatomical Structure Enforcement for the Fused Image by IHSLG+

Anatomical information is particularly rich in regions containing GM and CSF pixels in MRI images. Thus, we need to locate the regions containing GM and CSF pixels in the fused image first then add back the missing anatomic information into these regions.

3.2.1 Allocation of GM and CSF

Observing MRI images, we found that GM and CSF pixels are visually darker than WM pixels. Thus, intensity can be used to differentiate GM and CSF pixels from WM pixels. Our GM and CSF pixel allocation procedure is as follows.

1. Obtain the histogram of both high and low activity regions in the MRI image, respectively.
2. Calculate $p\%$ number of pixels N_{pk} (k represents either high or low activity region) in each of high and low activity region.
3. From each histogram, locate the smallest intensity L_k that the total number of pixels whose intensity is less than L_k is greater than N_{pk} . Set L_k as the threshold for separating GM and CSF pixels from WM pixels.
4. Apply the threshold L_k to each activity region in the MRI image and classify each pixel to either GM/CSF or WM.

3.2.2 Anatomical Structure Enforcement

After allocating all the GM and CSF pixels, we adjust the fused intensity after color adjustment ($I_{f,k}$) of each GM or CSF pixel in the regions of high and low activity using

$$I_{f,k}(x, y) = I_{f,k}(x, y) - (I_{f,k}(x, y) - I_{m,k}(x, y)) \times w$$

$$(x, y) \in Q$$

where Q represents the set of GM and CSF pixels, and $w = 0 \sim 1$ is the weight for proper intensity adjustment. When $w = 0$, no adjustment is required, and when $w = 1$, the fused intensity of this pixel will be replaced by the intensity of the pixel at this location in the MRI image. Through different w , various amount of structural information is added to result in different fusion image.

4. EXPERIMENTAL RESULTS AND ANALYSIS

The test data consist of three sets of brain images downloaded from the web site (<http://www.med.harvard.edu/AANLIB/home.html>), where a color PET image and a high resolution MRI image are in each set. The images in Dataset -1, -2, and -3 are normal axial, normal coronal, and Alzheimer's disease images, respectively. All three sets of images are fused firstly by our proposed fusion method IHSLG+, and then by IHSLG+ with color adjustment and anatomical structure enforcement, IHSLG++.

4.1 Performance Assessment

The assessment of fusion image quality can either be subjective or objective. A subjective assessment is best performed by brain doctors to check if all valuable information is contained in a fused image visually. On the other hand, an objective assessment is based on various statistics of the fused image. In this study, we use the following metrics for objective assessment.

4.1.1 Spectral Discrepancy (SD)

The spectral discrepancy measures the spectral quality of a fused image by [3]

$$D_k = \frac{1}{P \cdot Q} \sum_{x=1}^P \sum_{y=1}^Q |f_k(x, y) - f_k'(x, y)| \quad k = R, G, B$$

$$D = \frac{D_R + D_G + D_B}{3}$$

where $f_k(x, y)$ and $f_k'(x, y)$ are the pixel values of the fused and original PET images at position (x, y) ,

respectively, $P = 256$, and $Q = 256$. A tiny discrepancy means an acceptable fusion result.

4.1.2 Overall Performance (OP)

Overall performance is an assessment that integrates both spectral discrepancy and average gradient. The smaller the OP is, the better quality of the image is. OP can be computed by [1, 7]

$$OP = \frac{\sum DP_b - AG_b}{3}, \quad b = R, G, B$$

4.1.3 Mutual Information (MI)

Mutual information (MI) [1, 7] is another common measure to assess the similarity between a fused image and each of its source images. It is a fundamental concept of information theory to compute the statistical dependence between two random variables. Given two random variables, A and B , with marginal probability distributions, $p_A(a)$ and $p_B(b)$, and joint probability distribution $p_{AB}(a, b)$, MI can be calculated by

$$MI(A, B) = \sum_{a,b} p_{AB}(a,b) \log \frac{p_{AB}(a,b)}{p_A(a)p_B(b)}$$

If we represent the original multi-spectral PET image, and the fused image F as two random variables, then $MI(F, PET)$ can be used to estimate how much information of the PET image is contained in F . Similarly, $MI(F, MRI)$ can be used to estimate how much information of the MRI image is contained in F , when we consider MRI as another random variable. The computation equations are:

$$\alpha_1 = MI(F_R, PET_R), \quad \alpha_2 = MI(F_G, PET_G), \quad \alpha_3 = MI(F_B, PET_B)$$

$$\alpha = \frac{\alpha_1 + \alpha_2 + \alpha_3}{3}$$

$$\beta_1 = MI(F_R, MRI), \quad \beta_2 = MI(F_G, MRI), \quad \beta_3 = MI(F_B, MRI)$$

$$\beta = \frac{\beta_1 + \beta_2 + \beta_3}{3}$$

where PET_R , PET_G , and PET_B are the respective R, G, B band of the image PET. The overall MI between the fused image F and PET and MRI can be computed by:

$$MI_F = \frac{\alpha + \beta}{2}$$

4.2 Experimental Results and Comparisons

Dataset-I (Normal axial brain image): The fused images of PET in Fig.4(a) and MRI in Fig. 4(b) by IHS&RIM, IHS&LG+, and IHS&LG++ with $p=10\%$, $w=0.7$ are in Figs. 4 (c)-(e), respectively. Comparing the fused images to the original PET and MRI, we found that (i) the color distortion in 4(c) is visible, particularly in the blue low activity region; (ii) the color distortion in 4(d) is less than that in 4(c), but the color of the red block in the left upper half is still visibly different than PET; (iii) the color in 4(e) has the least color distortion and the structural information of MRI is well preserved as well.

Dataset-II (Normal coronal brain image): The fused images of PET in Fig.5(a) and MRI in Fig. 5(b) by IHS&RIM, IHS&LG+, and IHS&LG++ with $p=10\%$, $w=0.7$ are in Figs. 5 (c)-(e), respectively. Comparing the fused results to the original PET and MRI, we found that both the colors and the structural information in all three images are close to the original PET and MRI images, respectively. However, looking more carefully, we can find that (i) the color of the two blue areas at the bottom of 5(c) appears lighter and the size appears slightly bigger; (ii) the structural information in the upper region of 5(d) is a little weaker; (iii) IHS&LG++ preserves both the color of PET image and the structural information of MRI the best.

Dataset-III (Alzheimer's disease images): The fused images of PET in Fig.6(a) and MRI in Fig. 6(b) by IHS&RIM, IHS&LG+, and IHS&LG++ with $p=15\%$, $w=0.6$ are in Figs. 6 (c)-(e), respectively. Similar to the results of normal coronal brain images, both the colors and the structural information in all three images are close to the original PET and MRI images, respectively. Nevertheless, slight differences among them can still be found when observed carefully: (i) the structural information in the upper right of 6(c) is not all clear enough; (ii) the color in 6(d) is closer to PET than 6(c), though the structural information is not as clear; (iii) the structural information in the upper half region of 6(e) is clearer than that in both 6(c) and 6(d), while the

color is as close to PET as in 6(d).

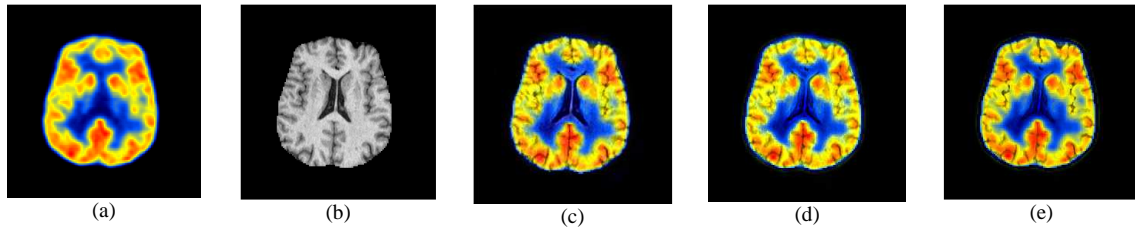


Figure 4. Comparison of the fused normal axial brain images. (a) PET image (b) MRI image (c) IHS&RIM (d) IHS&LG+ (e) IHS&LG++ with $p=0.1, w=0.7$

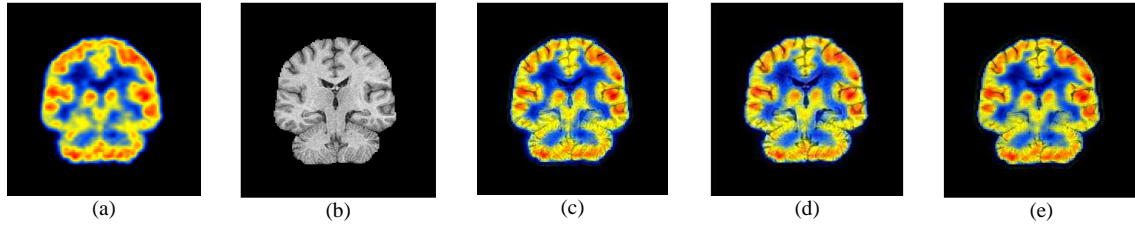


Figure 5. Comparison of the fused normal coronal brain images. (a) PET image (b) MRI image (c) IHS&RIM (d) IHS&LG+ (e) IHS&LG++ with $p=0.1, w=0.7$

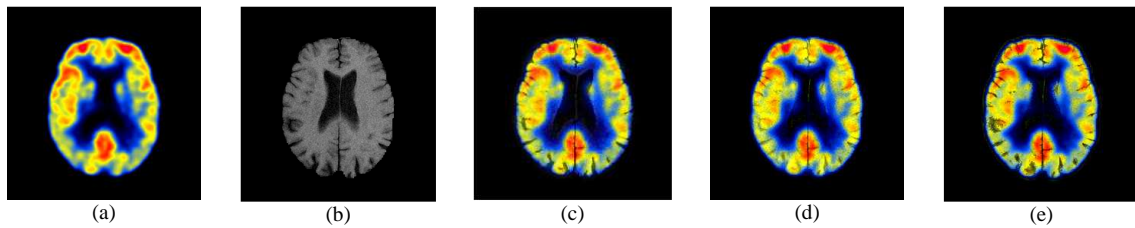


Figure 6. Comparison of the fused Alzheimer's disease brain images. (a) PET image (b) MRI image (c) IHS&RIM (d) IHS&LG+ (e) IHS&LG++ with $p=0.15, w=0.6$

Tables 1-3 list the data of spectral discrepancy, overall performance, and mutual information between the images fused by IHS&RIM, IHS&LG+, IHS&LG++, respectively, and the source multi-spectral image. From Table 1, it is clear that all three images fused by IHS&LG+ have less spectral discrepancies than images fused by IHS&RIM, and all three images fused by IHS&LG++ have the least spectral discrepancies. Similarly, Table 3 shows that all three images fused by IHS&LG+ have smaller overall performance values than images fused by IHS&RIM, and all three images fused by IHS&LG++ have the least overall performance values. Table 2 shows that all three images fused by IHS&LG++ have the largest mutual information values. Images 2 and 3 fused by IHS&LG+ have larger mutual information values than the images fused by IHS&RIM, but the MI of image 1 fused by IHS&LG+ is slightly smaller than that of image 1 fused by IHS&RIM.

Table 1. The spectral discrepancies between the fused image and the PET image

Fusion method	Dataset 1	Dataset 2	Dataset 3
IHS+RIM	7.7061	7.9031	6.0118
IHSLG+	6.5838	7.7461	5.7270
IHSLG++($p=0.05, w=0.7$)	4.6002	4.7612	4.5666
IHSLG++($p=0.10, w=0.7$)	4.7345	4.9118	4.9916
IHSLG++($p=0.15, w=0.6$)	4.6744	4.8017	5.1621

Table 2. Performance comparison based on overall performance

Fusion method	Dataset 1	Dataset 2	Dataset 3
IHS+RIM	2.3457	1.6104	0.9765
IHSLG+	1.2594	1.6585	0.8580
IHSLG++($p=0.05, w=0.7$)	0.3968	0.8597	0.5860
IHSLG++($p=0.10, w=0.7$)	0.3223	0.8100	0.5823
IHSLG++($p=0.15, w=0.6$)	0.2594	0.7787	0.7086

Table 3. Performance comparison based on mutual information

Fusion method	Dataset 1	Dataset 2	Dataset 3
IHS+RIM	0.6541	0.6551	0.623
IHSLG+	0.6484	0.7517	1.1188
IHSLG++($p=0.05, w=0.7$)	0.8817	0.9426	1.2188
IHSLG++($p=0.10, w=0.7$)	0.8828	0.9413	1.2141
IHSLG++($p=0.15, w=0.6$)	0.8799	0.9340	1.2027

5. CONCLUSION

In this paper, we presented two methods for PET and MRI images fusion based on IHS and log-Gabor wavelet. In the first method IHS&LG+, both PET and MRT images are decomposed into high and low activity regions. The intensity channels of high and low activity region images are then decomposed to high and low frequency coefficients using log-Gabor wavelet transform with different scale factors for high and low activity regions, respectively. The high frequency coefficients of each activity region in the MRI image and the low frequency coefficients of each activity region in the PET image are inverse log-Gabor transformed to result in a fused intensity. Finally the fused intensity along with the hue and saturation components of the PET image is inverse IHS transformed to result in a fused image. In the second method IHS&LG++, adjustments to the fused intensity at various pixels is applied for further recuing color distortion and enforcing anatomical structure. The experimental results in normal axial, normal coronal, and Alzheimer's disease brain images demonstrate that all three images fused by IHS&LG+ are with less color distortion and about the same structural information as the images fused by IHS&RIM, and all three images fused by IHS&LG++ are with both color and anatomical structural information closest to PET and MRI images both visually and quantitatively.

REFERENCE

- [1] Chen, Y., Blum, R.S., 2005. Experimental Tests of Image Fusion for Night Vision. *International Conference on Information Fusion (7th)*, pp.491-498.
- [2] Chibani, Y. Houacine, A., 2003. Redundant versus orthogonal wavelet decomposition for multisensor image fusion. *In Pattern Recognition*, Vol.36, pp.879-887.
- [3] Daneshvar, S, Ghassemian, H, 2010. MRI and PET image fusion by combining IHS and retina-inspired models. *In Information Fusion*, Vol.11, pp.114-123.
- [4] Fischer, S. et al, 2007. Self-Invertible 2D Log-Gabor Wavelets. *In International Journal of Computer Vision*, Vol.75, No.2, pp.231-246.
- [5] Hu and Li, 2011. The multiscale directional bilateral filter and its application to multisensor image fusion. *In Information Fusion*. In Press.
- [6] Pajares, G., Jesus Manuel de la Cruz, 2004. A wavelet-based images fusion tutorial. *In Pattern Recognition*, Vol.37, pp.1855-1872.
- [7] Qu, G. et al, 2002. Information measure for performance of image fusion. *In ELECTRONICS LETTERS*, Vol.38, No.7.
- [8] Redondo, R. et al, 2009. Multifocus image fusion using the log-Gabor transform and a multisize Windows technique. *In Information Fusion 10*, Vol. 2, pp 163-171.
- [9] Tu T.M. et al, 2001 .A new look at IHS-like image fusion methods. *In Information Fusion*, Vol.11, pp.177-186.
- [10] Wang, A.et al, 2006. The application of wavelet transform to multimodality medical image fusion, *Proceedings of IEEE International Conference on Networking, Sensing and Control*, pp. 270-274.
- [11] Zhang, H. et al, 2007. A novel wavelet medical image fusion method. *International Conference on Multimedia and Ubiquitous Engineering*, pp.548-553.

SUPER-RESOLUTION IMAGING: THE USE CASE OF OPTICAL ASTRONOMY

Thilo Bauer

*Bluewater Multimedia Concepts
Heimerzheimer Str. 2, D-53332 Bornheim, Germany*

ABSTRACT

Super-resolution is well known to increase the resolution of images below the pixel barrier. The work outlines the use case of super-resolution in astronomy science. The current state-of-the-art of high-resolution imaging in astronomy is reviewed, which leads to the problem of super-resolved images. Fundamental properties of astronomical images are demonstrated to define requirements. Astronomical images suffer from low illumination and noise. The range of spatial resolution is found from long-exposure image limited due to atmospheric blur, and a scale of perfect sampling adapted to diffraction-limited observations. Astronomical imaging is a trade-off between resolution and signal-to-noise ratio (S/N). The success of super-resolution is interfered by the quality of data, selection of the method, and a proper definition of the image formation and image reduction pipeline. This work provides a precise definition of the ill-posed and ill-described problem of super-resolution imaging in astronomy. A new method of automatic image registration is presented, which is especially designed for astronomical imagery. Based on the exploration of real astronomical images, the new method provides an accuracy in the order of 0.05 of the pixel dimension. Future milestones are found as the selection of an appropriate method of image deconvolution. With astronomical wide-field imaging, certain optical effects introduce varying point spread and image distortion. These will define the requirements for an appropriate solution to image deconvolution and to complete the task of super-resolution in astronomy.

KEYWORDS

Super-resolution, image reconstruction, astronomy.

1. INTRODUCTION

Astronomy science has a long tradition with roots found in most cultures on this planet. Astronomy enabled agriculture, as it introduced the time to measure and predict the seasons of the year. The outcome of the astronomy still builds up a fundamental key for human civilization. One example is the definition of a precise reference coordinate system which enables modern navigation of vehicles on ground, ships, airplanes and spacecrafts. Astronomy also asks questions about the origins and processes of the universe. Objects of interest are stars, star clusters, galaxies, clusters of galaxies and the interstellar matter. The more distant these objects are, the more important it seems to obtain better resolved images from the observations.

Digital image processing and analysis started in the 1970ies based on electronic imaging of radio signals and digitized photographic plates (see section 2). The detection of light is different from the detection of a radio signal. A major reason is, that the phase of the electro-magnetic wave cannot be recorded with optical imagers available. Therefore, only intensity distributions are recorded by optical sensors. A new revolution of optical imaging has begun in the 1980ies with the advent of silicon image detectors. *Mackay (1986)* described the use of charge coupled devices (CCD) in the field of astronomy. *McLean (2008)* presents the current work on imaging detectors which is still large field of interest.

With the introduction of digital imaging, the improvement of image resolution is probably one of the most important requirements to astronomical instrumentation and image analysis. An obvious reason is the inability of a ground-based telescope to perform its theoretically possible resolution. If super-resolution methods ever could be applied successfully to astronomy, there might be any expectation to find an efficient and inexpensive method of data reduction which may support or supersede current techniques of high-resolution imaging. The goal of this work is to obtain the requirements from the review, describe the

preferred method of super-resolution and present a new solution to the first milestone: A reliable task of image registration is presented, which achieves high accuracy to evaluate the image motion between images.

2. ANALYSIS

2.1 High-Resolution Imaging in Astronomy

Speckle interferometry was presented by *Labeyrie (1970)* to achieve diffraction limited resolution with a ground-based telescope and a Fourier analysis of the images. *Welter & Worden (1980)* applied this technique and presented the measures of diameters of nearby stars. The image of a star usually is taken as a prominent example of the image of an unresolvable point light source. Hence, the result of *Welter & Worden (1980)* is notable. The measured diameters of certain stars are found in the order of the diffraction limit of a large telescope. Speckle interferometry will improve the resolution of an astronomical image up the optical diffraction limit. However, the technique is very limited, because short exposure times of a fraction of a second are needed. This is sufficient for the observation of a bright star, but will not match the needs to obtain a photo of a faint galaxy (see figures 1, 2). Application of several high-resolution techniques began to awake the interest of astronomers. New methods are researched, applied and improved in a competitive manner to find out which method performs best. The CLEAN algorithm and maximum entropy method have been experimentally applied to astronomical images to evaluate the performance of these methods (*Högbom 1974, Nityananda & Narayan, 1982, Sanromà & Estalella 1984*). *Starck et al. (2002)* provided a review on performance of image deconvolution techniques modified by additional wavelet analysis. Further methods encountered with this review are the Richardson-Lucy method and a technique called the pixon method. Newer techniques are demonstrated to outperform and supersede CLEAN and maximum entropy method. The methods are well suited to high-resolution imaging at faint illumination level.

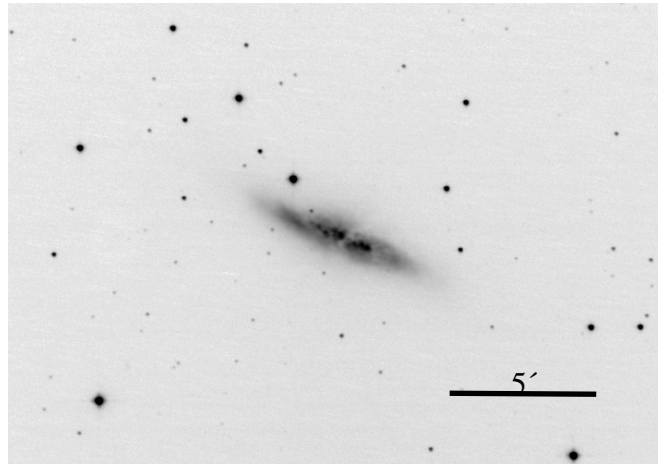


Figure 1. Image of the galaxy M82 taken by the author with a 20 cm telescope. A modified digital single-lens reflex camera provided a total exposure time of 56 minutes. Crop and magnification of the image demonstrates the small scale of the target. Only 8% of the whole field of view are shown. Atmospheric point spread limits image resolution. Use of the Hubble telescope increases resolution by a factor of 12, but required an investment of more than one Billion \$US until the day of writing. Super-resolution methods also help to improve the resolution of the Hubble telescope.

From our neighboring galaxies some very bright stars have been considered as super-massive stars. One reason for thinking of a super-massive star is given by the fact, that a very luminous object will appear like an unresolved point light source due to its huge distance. An example is given with the star HD 32228 (also referred as R64) which has been successfully resolved as a dense star cluster by *Schertl et al. (1995)* and *Bauer et al. (1996)*. From the latter work it seems, larger exposure time and less images enabled a first successful astronomical examination of the single stars of the cluster. One might ask, how to optimize a given observation time frame in the sense of dividing total time into portions of exposure time to obtain

optimal resolution from the number of images obtained. *Smith & Gallagher (2001)* discuss properties of a peculiar, unresolved star cluster in the distant galaxy Messier 82. Illustrations and measures provide insights in how far ground-based observations differ from observations of a space telescope. The star clusters observed remained unresolved due to the large distance of the galaxy where the cluster is embedded. Figure 1 demonstrates a typical image scale of the galaxy M82 seen by a ground-based telescope. *Jacoby (1989)* and *Ciardullo et al. (1989)* estimated distances of galaxies in the neighborhood of the Milky Way. The estimation is based on the measure of brightness of certain gaseous nebulae contained in these galaxies. Because distances of the galaxies are huge, these objects appear like point sources. It is hard to distinguish the nature of point light sources observed. Statistical assumptions about the emitted light is taken as a hint whether the identified sources represent stars, clusters or nebulae. Better resolution is clearly a requirement. *Gilliland and Dupree (1996)* presented images of the surface of a star observed with the Hubble space telescope. The result from a pixel interpolation and super-sampling is discussed as the features observed from the surface of the star. The method applied is a typical approach to statistical super-resolution (see section 3). This enumeration of examples will not claim for completeness. The field of astronomical imaging is experimental and seems to change with every new telescope built and camera used for the observations.

2.2 Properties of Astronomical Images

At this point one shall consider the typical properties of astronomical images. Except images of the bright surfaces of the sun, moon and bright planets like Venus and Jupiter, the vast majority of astronomical images is obtained in the dark night sky. *Geyer (2010)* presented considerations and constraints between signal-to-noise ratio and the limiting magnitude of a point light source detected in an astronomical image. The term magnitude refers to a logarithmic measure of a ratio of luminosity between two measured objects (a.k.a. Pogson's law). The limiting magnitude describes the apparent luminosity of the faintest intensity detected in an image. Obviously, the faint light signal plays an important role in the astronomy. *Mackay (1986)*, *Berry & Burnell (2006)* and *McLean (2008)* described further properties of common detectors leading to a standard process established to the image calibration. Noise parameters are evaluated and a linearization of the image intensities is done carefully. The process chain also includes subtraction of bias and dark signal. The flatfield task compensates for the effects of varying pixel sensitivity and vignetting. Image calibration is a requirement to obtain results which shall be photometrically examined at high accuracy and even with the faintest signal available in the raw images.

Back to the viewpoint of a software engineer, one shall assume image content in the astronomy is completely different from conventional photography and artwork. Rakishly spoken, astronomical images taken in the night sky contain a handful of photo events detected, which are mixed up with certain amount of detector noise of nearly the same count rate. Anything which seems interesting appears small and almost close to noise level of the astronomical image itself. The impact of noise is omnipresent and different from conventional photography and artwork.

2.3 Limitations to resolution

There are certain limiting factors of resolution at different scales. These are defined by properties of the telescope optics and detector. Resolution is also given by environmental influence at the observing site. Finally, there are limits introduced by the method of observation and processing the data.

The limit to clearly separate two images of point light sources is approximated by the equation $\alpha = 1.22\lambda/D$ with the angular separation α , the wavelength λ and the aperture diameter D of the optics. This is also referred as the classical Rayleigh criterion of the optical diffraction limit. It is seen as a fundamental limit to resolution and caused by the physics of light (*Born & Wolf, 1953*). *Den Dekker & Van den Bos (1997)* presented a comprehensive review on resolution from several viewpoints, including optics and information theory. *Den Dekker & Van den Bos (1997)* found the term „resolution limit“ an arbitrary limit. One shall assume any resolution beyond any (arbitrary) limit, like the Rayleigh criterion. Resolution will depend on certain impacts, like design of optics, detector, and noise properties found. As resolution is found arbitrary, this will mean the achievable resolution is unlimited. However, the exercise is left open to estimate the effort needed to obtain a certain resolution. Because image quality also depends on noise characteristics, resolution is usually a trade-off between signal quality and spatial resolution. Optimal detector performance is obtained,

if the Whittaker-Shannon sampling theorem is fulfilled. Hence, resolution undergoes limitations introduced by the design and adaptation of the telescope optics and camera used for the observations. Finally, a limit to resolution will also be introduced by the method of data reduction.

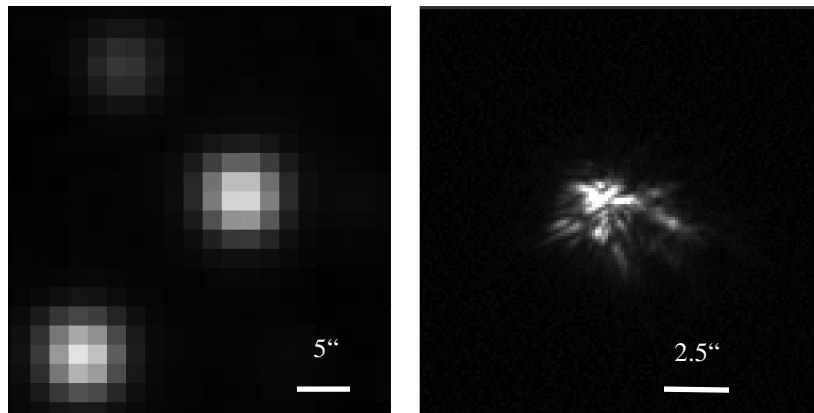


Figure 2. Resolution at different scales: Detail of a long exposure of a star cluster dominated by large pixel dimension, with sampling adapted to the dimension of seeing (left). This is compared to a short exposure of a bright binary star observed with speckle interferometry at the Hoher List 1 m Cassegrain telescope with focal length extended to 30 m (right, taken in the year 1994). Pixel scale is given in arcseconds.

3. SUPER-RESOLUTION AND THE ASTRONOMY

Until the day of writing this review, super-resolution cannot be seen as a term referring to a well-defined expression, precisely describing a certain technique or referring to a certain limit of resolution. This is especially true with the review of the literature available in the interdisciplinary fields of information technology and astronomy. *Baranov et al. (1996)* and *Farsiu et al. (2004)* described digital super-resolution algorithms and also referred to astronomical applications. *Cristóbal, et al. (2008)* presented a comprehensive survey of current super-resolution techniques. They also mentioned applications to astronomy. *Willet & Nowak (2004)* presented a wavelet based approach to super-resolution in astronomy. *Willet & Nowak (2004)* applied their method to simulated images, which more likely seem to resemble satellite observations of the surface of the earth, but do not match deep sky imagery found in the astronomical literature (see section 2). It shall be outside the focus of this article to count all interdisciplinary work reviewed until the writing of this paper. From the interdisciplinary side no work referring to astronomical applications showed significance, whether the proposed methods of super-resolution could be applied to astronomical images. Samples provided with the literature more likely resemble conventional photography and artwork, which is well exposed and does not contain too much noise.

Lucy (1992a, 1992b) and *Semintilli et al. (1993)* presented considerations to the dependency between the signal-to-noise ratio and super-resolution. Here super-resolution means to break the optical diffraction limit. *Hook & Fruchter (2000)* presented a method to obtain sub-pixel resolution by the drizzle method with an application to astronomy. The drizzle tool is a command-line oriented software, designed to perform resampling and image co-adding with certain parameters, like image motion, rotation and predefined distortion functions. The tool will not evaluate these parameters from the data and will not provide better physical resolution in the sense of any image enhancement, like a deconvolution, however.

The use of the term super-resolution is ambiguous and sometimes misleading if comparing the viewpoints of physics and information technology. Opticians, astronomers or physicists in general tend to speak about super-resolution only, if the resolution achieved is better than diffraction limit of the optics. Following the common trend in the literature of information technology, the term super-resolution more likely shall reflect a digital method to obtain any resolution at a finer sampling grid. A certain danger is indicated to compare different methods and leading to possible misinterpretation of the results, like the comparison of results obtained from statistical super-resolution and single image super-resolution techniques presented by

Puschmann & Kneer (2005). Hence, the term super-resolution shall be substantiated to clearly describe the goal and method.

4. DESCRIPTION OF THE PROBLEM

In astronomy a super-resolved image shall be reconstructed without any a priori knowledge about the source. The source itself defines the objective which shall be examined. Hence, one shall assume statistical super-resolution is the preferred way to proceed.

Shall M , N be the dimensions of the measured low resolution images in width and height. These requirements have to be fulfilled by methods of *statistical super-resolution*:

1. **Enhancement of sampling:** *Super-resolution shall provide a high-resolution image representing a larger number of pixels with any multiple $K=[2,3,4,\dots]$ of the original pixel amount of the low-resolution input images in both dimensions M , N .*

2. **Enhancement of angular (spatial) image resolution:** *Super-resolution shall incorporate certain image sharpening (like deconvolution) which reflects the gain in the number of pixels.*

3. **Fidelity of photometry:** *Intensities detected shall be preserved with the proposed method of super-resolution.*

These requirements reflect the situation of having said „information found and confirmed between pixels at improved point spread and intensities preserved“. Any result obtained with a (small) telescope and reaching super-resolution performance, therefore, shall present better sampled image with point spread of better angular resolution, and compared to original seeing conditions found with the observations. If this is found and structures can be confirmed with independent observations of any telescope with better seeing performance or compared to observations with a space telescope not suffering from atmospheric image degradation, super-resolution is assumed to be successfully demonstrated. A method presenting a larger amount of pixel, smoother images, but not showing reduced point spread (better angular resolution) shall not be taken as a super-resolution method. Breaking the diffraction limit of optics could be a secondary outcome, but is not necessarily a requirement. Thus, super-resolution beyond the diffraction limit is a special case of the problem. The effort to break the Rayleigh criterion might be worth to be evaluated. Fidelity of intensities detected shall be preserved with the super-resolution method chosen. The method is expected not only to demonstrate a better spatial resolution, but also shall allow photometric evaluation of the source.

5. A NOVEL METHOD OF IMAGE REGISTRATION

5.1 Development of the Idea

Statistical super-resolution is the process of combining a sequence of low-resolution images in order to obtain a high resolution image. The task of statistical super-resolution imaging can be divided into a two stage process consisting of (1) the image registration of a set of low resolution images to be put into a larger pixel grid and (2) sharpening of the resulting oversampled image. This work shall consider the first step to establish a robust task of motion estimate for the image registration. *Cristóbal, et al. (2008)* presented a review of super-resolution methods. From the work of *Zitová & Flusser (2003)* and *Borman (2004)* the task of motion estimation is found crucial to the success of super-resolution methods. *Robinson & Milanfar (2004)* discussed fundamental performance limits to the image registration based on considerations of the several features found in images, like gradients or edges. Such features more likely will resemble conventional artwork and photography. With typical images demonstrated in this work, one hardly will find similar features in astronomical imagery. To collect the many images needed to obtain a single super-resolved image, the total exposure time has to be divided into smaller portions to obtain short exposures. In most cases of observation of galaxies and nebulae, short exposures will not yield a reliable image of the object itself. Therefore, a robust anchor to measure image motion cannot be obtained from the observed object itself (see figure 3). Hence, it seems more reasonable to obtain a motion estimate from a stellar reference coordinate system. In other words, parameters like image motion, rotation and distortion shall be obtained from the amount of brighter stars found in the image. A fundamental question will be the

determination of the precision and limit of an estimated position of a star. *Guseva (1995)* and *Anderson & King (2000)* reported the possibility to obtain precise position measures with subpixel accuracy. However, systematic errors have been found in the order of a small fraction of a pixel dimension. *Bauer (2009)* presented a review of methods to fit the point spread of stars and to measure stellar positions. *Bauer (2009)* also proposed a new approach to eliminate the systematic errors introduced by the methods of digital image processing. The new method to determine the position of a star now is understood as a new method to determine a motion estimate with high precision.

5.2 Collection of Data

Optimal sampling is derived by considerations on the point spread of a star image. This will depend on the telescope and observation site. Geometric consideration yields the equation $\tan(\beta)=d/f$ with the angle β , focal length f and pixel dimension d (f and d measured in the same unit). With atmospheric seeing angle of approximately 3 arcsec and using an image sensor having a pixel dimension of 10 microns, this yields a focal length of about 1.5 m. The requirement is fulfilled using a mid-range amateur telescope of 20 cm aperture, relative aperture $f/8$. A digital single lens reflex camera or a dedicated CCD are typical detectors (RGB cameras with Bayer pixel pattern yield twice the pixel dimension of the individual color pixels).

From the review it is seen, that certain authors prefer simulated imagery. However, a primary goal is to demonstrate the capabilities and limits of algorithms with real observations. In the year 2007 new astronomical observations have been started by the author of this work. The telescope selected for the observations is a dedicated 20 cm Cassegrain telescope, designed for photographic applications. A respectable amount of images has been collected until today. Some 10.000 images of different celestial objects have been photographed, manually inspected and archived in a digital database containing more than 250 GB of images. The individual image series for the several objects contain between 20 to 300 images of star clusters and nebulae contained in our Milky Way and also images of the very faint and distant galaxies containing unresolved star clusters and nebulae at noise level. A similar amount of calibration data has been collected run to apply the standard image calibration process defined within the astronomy.

5.3 Implementation of the Method

The new method of image registration proposed with this work will perform the following tasks: (1) Take a series of (many) low-resolution images of an astronomical object, (2) determine parameters of relative image motion, rotation and image distortion between the images, (3) over-sample the images and (4) co-add the images to a composite image at a larger pixel grid.

The determination of image motion is divided into the following sub-tasks: (1) The standard image calibration is applied to every single input image. (2) A *search of local maxima* is used to detect the stars within the calibrated image. (3) With every star successfully detected, a fit of a synthetic point spread function is performed to obtain a position with sub-pixel accuracy. A correction of the systematics of a relative deviation function (RDF) is applied in this processing step (*Bauer, 2009*). (4) From a correlation of two consecutive images, the relative image motion, rotation and distortion are determined. The images finally are registered and oversampled with sub-pixel accuracy.

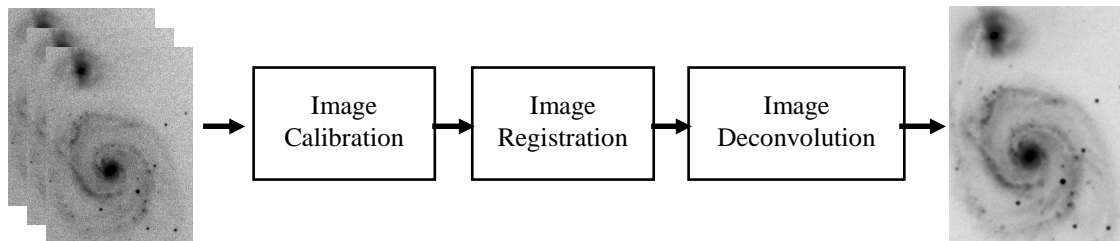


Figure 4. The process chain of statistical super-resolution in astronomy to be completed by future work.

The *search of local maxima* yields a first position estimate of stars detected above a certain noise level. The star detection is done by the computation whether the central pixel of a 3x3 pixel frame represents the maximum intensity of all 9 pixels. If this condition is true, and the maximum intensity found is larger than a

multiple of the standard deviation of the sky background (noise), then it is assumed a star is detected at the position (x,y) of the central pixel. The determination of a multiple of the standard deviation yields control about the probability to safely detect a star. From the *fit of the point spread function* a precise sub-pixel position is obtained. The correction of systematic deviations of measures from the true positions is performed by the *relative deviation function* described earlier (Bauer, 2009). There are several models of point spread discussed within the astronomy and also asymmetric point spread found from the real observations. Within this work, a simple Gaussian function is fitted to the observed point spread to obtain a sub-pixel position of the intensity distribution. Parameters, like full-width at half maximum of the Gaussian, are estimated from the point spread (seeing) found in the image. It shall be mentioned, however, that a wrong estimation of the parameters of the Gaussian function will have an impact to the position accuracy. However, the examination of the systematic deviations are not in the focus of this work and shall be evaluated with future work. It is sufficient for the moment to have found a very stable method of image registration with an accuracy below 1/10 of a pixel dimension.

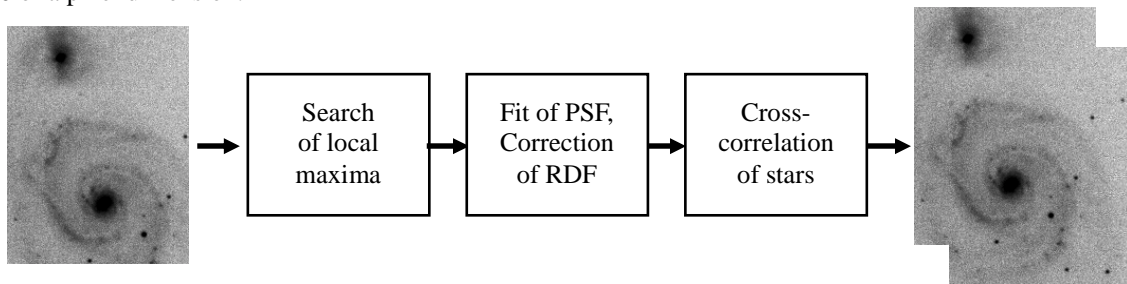


Figure 5. The image registration task proposed by this work. The fit of stellar positions include a correction of systematic errors from the relative deviation function.

5.4 First Results

The author implemented and applied the proposed method to the image sequences collected. Certain sequences or parts of large sequences have been inspected, and images rejected due to impacts of observation errors, like technical problems with the telescope mount, accidentally defocused images or other mistakes, like the authors cat climbing the telescope. As a secondary outcome, the method enabled the complete image gallery of the authors website over years (Bauer, 2011). Therefore it is demonstrated, the method works with every material collected so far. This is a great success and the stability of the approach is demonstrated well as the first fully automatic method of image registration in astronomy with high position accuracy.

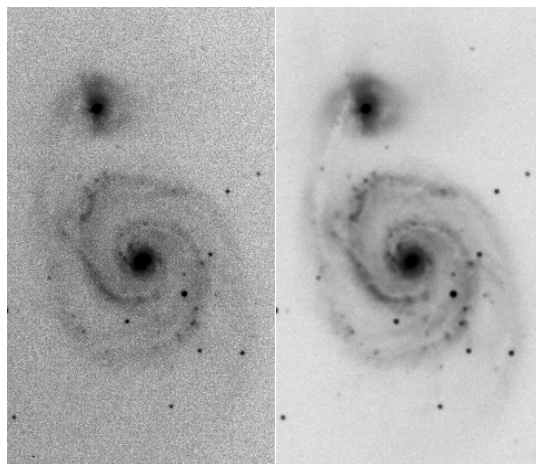


Figure 3. A single raw image of 4 minutes exposure time of the galaxy M51 (left image) is compared to the result of the image registration of 24 input images (right image). The proposed algorithm to determine image motion with sub-pixel accuracy has been applied to the image registration. The image registration is automatic and stable with imagery of such a low illumination and does not show image displacement.

As noted earlier, point spread of a star will vary over the field of view. A real optics also suffers from geometric distortion of the image. This causes an additional impact to super-resolution (*Gilliland and Dupree, 1996*). Several systematic deviations can be found which affect the process of image registration. This will require a more sophisticated method of correction for image distortion and rotation, if super-resolution requirements shall be fulfilled with a large field of view required at the same time. With a small relative image motion, errors are found very small. The following drawings shall give insights into the accuracy and constraints between location and deviation of stars detected from the observations. The accuracy of the method depends on the brightness of the stars. A first limit can be found at a few 1/100 of the pixel dimension (figure 5). Starting from a certain illumination of a star, the accuracy will not increase towards brighter stars. This limit is caused by the method of fitting a point spread into digital data which introduces systematic deviations (*Author*, 2009*). It can be improved by a relative deviation function modeled to the data. Position deviations also are found dependent on the location of the star detected. Such errors can be made visible, if the position detection is very accurate and a large amount of images is used for the computations. Figure 6 shows the dependency between position deviations and the distance of the star from the center of the image obtained from the same images of a stellar region. Such constraints are expected due to geometry of optics and usually described in the literature about astrometry (the astronomical task to measure positions).

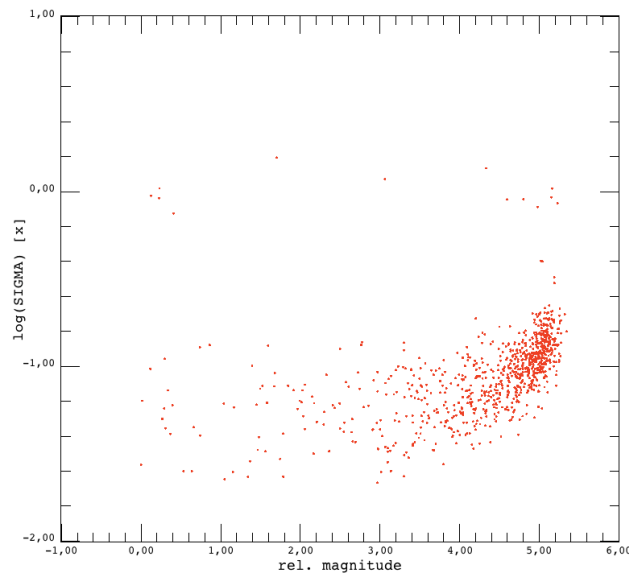


Figure 5. Evaluation of the position accuracy of several star clusters clearly demonstrates the dependency on the signal-to-noise ratio. In this double-logarithmic drawing the evaluation of the position error is shown for 25 images of the star cluster Messier 29. A first lower limit bound is found in the order of 0.07 pixel. With the few brighter stars (logarithm close to zero, left side) the error again tends to increase due to non-linearities. A few points in the center are caused by wrong detection of stars.

Image calibration is needed to avoid non-linearities which may introduce erroneous positions. This influence is seen in figure 5: Bright or saturated star images will yield an increase of the position error. The best accuracy is detected with mean values of star intensities.

A few issues have been found by the experimental use of the large library and shall be mentioned. The method is found to work well especially with imagery of very low intensities detected. This is especially true for the faint galaxies, where the signal-to-noise ratio of intensities of the object is clearly found below the value one, which means no significance to have a signal detected. However there has been a limit found, where the method will produce certain problems with dense star clusters, like globular clusters. This is an impact caused by the noise. Globular clusters are very dense star clusters, with many stars concentrated and partially overlapping in a small portion of the image. Due to noise the faint stars cannot always be detected at the same places. Therefore, some stars disappear, while others are detected at different positions. This may lead to misalignment detected between images, and is caused by the large number of stars found scattered.

However, excluding stars with certain small distances from the cross-correlation identification, will avoid such misalignment.

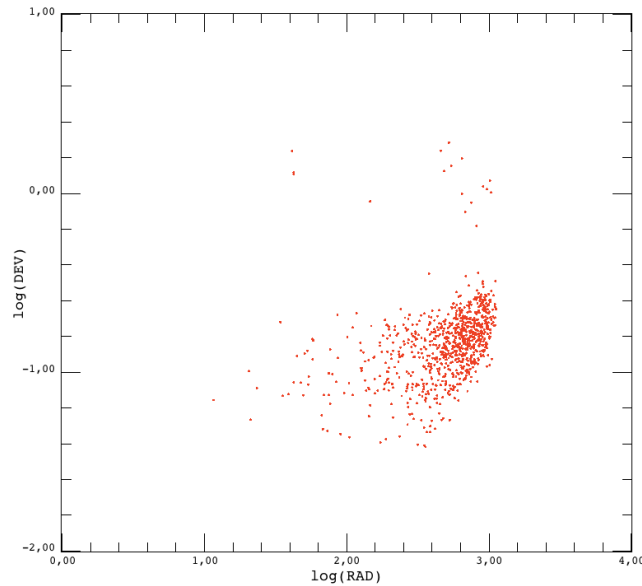


Figure 6. The position deviation (DEV) as a function of the radial distance (RAD) to the image center. A systematic deviation towards the outer image regions shows the dependency on the optical field distortion. This evaluation was taken from the same positions of stars detected from the images shown in figure 5. The images have a total size of 1944x1296 pixel at a field of view of about one degree.

6. CRITICAL EVALUATION

Astronomy defines strong requirements to any super-resolution method. Astronomical imaging suffers from very low intensities measured and a serious impact of noise. The development of a novel and reliable method to the image registration is presented with this paper. The proposed method is a very intuitive and basic approach. On the other hand, and over the last years, it is shown as a very stable method of automatic image registration. The method was enabled by the examination of methods which already have been applied to astronomy, but (originally) meant for a completely different use. Not any method of image registration reviewed has been shown to provide a similar performance expressed in units of a fraction of a pixel dimension. This is especially true, if considering the noise found in astronomical images. One requirement shall be fulfilled, however: There must be a certain amount of stars detectable. From the current experience with the method, the author found at least 50 stars with any observation to be taken for a position estimate to image motion. In such regions of the sky, galaxies are observed apart from the disk of the Milky Way. These are portions of the sky where the number of stars used for the image registration drops from at least thousand stars to only a few. With an angular field of view of about one degree, and until the day of writing, the author did not found an area on the sky where the method could not be applied. However, this could mean some restrictions to the usability with a very large telescope providing a smaller field of view. Several issues and limitations have been predicted and confirmed by the data analysis. These are limitations caused by the geometry of optics: optical distortion and varying point spread within a large field of view. A few issues could be solved in the meanwhile, like the problem to detect image motion from crowded star fields. The exercise to complete the process chain of super-resolution shall be a future task. From this review a few methods shall be considered proper to image enhancement. These include the wavelet modified Richardson-Lucy method used by *Bauer et al. (1996)* and several methods reviewed and demonstrated by *Starck et al. (2002)*. However, these algorithms are linear methods of image deconvolution. Taking into account the effects of varying point spread and optical distortion, such methods shall not be assumed to work in any case without further modification. At least one shall expect any method of super-resolution to produce artifacts

from residual errors introduced by the optics. Therefore, super-resolution in astronomy, which also requires a large field of view, will be a future task to convert a linear image deconvolution into a reasonable non-linear approach to solve remaining issues like spatially variant point spread. Therefore the image formation process shall be formulated very carefully. Certainly, such an approach is quite different from most linear methods to conventional super-resolution imaging. However, one shall expect similar problems to occur with any real optics. Hence, the basic ideas might be transferred to „normal“ applications of super-resolution.

7. CONCLUSION

This work presented a review on super-resolution in astronomy. Requirements are strong and the problems are identified with the low illumination of detectors and the impact of noise. Further issues extracted are spatially variant point spread found with real images and geometric distortion of fields. These issues clearly demand for new approaches to super-resolution compared to the methods described earlier and applied to conventional imagery. Hints are found from the review, that methods of image registration and deconvolution may supplement or supersede the more expensive methods of astronomical imaging including the use of space telescopes. As super-resolution was not found a well-defined term or method, the author presents a precise formulation of the problem. A new method of detection of image motion with high precision was developed based on the requirements. The novel technique is well adapted to typical astronomical image content and was applied to a large amount of real astronomical images. The new method is able to provide measures with a precision in the order of 1/50 of a pixel dimension. It is found as a systematic error which is introduced by the methods of digital imaging. The method shall form the first milestone of an accurate image registration within the statistical process of astronomical super-resolution imaging. The next milestone shall complete the super-resolution process chain. This will be the task of selection and/or development of a proper image deconvolution method to obtain a super-resolved image from an super-sampled high-resolution image. If wide-field imaging and super-resolution shall be requirements at the same time, this will lead to certain non-linear approaches to solve issues like spatially variant point spread and geometric image distortion. Aside from the hints collected to have already broken the optical diffraction limit of resolution, the task to estimate the effort by a simple super-resolution approach still is seen as a challenge. The author suggests the myth to be reality will be demonstrated only, if the obtained resolution clearly demonstrates more than twice the diffraction limited resolution. This must be taken from real observations and confirmed by independent observations in different colors and/or taken by a different, preferably larger telescope. Such a method shall demonstrate a resolution, which is more than six times better than the resolution found with the long-exposure images shown with this work and observed with a telescope of the same aperture.

REFERENCES

- Anderson, J. and King, I. R., 2000. Toward High-Precision Astrometry with WFPC2. I. Deriving an Accurate Point-Spread Function. *The Publications of the Astronomical Society of the Pacific*, Volume 112, Issue 776, pp. 1360-1382
- Bauer, T., 2011, Image gallery, collected between the years 2007 and 2011, <http://www.astroinformatics.de/>
- Bauer, T., 2009. Improving the Accuracy of Position Detection of Point Light Sources on Digital Images, *Proc. of the IADIS Multiconference, CGVCVIP*, Algarve, Portugal, p. 3-15, ISBN: 978-972-8924-84-3
- Bauer, T., 2008. Astroinformatics - A Study about Constraints and Requirements for Next Generation Astronomical Image Processing, *Proc. of the IADIS Multiconference on Computer Science and Information Systems*, Amsterdam, Netherlands, p. 83-90, ISBN 978-972-8924-63-8
- Bauer, T. et al., 1996. The young star cluster R64 in the OB association LH9 resolved with ground-based CCD observations. *Astronomy and Astrophysics*, vol. 305, pp. 135
- Berry, R. and Burnell, J., 2006. *The Handbook of Astronomical Image Processing*. 2nd. english ed., Wilmann-Bell, Inc., Richmond, VA, USA
- Borman, S., 2004. Topics in Multi-frame Superresolution Restoration. Dissertation at the Graduate School of the University of Notre Dame, Indiana, US.
- Born M. and Wolf, E., 1953. *Principles of Optics*. Pergamon Press, Great Britain, 195

- Ciardullo, R. et al., 1989. Planetary nebulae as standard candles. II - The calibration in M31 and its companions. *Astrophysical Journal*, vol. 339, p. 53-69.
- Cristóbal, G. et al. 2008. Superresolution imaging: a survey of current techniques. *Proc. of the SPIE 7074, 70740C-70740C-18*, Advanced Signal Processing Algorithms, Architectures, and Implementations XVIII.
- Den Dekker, AJ and Van den Bos, A., 1997. Resolution: a survey. *Journal of the Optical Society of America A*, vol. 14, no.3, p.547-557.
- Farsiu, S. et al., 2004. Advances and challenges in super-resolution. *International Journal of Imaging Systems and Technology*, vol. 14, issue 2, pp. 47-57.
- Geyer, E. H., 2010. Die Nachweisgrenze astronomischer Objekte bei teleskopischen Beobachtungen. *Jenaer Jahrbuch zur Technik und Industriegeschichte*, Vol. 13, ISBN 978-3-939718-56-7, Germany.
- Gilliland, R. L. and Dupree, A. K., 1996. First Image of the Surface of a Star with the Hubble Space Telescope. *Astrophysical Journal Letters*, vol. 463, p. L29
- Guseva, I.S., 1995. Some Problems of Wide-Field Astrometry with a Short-Focus CCD Astrograph. *IAU Symposium No. 167, New Developments in Array Technology and Applications (1995)*, A. G. D. Philip, K. A. Janes and A. R. Upgren, eds., Kluwer Academic Publishers, Dordrecht. Held in the Hague, the Netherlands, August 23-27, 1994., p.275
- Högbom, J. A., 1974. Aperture Synthesis with a Non-Regular Distribution of Interferometer Baselines. *Astronomy and Astrophysics Supplement*, Vol. 15, p.417.
- Hook, R. N. and Fruchter, A. S., 2000. Dithering, Sampling and Image Reconstruction. *Astronomical Data Analysis Software and Systems IX, ASP Conference Proceedings*, Vol. 216, edited by Nadine Manset, Christian Veillet, and Dennis Crabtree. Astronomical Society of the Pacific, ISBN 1-58381-047-1, 2000., p.521
- Jacoby, G. H., 1989. Planetary nebulae as standard candles. I - Evolutionary models. *Astrophysical Journal*, vol. 339, p. 39-52.
- Labeyrie, A., 1970. Attainment of Diffraction Limited Resolution in Large Telescopes by Fourier Analysing Speckle Patterns in Star Images. *Astronomy and Astrophysics*, Vol. 6, p. 85
- Lucy, L. B., 1992(a). Statistical limits to super-resolution. *Astronomy & Astrophysics*, vol. 261, pp. 706-710.
- Lucy, L. B., 1992(b). Resolution limits for deconvolved images. *Astronomical Journal*, vol. 104, pp. 1260.
- Mackay, C. D., 1986. Charge-coupled devices in astronomy. *Annual review of astronomy and astrophysics*. Volume 24, A87-26730 10-90). Palo Alto, CA, Annual Reviews, Inc., 1986, p. 255-283
- McLean I. S., 2008. *Electronic Imaging in Astronomy: Detectors and Instrumentation*. 2nd ed. Springer, Berlin
- Nityananda, R. and Narayan, R., 1982. Maximum entropy image reconstruction - a practical non-information-theoretic approach. *Journal of Astrophysics and Astronomy*. vol. 3. no. 4, p.419-450.
- Puschmann, K. G. and Kneer, F. 2005. On super-resolution in astronomical imaging. *Astronomy & Astrophysics*, Vol. 436, p. 373-378
- Robinson, D. and Milanfar, P., 2006. Statistical Performance Analysis of Super-Resolution. *IEEE Transactions on Image Processing*, VOL. 15, NO. 6
- Sanroma, M. and Estalella, R., 1984. Hybrid mapping using the maximum entropy method. *Astronomy and Astrophysics*, vol. 133, no. 2., p. 299-306.
- Schertl, D. et al., 1995. Speckle masking observations of R 64, the dense stellar core of the OB association LH9 in the Large Magellanic Cloud. *Astronomy and Astrophysics*, vol. 302, pp. 327.
- Sementilli, P. J. et al., 1993. Analysis of the Limit to Superresolution in Incoherent Imaging. *Journal of the Optical Society of America*, vol. 10, no. 11.
- Smith, L. J. and Gallagher III, J. S., 2001. M82-F: A doomed super star cluster? *Monthly Notices of the Royal Astronomical Society*, vol. 326, pp. 1027-1040.
- Starck, J. L. et al., 2002. Deconvolution in astronomy: A review. *Publications of the Astronomical Society of the Pacific*, vol. 14, no. 800, pp. 1051.
- Welter, G.L. and Worden, S.P., 1980. The angular diameters of supergiant stars from speckle interferometry. *The Astrophysical Journal*, vol. 242, p. 673-683.
- Zitová, B. and Flusser, J., 2003. Image registration methods: a survey. *Image and Vision Computing*, 21, pp. 977-1000

RECOGNITION OF FIRE/SMOKE BASED ON COMPUTER-VISION IMAGE SEGMENTATION

Marie Providence Umugwaneza, Jean Paul Dukuzumuremyi, Bei-Ji Zou,
Lei Wang and Yixiong Liang
*School of Information Science and Engineering, Central South University,
Changsha, 410083, China.2008*

ABSTRACT

Dangers of fire/smoke are recently serious whereas scientific research to fight them seems to be left behind. The newest innovations use cameras and computer algorithms to analyze the visible effects of fire and its motion in their applications. Their approaches present some drawbacks when working in spatial domain. The main difficulty is to identify objects if they do not occur at the expected position.

In this paper, we present a fast and robust algorithm for recognition fire/smoke in a cluttered scene from a pair of cameras. The input images are first segmented according to a pre-determined disparity threshold map and the real-time disparities of the fire. Binary image processing techniques are used to reject noise introduced into the segmented images through low-resolution disparity calculations which consequently can lead to the gain of clearer results. In order to reduce the false alarms, a new segmentation method used in this approach shows that stereo vision segmentation is more accurate than the color-based method based on k-means segmentation for the overall recognition; more specifically for images taken in night. The wavelet and discrete cosine transforms are then used for image feature extraction for a neural network classifier hence the system could generate a warning in case the fire/smoke is recognized.

KEYWORDS

Computer vision, fire recognition, neural network, k-means, wavelet transform, discrete cosine transform.

1. INTRODUCTION

Traditional point-type thermal and smoke detectors that are widely used nowadays practically take charge of a limited area in space. In large rooms and high buildings, it may take a long time for smoke particles and heat to reach a detector. Video-based fire detection (VFD) is a newly developed technique in the last few years, and it can greatly complete the fire detection/recognition requirements in large rooms and high buildings, and even for outdoor environment as shown by George et al. Researchers all over the world have done a lot of work on this new technique. Up to now, most of methods make use of the visual features of fire including color, textures, geometry, flickering and motion.

Many researches used RGB input as a simple and effective procedure for real-time application. To improve the RGB systems, a fuzzy logic enhanced generic color model for fire pixel classification is proposed. The model uses YCbCr color space to separate the luminance from the chrominance more effectively than color spaces such as RGB as studied by Celik et al. An algorithm of early fire image detection/recognition based on discrete fractal brownian incremental random field model is proposed by Chen et al.

It is important to realize that most of fire detection/recognition systems used the heuristic fixed threshold values in their specific methods as did Bo-Ho et al. However, as input images may be changed in general, the heuristic fixed threshold values used in those fire recognition systems might be modified on a case by case basis. So, an automatic fire recognition system without the heuristic fixed threshold values was studied and an automatic method using the statistical color model and the binary background mask was presented.

In addition to motion and color clues used in the literature reviewed above, another approach proposed smoke/fire recognition by analyzing the video in wavelet domain has been treated by Behcet U et al. To improve the solutions of the shortcomings enumerated above, many researchers opted computer vision based

fire/smoke detection/recognition. Some techniques such as combination of color and stereo vision used by George et al, Behcet U et al and Ug̃ur et al have been developed to compensate small or gradual changes in the scene or the lighting, however, they cannot deal with large or sudden changes in the background. Although stereo vision and edge detection proposed by Moulay et al can be used to detect independently moving targets in the presence of camera motion, it is not feasible for no-rigid object extraction since the movements of the fire/smoke are different.

To increase reliability, some systems like the one used by Ug̃ur et al and Nicholas True, integrate multiple cues such as stereo, color, wavelet pattern to detect fire /smoke. However, color is very sensitive to illumination changes; wavelet detection can only identify smoke/fire facing the camera. These systems prove that stereo and wavelet are more reliable and helpful cues than color and wavelet fire detection in general situations. To overcome the shortcomings listed above, we propose stereo vision for fire/smoke recognition and wavelet transform combined with discrete cosine as features extraction plus neural network for fire classification as recent research in neuroscience has shown that object recognition in primates makes use of features of intermediate complexity that are largely invariant to changes in scale, location, and illumination known that many of the neurons respond to color and texture properties in addition to shape by DG Lowe.

Our new fire warning system is equipped with a stereo-based object recognition module that detects foreground objects in real-time based on disparity map segmentation, and its main function is to reduce false fire alarms as it will distinguish fire information from other objects present in the scene. This is achieved through stereo-based segmentation, wavelet transform and discrete cosine for features extraction and neural network-based recognition.

2. DISPARITY BASED SEGMENTATION PROPOSED SYSTEM LAYOUT

A general flowchart of our fire/smoke recognition system is depicted in figure 1. The system input is a pair stereo IR images. The system output is the class label (from 1:8) to which the fire belongs. First the object is extracted using the proposed technique (see section 3). Secondly two image features are computed simultaneously on the segmented object. Thirdly, a neural network classifier (NN) trained offline is used to classify the objects into one of the eight images categories: DWF(Day With Fire), DNF(Day No Fire), NWF(Night Without Fire), NNF(Night No Fire), DWS(Day With Smoke), DNS(Day No Smoke), NWS(Night With Smoke), and NNS(Night No Smoke).

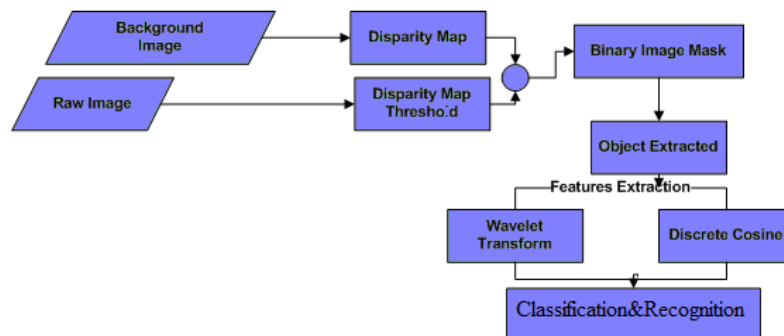


Figure 1. Flowchart of fire/smoke classification

Disparity is an effective method for foreground/background segmentation. Compared with the intensity-based method, disparity-based segmentation is less affected by light constraints, shadows and occlusion; compared with the 3-D-based approach or color based, it is less expensive computationally and suitable for real-time implementation. Most stereo-based segmentation algorithms assume a static background and a pair of stationary cameras, thus, they do not work properly with a changing scene. In contrast, our segmentation and grouping technique does not require a static background and therefore is robust to the changing scene.

Considering a pair of IR stereo images, the aim is to automatically extract a new targeted object from the background, localize it with the use of their stereo information. As mentioned earlier, the main advantage of

relying on stereo instead of the grayscale levels is that the disparity information is quasi invariant to changes of lights and does not require any assumption on the shape of the object, its texture, and/or its appearance.

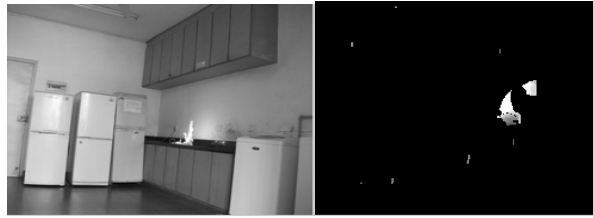


Figure 2. The objects (fire in the kitchen with its background) and their corresponding disparity maps

3. STEREO AND OBJECT EXTRACTION ALGORITHM

The main issue to solve is the determination of the lowest boundary of the house interior in the disparity space given the 3D information recovered from the calculation of the disparity map. The object of interest (see figure 2) will then be defined as all connected pixels with disparity values larger than the predetermined disparity threshold map. The proposed algorithm mainly consists of three steps:

1. Off-line determination of a unique disparity threshold map: Establish a pre-determined disparity threshold map (disparity template) that defines the background surfaces for suppression.

2. Current creation of a binary suppression mask by using the disparity map of the present situation and the pre-determined threshold map. The disparity map of a new pair stereo is generated during the operation and compared with the disparity threshold map. The output is a binary mask. The background pixels are assigned to zeros.

3. Employ morphological closing operator to remove the noise and smooth the foreground regions: due to noise and other artifacts coming from the calculation of the disparity map, the extracted object from the previous step might appear as containing holes or even being sliced into several parts.



(a)



(b)

Figure 3. Illustration of the results of the disparity-based object extraction approach on IR images in indoor scenes. The rows (a), and (b) show the disparity map of an original image (right column) and the extracted object image (left column) of kitchen background, high flame in kitchen, low flame and very low flame, respectively.

3.1 Disparity Threshold Map Determination

The disparity threshold map is generated off-line using a set of representative images of the interiors of a kitchen without fire/smoke presence. Both physical and virtual boundary surfaces with added artificial textures to the physical boundary are used. A physical boundary is defined as the interior of the kitchen or a structure that limits the view depth of the cameras.

For better range accuracy, artificial textures are added to the physical boundary surfaces in creating the disparity threshold map. For the same reason, averaged disparity values of a number of image frames are used in the process. We found experimentally that an average over 200 frames (see figure 4) is good enough to establish the disparity threshold map.

3.2 Background Suppression

A comparison between the measured disparity value and the disparity threshold for all the pixels in the image creates a binary map having the same matrix correlation as the original image, see figure 3. For a given pixel location (i, j) , assume the measured disparity value is $Db(i, j)$ and the disparity threshold is $Dt(i, j)$. The new binary map $Q(i, j)$ with the same matrix correlation as the original image is first created with the following rules: If $Db(i, j) \leq Dt(i, j) + \mu(i, j)$. Then $Q(i, j) = 0$ Otherwise $Q(i, j) = 1$, where the parameter $\mu(i, j)$ is used to control the degree of background suppression.

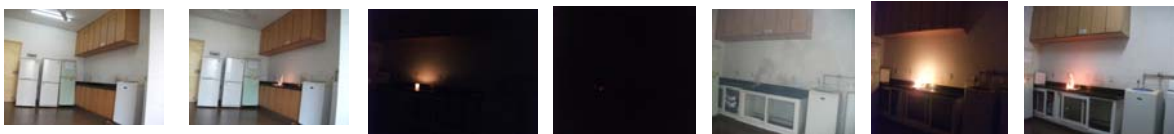


Figure 4. Pre- processing images sample of the results experimental

3.3 Object Extracted Enhancement using Morphological Operators

In order to reduce the noise effects, fill out the holes and/or connect several parts of the object of interest, binary image processing techniques of type opening and closing operations are used to achieve a smoother and reliable background suppression mask. In our implementation, morphological image dilation, singularity suppression and erosion were used. Those operators are implemented in terms of union and intersection; however a faster implementation in terms of convolution and binarisation is underway by Hagyard et al. A 3×3 8-way flat structuring element for morphological operators is applied to the initial binary mask $Q(i, j)$. Two different levels of erosion and other singularity suppression operators are introduced here and applied as follows:

- Erosion Level 1*: If the summation of its 8-way neighbors is greater than or equal to 1, set $Q(i, j)$ to 1; otherwise set $Q(i, j)$ to 0. *Erosion Level 2*: If the product of its 8-way neighbors is 0, then set $Q(i, j)$ to 0, otherwise set $Q(i, j)$ to 1.

- Suppression*: If the summation of its 8-way neighbors is less than or equal to 3, set $Q(i, j)$ to 0; otherwise no change to $Q(i, j)$.

4. FEATURES EXTRACTION AND TRAINING

In our algorithm we have extracted discrete cosine transform (DCT) and discrete wavelet transform (DWT) coefficients as features. First, the first coefficient referred to as the DC coefficient is the average value of the image pixel intensities. Low frequency components represent the general shape and high frequency components represent edges and finer details in an image. The low frequency components are towards the top left and high frequency components are towards the bottom right of the block of transformed data. The DCT is applied to the image block and only the first 100 of the total transformed coefficients are retained after zigzag coding for each image.

Secondly for DWT, the five features chosen are arithmetic mean, geometric mean, standard deviation, skewness and entropy. In normal (non-fire) images, these features follow a certain pattern which is quite different from images with fire. Hence all the features were retained and these were calculated for the horizontal, vertical and diagonal components at each level. This generates six features for each block which results in a total of 50 features for three levels. These 50 features are used as the input to the classifier.

For classification, a neural network approach has been used. The design of the input data to the neural network (NN) is important as it directly affects the performance of the network. Our goal is to make the input

data maintain the shape information for recognition while being reduced to a manageable amount. These 50 features from 10 blocks and 5 statistical approaches from the DWT and the first 100 coefficients of the DCT from each image are used as the input to the classifier. The network is trained to produce an output of 0.9 if a fire is present and 0.1 otherwise. Thus, we classify the recognized object by thresholding the output value of the trained network: if the output is larger than the threshold, then the input object is classified as a fire, otherwise as a no fire.

5. EXPERIMENTAL RESULTS

1330 images of fire/no fire and smoke/no smoke were used to test the proposed method. Every image has been resized to 256×192. The data set contains eight fire/smoke categories: DWF(Day With Fire), DNF(Day No Fire), NWF(Night Without Fire), NNF(Night No Fire), DWS(Day With Smoke), DNS(Day No Smoke), NWS(Night With Smoke), and NNS(Night No Smoke). Within DNF and NNF, we used several different environments such as candles, electrical tubes, paper fire and plastic fire in night shots or daytime shots to gain the variability. In order to get variability of background, we took ON/OFF lights for daytime and night shots. For the training of classifiers, a three-layer back propagation neural network structure was used. About 5000 iterations are run and the optimal solution was picked up.

Table 1. Fire/smoke recognition performance based on disparity segmentation method.

Class	Total	Found	Correct%
DWF	240	220	91.6
DNF	330	260	78.78
NWF	120	90	75
NNF	90	80	88.88
DWS	60	50	83.3
DNS	360	290	80.5
NWS	40	30	75
NNS	90	70	77.7
Overall	1330	1090	81.9

$$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) \times 100 \quad (1)$$

$$Sensitivity = TP/(TP+FN) \quad (2)$$

$$Specificity = TN/(TN+FP) \quad (3)$$

Where TP is the true positive, TN the true negative, FP the false positive, and FN the false negative see Equations (1), (2) and (3). The comparison provided below shows that the disparity based segmentation is a very promising way to improve the performance of fire/smoke recognition.

Table 2. Fire/smoke recognition performance: using k-means

Class	Total	Found	Correct%
DWF	240	230	95.83
DNF	330	310	93.93
NWF	120	60	50
NNF	90	70	77.77
DWS	60	49	81.66
DNS	360	270	75
NWS	40	24	60
NNS	90	50	55.55
Overall	1330	1 063	79.92

Note that the overall accuracy see equation (1) is rather low due to the fact of the significantly different conditions between the training and testing data set.

Table 3. Performance comparative analysis: disparity and k-means segmentation-based

Class	Segmented images			Non-segmented images		
	Accuracy (%)	Sensitivity	Specificit y	Accuracy (%)	Sensitivit y	Specificit y
DWF	88.8	0.95	0.8	66.7	0.72	0.65
DNF	91.6	0.96	0.93	80	0.89	0.62
NWF	80	0.9	0.85	40	0.75	0.43
NNF	75	0.88	0.66	56	0.69	0.52
DWS	66.6	0.83	0.66	50	0.54	0.36
DNS	92	0.96	0.87	49	0.9	0.27
NWS	57	0.75	0.87	44	0.6	0.46
NNS	75	0.875	0.8	42	0.69	0.39
Overall	62.6	0.71	0.644	42.7	0.578	0.37

As shown in literature review, most fire and flame detection/recognition algorithms are based on color and motion analysis in video. However, all of these algorithms focus on either day-time flame/smoke recognition. Fires occurring at night have different temporal and spatial characteristics than daytime fires, as shown in table 1 and table 2. The Color method based on k-means in table 3 misses 25% up to 40% for images taken night with smoke (NWS) and it can be seen that the accuracy as we compare disparity versus k-means varies from 78.25 down to 50.6 % (NWS and NNS), see table 3; critical case comes when we consider images taken night. This makes it necessary to develop explicit segmentation methods for video based fire detection/recognition at both day and night. Other advantage of our algorithm is that we take into account such color variations/motion using wavelet transform features to detect high-frequency behavior inside the regions.

6. CONCLUSION

A new approach based on stereo vision image segmentation has been proposed for fire/smoke recognition. To check the robustness of the used technique, stereo vision segmentation was carried out on indoor fire/smokes frames. The input images were first segmented according to a pre-determined disparity threshold map and the real-time disparities of the fire. Binary image processing techniques are used to reject noise. The system is implemented in Matlab on a Window PC and is shown to work well using a disparity based approach as well as on motion segmented images. An excellent cross validation of the classifier with image segmented based on disparity performed better in all categories by 1% to 37.03% across the eight categories. The averaged overall accuracy gain is about 27.6% for disparity based segmentation better than that of the classifier when the images were segmented using the k-means algorithm.

ACKNOWLEDGEMENT

This research is partially supported by National Natural Science Funds of China (No.60970098 and No.60803024), Fundamental Research Funds for the Central Universities (No.201021200062), Specialized Research Fund for the Doctoral Program of Higher Education (No.20090162110055 and No.200805331107), Open Project Program of the State Key Lab of CAD&CG, Zhejiang University (No.A1011 and No.A0911).

REFERENCES

- Behcet U, gur TÅoreyin. 2009. "Fire Detection Algorithms using Multimodal Signal and Image Analysis". *Thesis of doctor of philosophy, Institute of engineering and science of Bilkent university.*
- Bo-Ho Cho, Jong-Wook Bae, and Sung-Hwan Jung. 2008. "Image Processing-Based Fire Detection System using Statistic Color Model". *Advanced Language Processing and Web Information Technology, 2008. ALPIT '08. International Conference. Dalian Liaoning . 245 - 250*

- B. Uğur Toreyin , Yiğithan Dedeoğlu , Uğur Guđuđkbay , and A. Enis Cetin. Received 26 June 2004; revised form 20 June 2005. "Computer vision based method for real-time fire and flame detection". *Elsivier, Pattern recognition*
- Chen Jun, Du Yang, Wang Dong. 2009. "An Early Fire Image Detection and Identification Algorithm Based on DFBR Model". *Computer Science and Information Engineering, 2009 WRI World Congress*. CA USA. page(s): 229 - 232
- D.M.P. Hagyard, M. Razaz, and P. Atkin. 1997 "A Fast Algorithm for Computing Morphological Image Processing Primitives". *IEEE Workshop on Nonlinear Signal and Image Processing*. Michigan USA.
- DG Lowe iccv, 1999. "Object recognition from local scale-invariant features"
- George Privalov and James A. Lynch. "Video Image Detection Systems for Fire and Smoke".
- Moulay A. Akhloufi, Lucile Rossi1, Lilia Abdelhadi and Yves Tison. 2008. "Dynamic fire modeling in Three-dimensional space". *Image Processing Theory. Tools & Applications*.
- Nicholas True. "Computer Vision Based Fire Detection". University of California. San Diego, 9500 Gilman Drive.
- T. Celik, H. Ozkaramanli, H. Demirel. 2007. "Fire Pixel Classification Using Fuzzy Logic and Statistical Color Model". *ICASSP 2007. IEEE International Conference on Acoustics, Speech, and Signal Processing. Hawaii, USA*. vol. I. pp. 1205-1208.

AN IMAGE FEATURE DESCRIPTORS-BASED RECOVERY ACTIVATION METRIC FOR FLIR TARGET TRACKING

Gianluca Paravati, Andrea Sanna and Fabrizio Lamberti
*Politecnico di Torino – Dipartimento di Automatica e Informatica
Corso Duca degli Abruzzi, 24 - 10129 Torino, ITALY*

ABSTRACT

The reliability of image processing algorithms is crucial in target tracking applications where objects need to be identified along a video sequence. In general, the effectiveness of target tracking algorithms is evaluated in terms of number of target losses. Several algorithms that are triggered when the tracking phase fails exist. The challenge in the considered research area is to find a generic recovery activation metric that does not need to be finely tuned to properly work with all the video sequences. This paper describes a novel method to detect false alarms in target tracking applications based on image-based feature descriptors. The number of matches between the local feature descriptors in the reference target and in the candidate target is used to assess the reliability of the tracking algorithm used and, in case, to activate a suitable recovery mechanism. The proposed approach has been evaluated on a series of Forward Looking Infrared (FLIR) sequences provided by the U.S. Army Aviation and Missile Command (AMCOM).

KEYWORDS

Target tracking; Forward Looking Infrared (FLIR) imagery; Intensity Variation Function (IVF); Template Matching (TM); Image-based Features; Harris Corner Detector.

1. INTRODUCTION

Target tracking algorithms are an important research topic in computer vision. Different application scenarios attempt to infer information through the analysis of recorded or real-time digital video sequences. For example, in Nykolaychuk (2010) objects are tracked to classify their motion in order to decide whether they are moving in an admissible way or not; this kind of analysis may be useful for tracking vehicles (e.g. in automatic traffic monitoring tools), people (e.g. for the detection of potential dangerous situations), etc.

In this paper, the focus is on target tracking in Forward Looking Infrared (FLIR) video sequences, where the thermal imprint of a target is a distinctive feature with respect to background and clutter. Since FLIR imagery are created by exploiting thermal energy, they are particularly suitable for surveillance applications, for locating pinpointing sources of ignition during firefighting operations, for helping piloting of aircraft, for warning drivers about obstacles on the road as well as for various military applications. For example, in the automatic control of security areas, a FLIR-based object tracking algorithm may be used to control a camera with a servo motor in closed-loop, thus allowing the camera to follow a target as it moves in the field of view. Another possible scenario concerns the possibility of detecting and localizing targets in hazardous or disaster areas, where rescue robots might be partially controlled by tracking algorithms. All these tasks are particularly challenging at night, or through fog, smoke and haze. FLIR cameras effectively help detecting and following targets in all the above low-visibility conditions.

Although the literature is thriving of works concerning object tracking in visible wavelengths (Pathan, 2009; Lucena, 2008), only a limited attention has been devoted to infrared imagery. A comprehensive survey on target tracking methods can be found in Moeslund (2006).

Generally, the effectiveness of target tracking algorithms is evaluated in terms of number of target losses. There exist several algorithms that are triggered when the tracking phase fails, ranging from template model-based approaches to multi-decision fusion techniques. In general, recovery algorithms are activated (and exhibit good performance) only on the trickiest frames of a video sequence; the reason is that they might be

computational intensive and some of them can even drift when applied at any frame, like for example the approaches based on Template Matching (TM).

In this paper, a novel strategy for identifying frames where the tracked target is likely to be lost and for controlling the activation of the necessary recovery step is proposed. The designed strategy is based on a features matching technique that is applied onto consecutive frames of the video sequence under analysis. A lightweight target tracking algorithm is in charge of tracking the object of interest within an image sequence. Whenever the activation strategy perceives a potential false alarm in the tracking phase, a recovery algorithm is triggered.

For evaluation purposes, since in this work the focus is on infrared target tracking, experimental tests have been carried out by using the well-known Intensity Variation Function (IVF) technique as the tracking algorithm, whilst the recovery phase has been deployed by means of a TM-based method. The proposed recovery activation strategy relies on an implementation of image-based feature descriptors (Azad, 2009). The challenge in the considered research area is to find a generic recovery activation solution that does not need to be finely tuned (e.g. through a set of thresholds) to properly work with the majority of the video sequences. Specifically, the presented contribution provides an answer to this need by proposing a solution that does not require any prior training or study, because the designed recovery phase is able to self-adapt on a frame-by-frame basis to the specific characteristics of the sequence or target being considered by exploiting the number of local features belonging to the target to be tracked.

The remainder of this paper is organized as follows. Section 2 briefly reviews the background and related work in the field of infrared target tracking. Section 3 focuses on the reference algorithms selected for the tracking and recovery phases. The proposed metric for controlling the activation of the recovery step is described in the details in Section 4. The evaluation of the proposed solution is analyzed in Section 5. Finally, Section 6 draws conclusions and sketches future work directions.

2. BACKGROUND

Different methodologies have been proposed for automatic target tracking, e.g., based on spatio-temporal correlation techniques, on the mean-shift algorithm, on filtering and data association methods, and so forth.

Mean-shift based approaches iteratively shift the position of a kernel in the image space so as to find similar past and current observations (Comaniciu, 2000). As a matter of example, in Lucena (2008) a generalization of the mean-shift algorithm has been proposed, exploiting a set of probabilistic rules for choosing among different update models that represent the same object. Mean-shift based approaches are widely used both in the visible and infrared spectrum, as they provide for a general optimization solution that is independent of target features (Yilmaz, 2001). The main drawback of the standard mean-shift algorithm is that it requires that, in a new frame, at least some parts of the target reside inside the kernel. Kalman filters have traditionally been used in automatic target tracking with the aim of estimating the target kinematics and therefore predicting its position; thus, they can be effectively used in combination with mean-shift techniques and additionally represent a viable approach to deal with partial occlusions (Salmond, 2001).

Because of the distinguishing nature of infrared images (targets in FLIR images show up either as particularly bright or dark objects), they have historically been analyzed as a template correlation process (Parry, 1997). This process starts by initializing a template which contains a representative target signature; then, tracking is performed by trying to maximize the matching with the above template. After initialization, the template can be maintained unaltered during the tracking phase till the target signature does not exhibit significant temporal evolutions. However, one of the main challenges in FLIR target tracking is that two consecutive frames may exhibit little correlation between the regions of interest (i.e., image regions containing the target). This challenge is known as the “template update problem” (Iain, 2004); according to the authors, specific strategies need to be designed in order to prevent the template from becoming “stale” (i.e. the template no longer provides for a thorough characterization of the target). For instance, in Johnston (2009) and Mould (2008), a dual domain approach is presented to improve tracking accuracy by automatically detecting when a template update is needed through a combination of pixel domain and modulation domain correlation trackers. As the failure modes in the two domains are rarely the same, the above technique proved to be able to hold on the target in almost all the considered frames from the experimental data set.

The tracking technique chosen as a reference for this work (Bal, 2005) represents an interesting test-bed for the experimentation of feasible solutions to the template update problem. A compact signature based on a target intensity variation is used for frame-to-frame detection and tracking. However, it may generate false alarms that could lead to a target loss. When the reliability of the IVF-based tracking cannot be guaranteed, i.e. a false alarm is generated, a TM-based algorithm making use of an up-to-date template model is triggered as a recovery tool.

A false alarm may be due to different reasons. Basically, three failure modes have been identified so far (Dawoud, 2006). The first one is related to non-fixed cameras that introduce abrupt discontinuities in target position due to their ego-motion. The second failure mode occurs because of the presence of other targets as well as of other unwanted image artifacts with similar target signature, e.g. blurring, cluttering, and intensity variation. The last failure mode is known as the “drifting problem” (Han, 2007), and it is related to the behavior of the TM phase. More in details, when the TM technique is activated, it uses information from the target in the previously determined position to recover from the false alarm. However, because of changes in target shape, size, orientation, etc., a partially wrong template can be obtained in some cases; when this behavior is repeated over consecutive frames, small tracking errors are accumulated and recovery leads to incorrect results.

In Bal (2005) a strategy for identifying possible false alarm situations and for activating the recovery algorithm is proposed. Basically, it is based on a distance measure between the reference and the estimated target position. When the distance is larger than a given threshold, the recovery mechanism is activated. The main drawback of this solution consists in the difficulty associated with finding, if existing, the best threshold value for a given sequence (and, possibly, for other sequences). As, in general, each sequence requires a different threshold value to let the tracking algorithm properly work, the recovery activation mechanism should be improved, in order to make it self-adaptive.

3. THE REFERENCE FLIR TRACKING ALGORITHM

The tracking technique selected as the reference for this work is based on the assumption, which is made by many FLIR tracking systems, that the target is brighter than the background (Bal, 2005). In this technique a model of the target intensity profile is built. This model produces a maximum peak where the intensity variation of the reference target is similar to that of the candidate target. The IVF technique requires two parameters to be defined, i.e. the size of the target window and of the subframe. The target window roughly corresponds to the rectangular area including the target in a given frame, whereas the subframe defines the area where the target is searched for in the next frame. It should be emphasized that, as the proposed methodology deals with the recovery phase, it is general and it could be easily applied also to other approaches not relying on the IVF technique.

In the original implementation of the algorithm (Bal, 2005), during the tracking phase for each frame n the subframe is positioned at the reference target location from the previous frame $n-1$, and the intensity variation function F^n is computed over the subframe domain S^n as:

$$F^n(v, z) = \frac{1}{\Lambda} \sum_{j=1}^l \sum_{i=1}^k |S^n(i+v, j+z) - \omega^{n-1}|, \quad (1)$$

Where Λ is the size ($k \times l$) of the target window, in pixels and (v, z) are the spatial coordinates in the subframe; finally, ω^{n-1} is a 3×3 pixel matrix that represents the local maximum of the target window in the previous $(n-1)$ -th frame. The local maximum within a small 3×3 pixel window is used instead of the maximum value based on a single pixel in order to minimize the impact of noise usually affecting FLIR video sequences.

Candidate target coordinates can be represented as a function of the peak value obtained by an exponential function C_F , which is called the correlation output plane; this function directly depends on the intensity variation function F^n , and it can be expressed as:

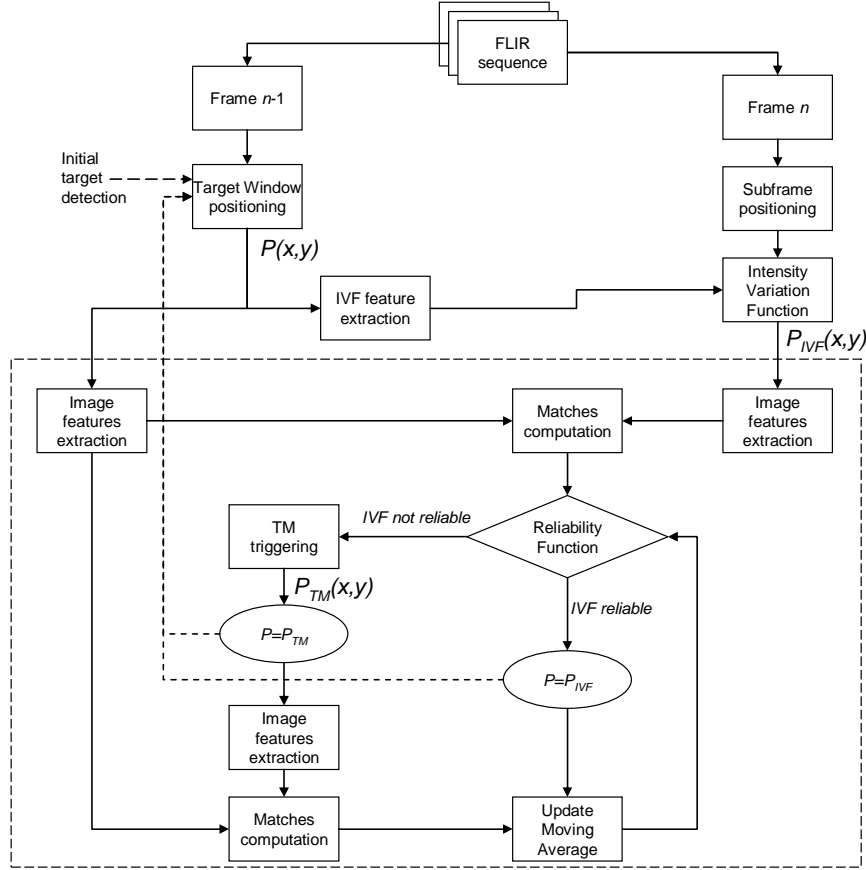


Figure 1. Block diagram of the proposed tracking system.

$$C_F(v, z) = e^{-\lambda F^n(v, z)}, \quad (2)$$

Where λ is a constant value and (v, z) represent the coordinates of the target window in the subframe.

The maximum peak in the correlation output plane is generated where the candidate target intensity variation is similar to that at the reference target position. The coordinates of the peak in the subframe coordinate system are therefore converted into the candidate target position P_{IVF} .

Whenever a false alarm is detected, a recovery mechanism based on the activation of a TM-based method is run. In its original implementation (Bal, 2005), the template model is built as:

$$T^n(v, z) = \frac{1}{\Lambda} \sum_{j=1}^l \sum_{i=1}^k |S^n(i+v, j+z) - W^{n-1}|, \quad (3)$$

Where (v, z) , Λ , and (k, l) have the same meaning in (1), whereas W represents a window slightly larger than the reference target window (i.e., it contains both the target and background information); in this way, information about target shape are taken into account for the recovery phase. A correlation output plane can be computed similarly to (2):

$$C_T(v, z) = e^{-\lambda T^n(v, z)}. \quad (4)$$

The candidate target position is updated with the coordinates of the maximum peak in the correlation output plane C_T , which are converted into a position P_{TM} in the image coordinate system.

In Bal (2005), the occurrence of a potential false alarm is detected by using a metric based on the Euclidean distance between the position of the target identified by the IVF step (P_{IVF}) and the position of the target in the previous frame. Whenever this distance exceeds a particular threshold value (β) that has been set for the considered image sequence, the recovery step is activated and P_{TM} is computed. However, finding the most appropriate value of β (possibly valid for any sequence) is a task hard to accomplish. The next section will address the above issue by presenting a self adaptive activation metric.

4. THE PROPOSED RECOVERY ACTIVATION METRIC

Figure 1 shows an overview of the designed tracking methodology. The block diagram contained in the dashed box highlights the proposed self-adaptive criterion for the activation of the recovery step.

Frame n is the current image frame to be processed, whereas frame $n-1$ represents the previous frame. A new target window is initialized and positioned whenever a new target is identified by a detection step. Since the detection step is out of the scope of this paper, here it is sufficient to say that an external module is able to provide the initial position $P(x,y)$ and size of any given target. Once this information is available, the position of the target window in the remaining of the sequence is updated automatically by the system.

The target window is analyzed to extract the features needed to both perform the tracking phase and a reliability control on the intermediate tracking results. The IVF feature extraction block takes as input the target position and it is aimed at determining ω in equation (2). When a new frame is available (n -th frame), the subframe is centered in the position of the target at frame $n-1$. The IVF algorithm is executed in the specified subframe domain by using the value of ω previously computed. The output of this block constitutes the “temporary” position P_{IVF} of the target according to the IVF technique. The reliability of this result is evaluated in the next steps.

The inputs of the recovery macro-block are the reference position P of the target at frame $n-1$, either determined by the IVF technique or by the recovery step, and the candidate position P_{IVF} of the target at the current (n -th) frame. For each frame, a list of keypoint locations around the specified input position is extracted to find distinguishing features of both the reference and the candidate target. The features extraction algorithm is based on an implementation which combines the Harris corner detector with image-based feature descriptors to improve feature matching speed (Azad, 2009). This implementation just computes the image-based feature descriptors for a set of identified Harris interest points, thus allowing fast object recognition. The size of the region of interest (RoI) where feature descriptors are calculated should be greater than the target size and smaller than the subframe size. Since FLIR images often present targets that appear as saturated, the features should be searched for in an area that comprises the edges of the target, thus maximizing the number of local descriptors to be possibly identified. As a rule of thumb, a rectangular search area ($W_{RoI} \times H_{RoI}$) whose size is the average between the target window size ($W_{TW} \times H_{TW}$) and the subframe size ($W_S \times H_S$) is used:

$$\begin{aligned} W_{RoI} &= \frac{W_{TW} + W_S}{2} \\ H_{RoI} &= \frac{H_{TW} + H_S}{2} \end{aligned} \quad (5)$$

The next step concerns the computation of the features matching, where the correspondences between the list of features of the reference and the candidate target are identified.

Defining N_m^n as the number of detected matches at the n -th frame, the reliability function of the candidate target position can be expressed as:

$$R(P_{candidate}) = \begin{cases} 1 & \text{if } N_m^n > SMA_m \\ 0 & \text{if } N_m^n \leq SMA_m \end{cases}, \quad (6)$$

Where SMA_m is the Simple Moving Average of the matches and the candidate target position $P_{candidate}$ is equal to the previously computed P_{IVF} . The unweighted mean of the previous p matches with respect to the current n -th frame is computed as follows:

$$SMA_m(p) = \frac{\sum_{i=n-p}^n N_m^i}{p}, \quad (7)$$

Where p is the selected period (e.g. $p=5$ frames) and N_m^i is the number of matches at the i -th frame, with $n-p < i < n$.

In this way, if the IVF module correctly determines the position of the target in the current frame, the number of matches (i.e. the reliability of the new position) should be high. On the other hand, when the IVF tracking algorithm fails to correctly locate the target, the number of matches should be low; hence, it

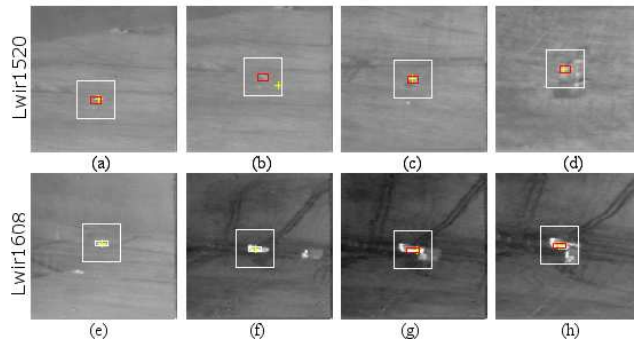


Figure 2. Tracking results obtained with the proposed metric on sequences Lwir1520 and Lwir1608.

becomes possible to detect a false alarm and trigger a suitable recovery mechanism. Since the number of matches strongly depends on the characteristics of the target (e.g., degree of saturation of the target in the FLIR image, target size, contrast with respect to the background and clutter), it has been decided to take into account its statistic information in order to determine whether the position P_{IVF} has to be considered as a false alarm or not. This is the reason why it has been introduced the SMA_m function, which can be also expressed as in the following in order to reduce its computational complexity:

$$SMA_m^n(p) = SMA_m^{n-1}(p) - \frac{N_m^{n-p}}{p} + \frac{N_m^n}{p}, \quad (8)$$

In (8), the SMA at the current n -th frame is expressed in terms of the SMA at the previous $(n-1)$ -th frame by adding and removing the contribution of the more recent (n) and the oldest ($n-p$) number of matches, respectively.

When the number of matches N_m^n is at least 3, there is a strong evidence for the presence of the searched object in the region of interest. In this case, if the reliability function (6) returns the value 1, the result of the IVF tracking is considered as reliable and the new position of the target window is updated to $P = P_{IVF}$. On the other hand, if the function returns the value 0 or if the number of matches does not reach the indicated threshold, a false alarm is detected and the TM recovery mechanism in (3) is triggered. This module generates a new position of the target P_{TM} , or confirms that the previously computed position P_{IVF} is not a false alarm; in any case, the position of the target window is updated to $P = P_{TM}$.

Prior to analyze the next frame of the sequence, it is necessary to update the moving average in (8). When the temporary tracking result is considered as reliable by (6), the number of matches has been already computed and no further steps are needed. On the other hand, when the recovery phase identifies a different target position, a list of image feature-based keypoint locations is extracted from the current n -th frame around the position P_{TM} . Therefore, the new number of matches N_m^n between the set of current keypoint locations and the set of keypoint locations computed at the previous $(n-1)$ -th frame is used to update the moving average SMA in (8).

5. EXPERIMENTAL RESULTS

In this section, experimental results obtained by using the proposed recovery activation metric on well-know infrared video sequences are discussed. In particular, the FLIR database provided by the U.S. Army Aviation and Missile Command (AMCOM) is used; the detection and tracking phases are particularly challenging for AMCOM sequences because of clouds and other obscurants, as well as because of the presence of targets which are difficult or impossible to see in a single frame and can be identified only thanks to their motion.

The activation strategy has been tested by using the IVF tracking algorithm and the TM method described in Bal (2005). The IVF algorithm is in charge to track the new position of a given target. The above algorithm is superseded by the TM technique when a target is likely to be lost with the IVF approach. As already said in Section 3, the implementation in Bal (2005) uses a distance-based strategy to detect false alarms. However, it is difficult to find a value of the threshold β which is valid for all the sequences. For example, tracking on a particular sequence can fail by using certain values of β but can succeed when other

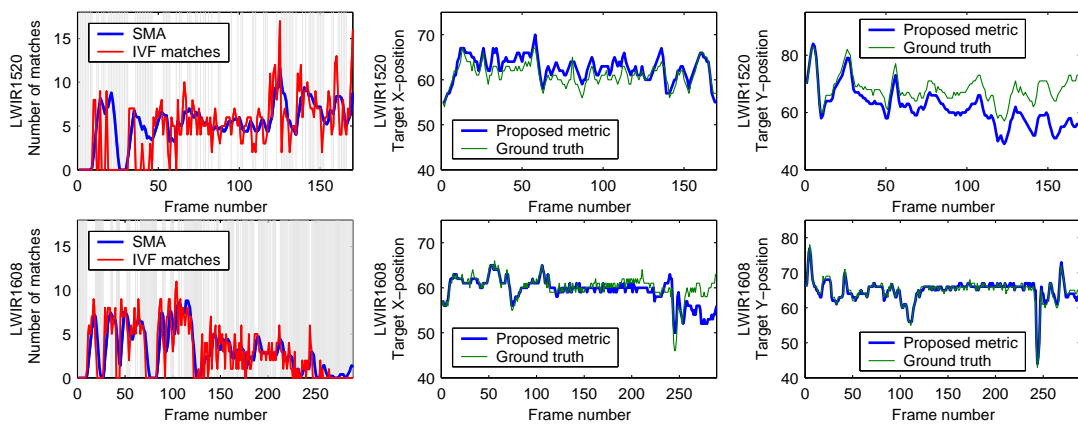


Figure 3. Behavior of the proposed activation metric on sequences Lwir1520 and Lwir1608 and tracking errors with respect to AMCOM ground truth.

values of β are considered. It is worth to highlight that none of the considered test sequences can be tracked correctly without the intervention of the recovery step. The proposed image feature-based recovery mechanism automatically detects a potential false alarm and triggers the TM algorithm, accordingly. By using the proposed technique all the considered sequences has been successfully tracked.

In order to allow for visual inspection, Figure 2 shows some snapshots taken during the tracking phase with the proposed recovery mechanism for a subset of test sequences. The small rectangular window identifies the target window, positioned after a possible activation of the recovery mechanism. The large rectangular window represents the area where the target is searched for (i.e., the subframe). A cross indicates the temporary position of the target suggested by the IVF algorithm, before the recovery mechanism is triggered. These sequences show some common false alarm situations. For example, in sequence Lwir1520, (Figure 2.b), the tracking phase generates a false alarm due to background clutter.

In order to better understand the insights of the proposed solution, additional details regarding the sequences shown in Figure 2 are provided in Figure 3. For each tracked sequence, three graphs are shown. The first graph reports, for any given frame, the trend of the SMA function and the number of image feature matches in the neighborhood of the target position given by the IVF algorithm before the false alarm detection. Since the area where the local features are searched for is a few pixels wide and FLIR images are not very detailed, the number of matches is only in the order of ten (this fact suggests that the application of the proposed method to visible tracking can be more effective). Frames where, according to equation (6), the recovery mechanism is activated are indicated in gray. The SMA function acts as a moving (adaptive) threshold, capable of choosing which is the most correct value to trigger the recovery mechanism on a frame-by-frame basis.

The second and third graphs for each sequence in Figure 3 show the reference target position for any given frame together with ground truth data by AMCOM. Each sequence is correctly tracked. As it can be seen, the curve of the target Y-position in sequence Lwir1520 tends to depart from the ground truth. This behavior is due to the particular algorithm chosen for the tracking phase. In fact, the IVF algorithm tracks the hot spot of a target, which does not necessarily correspond to the center of the target. As the camera approaches the target, the discrepancy becomes more evident. However, the tracker continues to follow the target.

6. CONCLUSIONS

The reliability of image processing algorithms in target tracking applications strongly depends on the activation of suitable recovery processes when a selected target is likely to be lost. The main difficulty in these situations concerns the automatic perception of a target loss. Some approaches face this problem by using distance-based metrics, e.g. activating a recovery mechanism whenever the new position of the target resides outside of a predefined maximum distance from the previous position of the target. However, these

solutions are not inherently able to be applied in general, as the threshold for a test sequence may not be valid for another video sequence. In this paper a local image feature descriptor-based recovery mechanism for detecting false alarms in infrared target tracking, is proposed. The main contribution is to propose a self-adaptive metric able to decide to activate a suitable recovery mechanism basing on the analysis of the target features. In particular, the IVF technique is used for tracking, whereas a TM-based technique acts as a recovery method; however, the idea behind the proposed recovery step activation strategy is general and it can be easily extended to other tracking and recovery algorithms. By automatically recognizing when a target is likely to be lost, the proposed methodology is able to self-adapt to various target and video sequence characteristics, thus improving the reliability of the considered tracking technique. Future works will be aimed at exploring the use of local descriptors to solve partial occlusions during the tracking.

ACKNOWLEDGEMENT

The authors wish to thank Luca Bertinetto for his precious help in setting up the testbed for the experiments.

REFERENCES

- Azad, P. et al, 2009. Combining Harris Interest Points and the SIFT Descriptor for Fast Scale-Invariant Object Recognition. In *Proceedings of Intelligent Robots and Systems*.
- Bal, A. and Alam, M.S., 2000. Automatic target tracking in FLIR image sequences using intensity variation function and template modeling. In *IEEE Transactions on Instrumentation and Measurement*, vol. 54, no. 5, pp. 1846–1852.
- Comaniciu, D. et al, 2000. Real-time tracking of non-rigid objects using mean shift. In *Computer Vision and Pattern Recognition*, vol. 2, pp. 142-149.
- Dawoud, A. et al, 2006. Target tracking in infrared imagery using weighted composite reference function-based decision fusion. In *IEEE Transactions on Image Processing*, vol. 15, no. 2, pp. 404–410.
- Han, T.X. et al, 2007. A drifting-proof framework for tracking and online appearance learning. *Proceedings of 8th IEEE Workshop on Applications of Computer Vision*, pp. 10.
- Höferlin, B. et al, 2010. Selection of an Optimal Set of Discriminative and Robust Local Features with Application to Traffic Sign Recognition. *Proc. of the Int. Conf. on Computer Graphics, Visualization and Computer Vision 2010*.
- Iain, M. et al, 2004. The template update problem. In *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 6, pp. 810–815.
- Johnston, C.M. et al, 2009. Dual domain auxiliary particle filter with integrated target signature update. *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops*, pp. 54–59.
- Lowe, D. G, 1999. Object recognition from local scale-invariant features. *Proceedings of the International Conference on Computer Vision*, Corfu, pp. 1150–1157.
- Lucena, M.J., et al, 2008. Spatially constrained model for mean shift. *Full Paper Proceedings of the International Conference on Computer Graphics, Visualization and Computer Vision (WSCG) 2008*.
- Moeslund, T., et al, 2006. A survey of advances in vision-based human motion capture and analysis. In *Computer Vision and Image Understanding*, 104:90–126.
- Mould, N.A., et al, 2008. Infrared target tracking with AM-FM consistency checks. *Proceedings of IEEE Southwest Symposium on Image Analysis and Interpretation*, pp. 5–8.
- Nykolaychuk, M., et al, 2010. Modeling trajectories of free moving objects with smooth flow fields. *Full Paper Proc. of the International Conference on Computer Graphics, Visualization and Computer Vision (WSCG) 2010*.
- Parry, H.S., et al, 1997. Tracking targets in FLIR images by region template correlation. *Proceedings of SPIE, Acquisition, tracking, and pointing*, vol. 3086, pp. 221–232.
- Pathan, S.S., et al, 2009. Feature-supported Multi-hypothesis Framework for Multi-object Tracking using Kalman Filter. *Proc. of the International Conference on Computer Graphics, Visualization and Computer Vision (WSCG) 2009*.
- Salmond, D., 2001. Target tracking: introduction and Kalman tracking filters. *Proceedings of IEEE Workshop on Target Tracking: Algorithms and Applications*, pp. 1-16.
- Yilmaz, A., et al, 2001. Target-tracking in FLIR imagery using mean-shift and global motion compensation. *Proceedings of IEEE Workshop on Computer Vision Beyond the Visible Spectrum*, pp. 54–58.

A FAST AND SPACE-EFFICIENT DISCRETE FOCAL STACK TRANSFORM

Fernando Pérez Nava and Sergio Fumero

*Departamento de Estadística, Investigación Operativa y Computación
Universidad de La Laguna*

ABSTRACT

Plenoptic cameras are a promising solution to increase the capabilities of current commercial cameras. They offer the possibility of focusing the scene after the capture of the photograph or to obtain 3D information. The use of plenoptic cameras to record a 3D scene has some benefits with respect to 3D capture systems based on camera arrays since there is no need of geometric and color calibration or frame synchronization. This paper presents a new technique to obtain the focal stack and 3D information from a plenoptic camera that drastically reduces the memory requirements of the previous Fast DFST algorithm without an increase in its computational complexity. From a practical point of view, in typical scenes we are able to decrease more than 60 times the memory needs of the Fast DFST. This new algorithm makes practical to obtain focal and 3D information with very high depth resolution from current plenoptic cameras. The information obtained can be used as the input of some autostereoscopic displays on the market and shows the feasibility of a plenoptic approach to 3DTV.

KEYWORDS

Computational Photography, Plenoptic cameras, Lightfield, Focal Stack, Multiview Stereo. 3DTV.

1. INTRODUCTION

This work is included in the field of Computational Photography using a plenoptic camera. A plenoptic camera uses a microlens array between the lens of the camera and the image sensor to capture the 4D lightfield by measuring the radiance and direction of all the light rays in a scene. The conventional 2D images are obtained by 2D projections of the 4D lightfield. The fundamental ideas behind the use of plenoptic cameras can be traced back to the works of Lippmann and Ives on integral photography [1], [2]. One of the first plenoptic cameras based on the principles of integral photography was proposed in computer vision by Adelson and Wang [3] to infer depth from a single image. In their design, the plenoptic camera consisted of a camera body with a single main lens, a microlens array replacing the conventional camera sensor, and a relay lens to form the image on the sensor. More recently Ng et al. [4] presented a similar design, but produced in a portable hand-held device.

In this paper we will show a new technique to obtain photographs with varying focus and 3D depth by means of a single lightfield image from a plenoptic camera. Our approach to the 3D recovery problem is based on depth-from-focus approach. First, we process the lightfield image to obtain a focal stack i.e. to produce conventional photographs focused at frontal planes of different depths in the scene, and then we examine the elements in focus to recover the 3D depth of the scene. Our main contribution is related to the refocusing process whose goal is to obtain the focal stack. Given the lightfield of a scene, it is theoretically possible to obtain a conventional photograph focused at a desired depth by means of the Photography integral operator [4]. A plenoptic camera is composed of $n \times n$ microlenses and each microlens generates a $m \times m$ image. In typical applications the number of desired planes is $O(m^2)$. In this case, the brute-force approach would need $O(n^2 m^4)$ operations to generate all the photographs refocused on the determined planes. A significant improvement to this performance was obtained in [4] with the “Fourier Slice Photography” (FSP) technique that reformulates the problem in the Fourier domain. This technique decreases the computational cost of the discretization of the Photography operator to $O(n^2 m^2 \log(n))$ operations to generate the refocused photographs. Another approach to the computation of the focal stack is the Discrete Focal Stack Transform

(DFST) [5]. The DFST has several desirable properties: algebraic exactness, geometric exactness and parallelism with continuum theory. However, it only generates $O(m)$ refocused photograph. A Fast DFST was presented in [6] that generates up to $O(nm)$ refocused planes with $O(n^3 m \log(nm))$ operations. The FSP and the Fast DFST have the same computational complexity if $O(n)=O(m)$.

The Fast DFST has an important drawback: its space complexity is $O(n^3 m)$ that is very high for real plenoptic cameras so its applicability is limited. For a plenoptic camera some typical values are $n=256$ and $m=16$ which leads to a requirement of 0.05 Gb. to store a color image. In this case, to generate 256 planes the Fast DFST needs more than 3 Gb. in memory to store the intermediate results. In this paper we propose a modification to the Fast DFST algorithm that drastically reduces the space complexity to $O(n^2 m^2)$ in practical situations where the space complexity of the desired output is the same as the input. In this paper we use this modification to generate the focal stack of the scene. This focal stack is used to obtain 3D information from the plenoptic camera.

This paper is divided in five sections. In Section 2 we introduce the Fast DFST based focal stack. In Section 3 we present our new efficient algorithm to compute the focal stack. Section 4 shows how to solve the 3D reconstruction problem using the focal stack by means of a laplacian-based focusing measure. Section 5 contains some experimental results and Section 6 includes conclusions and future work.

2. THE FAST DISCRETE FOCAL STACK TRANSFORM

In this section we present the Continuous Focal Stack Transform (CFST) which is based on the Photography Transform and the Fast DFST as a discrete approximation of the CFST.

2.1 Definition of the Continuous Focal Stack Transform

To introduce the Photography transform we parameterize the lightfield defined by all the light rays inside a camera. We will use the two-plane parameterization of this lightfield and write $L_F(\mathbf{x}, \mathbf{u})$ as the radiance travelling from position $\mathbf{u}=(u_1, u_2)'$ (apostrophe means transpose) on the lens plane to position $\mathbf{x}=(x_1, x_2)'$ on the sensor plane. F is the distance between the lens and the sensor (see figure 1 adapted from [4]).

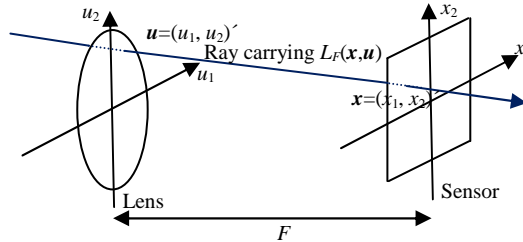


Figure 1. Two plane parameterization of the lightfield.

The lightfield L_F can be used to compute conventional photographs at any depth αF . Let \mathcal{P}_α be the operator that transforms a lightfield L_F at a sensor depth F into a photograph formed on a film at sensor depth αF , then we have [4]:

$$\mathcal{P}_\alpha[L_F](\mathbf{x}) = \frac{1}{\alpha^2 F^2} \int L_F \left(\mathbf{u} \left(1 - \frac{1}{\alpha} \right) + \frac{\mathbf{x}}{\alpha}, \mathbf{u} \right) d\mathbf{u}. \quad (1)$$

This equation explains how to compute photographs at different depths from the lightfield L_F . When we compute the photographs for every sensor depth αF we obtain the Focal Stack transform \mathcal{S} of the lightfield defined as [5]:

$$\mathcal{S}[L_F](\mathbf{x}, \alpha) = \mathcal{P}_\alpha[L_F](\alpha \mathbf{x}) \quad (2)$$

2.2 The Fast Discrete Focal Stack Transform

In practice, the continuous formulation of the focal stack can not be used since a plenoptic camera only captures discrete samples of the lightfield. Discrete lightfields captured from a plenoptic camera are shown on figure 2 [4]. To obtain a discretized version of the focal transform we need to interpolate the lightfield L_F and to discretize the integral in (1).



Figure 2. Plenoptic images and details from the white square in the plenoptic images [4].

To discretize the focal stack, the Fast DFST [6] supposes that lightfield L_F is periodic with periods T_x and T_u and that its values denoted by $L_F^d(\mathbf{x}, \mathbf{u})$ are known for the set of points on a grid $\mathbf{G} = \{(\hat{\mathbf{x}} \Delta x, \hat{\mathbf{u}} \Delta u)\}$, $\hat{\mathbf{x}} = -n_x \dots n_x \in \mathbb{Z}^2$, $\hat{\mathbf{u}} = -n_u \dots n_u \in \mathbb{Z}^2$,

which is a compact notation for a 4D grid where:

$$\hat{x}_i = -n_x \dots n_x \in \mathbb{Z}, \hat{u}_i = -n_u \dots n_u \in \mathbb{Z}, i=1,2, \Delta x = \frac{T_x}{N_x}, \Delta u_i = \frac{T_u}{N_u}, \text{ and:}$$

$$N_x = (N_x, N_x)', N_u = (N_u, N_u)', N_x = 2n_x + 1, N_u = 2n_u + 1,$$

$$\mathbf{n}_x = (n_x, n_x)', \mathbf{n}_u = (n_u, n_u)'$$

Then, L_F^d is defined as the trigonometric polynomial that interpolates the points in \mathbf{G} :

$$L_F^d(\mathbf{x}, \mathbf{u}) = \sum_{\hat{\mathbf{a}}=-n_x}^{n_x} \sum_{\hat{\mathbf{b}}=-n_u}^{n_u} C^d(\mathbf{a}, \mathbf{b}) e^{2\pi i(\mathbf{x}'\hat{\mathbf{a}} + \mathbf{u}'\hat{\mathbf{b}})} |\Delta \mathbf{a}| |\Delta \mathbf{b}|, |\Delta \mathbf{a}| = \frac{1}{|T_x|}, |\Delta \mathbf{b}| = \frac{1}{|T_u|}, \quad (3)$$

$$C^d(\mathbf{a}, \mathbf{b}) = \sum_{\hat{\mathbf{x}}=-n_x}^{n_x} \sum_{\hat{\mathbf{u}}=-n_u}^{n_u} L_F^d(\mathbf{x}, \mathbf{u}) e^{-2\pi i(\mathbf{x}'\hat{\mathbf{a}} + \mathbf{u}'\hat{\mathbf{b}})} |\Delta \mathbf{x}| |\Delta \mathbf{u}|,$$

Where $|\mathbf{x}|$ stands for the product of all components in vector \mathbf{x} and

$$\sum_{i=-n_i}^{n_i} f(\mathbf{i}) = \sum_{i_1=-n_{i_1}}^{n_{i_1}} \sum_{i_2=-n_{i_2}}^{n_{i_2}} f(i_1, i_2). \quad (4)$$

Then, the Fast DFST approximates the integral by sums and defines of the discrete Photography operator for $|\alpha| \Delta x \geq |1 - \alpha| \Delta u$ as [5], [6]:

$$\mathcal{P}_\alpha^d[L_F^d](\mathbf{x}) = \frac{1}{|\alpha|^{2F}} \sum_{\hat{\mathbf{u}}=-n_u}^{n_u} L_F^d\left(\mathbf{u}\left(1 - \frac{1}{\alpha}\right) + \frac{\mathbf{x}}{\alpha}, \mathbf{u}\right) |\Delta \mathbf{u}|, \mathbf{u} = \hat{\mathbf{u}} \cdot \Delta \mathbf{u}. \quad (5)$$

This definition implies a discrete Fourier slice theorem [5] that is used to compute the DFST in terms of the (unaliased) Discrete Fractional Fourier Transform F_N^α (FrFT) (6), which can be computed with the same complexity of the Discrete Fourier Transform (DFT).

$$\begin{aligned}
 \text{Unaliased 1D FrFT} \\
 (F_N^\alpha[f])(a) &= \sum_{x=-n}^n f(x) e^{-\frac{2\pi i a x a}{N}}, \\
 N &= 2n + 1 \\
 \text{Unaliased 2D FrFT} \\
 (F_N^\alpha[f])(\mathbf{a}) &= \sum_{\mathbf{x}=-\mathbf{n}}^{\mathbf{n}} f(\mathbf{x}) e^{-\frac{2\pi i \alpha_1 x_1 a_1}{N_1} - \frac{2\pi i \alpha_2 x_2 a_2}{N_2}} \\
 \alpha &= (\alpha_1, \alpha_2)', \mathbf{N} = 2\mathbf{n} + 1, \mathbf{n} = (n_1, n_2)', \\
 \mathbf{a} &= (a_1, a_2)', a_i = -n_i \dots n_i, \mathbf{x} = (x_1, x_2)'.
 \end{aligned}
 \tag{6}$$

When $\alpha = 1$ in the 1D FrFT we obtain the usual 1D unaliased DFT and when $\alpha = (1, 1)'$ in the 2D FrFT we obtain the usual 2D unaliased DFT. The algorithm to compute a subset of the Fast DFST [6] is described below. Fourier transforms are made over variables not fixed.

Subset Fast DFST rendering algorithm

Input: Sampled lightfield $\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}})$ of size N_x, N_u (both even).

$\hat{x}_i = -n_x \dots n_x - 1, n_x = N_x/2, \hat{u}_i = -n_u \dots n_u - 1, n_u = N_u/2.$

Output: $\hat{S}(\hat{\mathbf{x}}, \hat{p}) = \alpha^2 F^2 |\Delta \mathbf{x}| \mathcal{S}^d [L_F^d](\mathbf{x}, \alpha), \mathbf{x} = \hat{\mathbf{x}} \Delta \mathbf{x}$

$$\left(1 - \frac{1}{\alpha}\right) = \hat{p} \Delta p, \Delta p = \frac{\Delta x}{\Delta u n_p}, \hat{p} = -n_p \dots n_p \in \mathbb{Z}, n_p c = n_p^{ext}, c \in \mathbb{N}$$

Step 0

Pad $\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}})$ as in the DFST algorithm and obtain the extended lightfield $L_F^{d;ext}(\hat{\mathbf{x}}, \hat{\mathbf{u}})$.

Step 1

For every $\hat{\mathbf{u}}$ compute $S1(\hat{\mathbf{a}}, \hat{\mathbf{u}}) = (F_{N_x^{ext}}^{1,1} [L_F^{d;ext}(\hat{\mathbf{x}}, \hat{\mathbf{u}})])(\hat{\mathbf{a}}, \hat{\mathbf{u}})$

Step 2

Initialize $S2(\hat{\mathbf{a}}, \hat{qk}, \hat{mk}) = 0, -n_p \leq \hat{qk} \leq n_p, 0 \leq \hat{mk} \leq c$

For every $\hat{\mathbf{a}}, \hat{\mathbf{u}}$ evaluate $\hat{k} = \hat{\mathbf{u}}' \hat{\mathbf{a}}, \hat{qk} = \lfloor \hat{k}/c \rfloor, \hat{mk} = \text{mod}(\hat{k}, c)$ and update:

$$S2(\hat{\mathbf{a}}, \hat{qk}, \hat{mk}) = S2(\hat{\mathbf{a}}, \hat{qk}, \hat{mk}) + S1(\hat{\mathbf{a}}, \hat{\mathbf{u}}).$$

Step 3

For every $\hat{\mathbf{a}}$ compute $S3(\hat{\mathbf{x}}, \hat{qk}, \hat{mk}) = (F_{N_x^{ext}}^{-1,-1} [S2(\hat{\mathbf{a}}, \hat{qk}, \hat{mk})])(\hat{\mathbf{x}}, \hat{qk}, \hat{mk})$

Step 4

For every \hat{mk} compute $S4(\hat{\mathbf{x}}, \hat{p}, \hat{mk}) = F_{N_p}^{\frac{N_p c}{N_x^{ext} n_p}} [S3(\hat{\mathbf{x}}, \hat{qk}, \hat{mk})](\hat{\mathbf{x}}, \hat{p}, \hat{mk}), N_p = 2n_p + 1.$

Step 5

For every \hat{p} compute:

$$\hat{S}(\hat{\mathbf{x}}, \hat{p}) = \sum_{\hat{mk}=0}^{c-1} e^{\frac{2\pi i \hat{mk} \hat{p}}{N_x^{ext} n_p}} S4(\hat{\mathbf{x}}, \hat{p}, \hat{mk})$$

Step 6

Build $\hat{S}(\hat{\mathbf{x}}, \hat{p})$ extracting from $S5(\hat{\mathbf{x}}, \hat{p})$ the $\hat{\mathbf{x}}$ values $-n_x \leq \hat{\mathbf{x}} \leq n_x - 1$ and finally divide $\hat{S}(\hat{\mathbf{x}}, \hat{p})$ by $(N_x^{ext} N_u)^2$.

The computational complexity of the algorithm above is $O(n^3 m \log(nm))$ decomposed as follows: Step 1 computes m^2 DFTs of size n^2 so its complexity is $O(n^2 m^2 \log(n))$. Step 2 computes $O(n^2 m^2)$ sums. Step 3 computes n^2 DFTs of size $O(nm)$ so its complexity is $O(n^3 m \log(nm))$. If we assume that we want $O(p)$ planes ($p < mn$), Step 4 computes $O(n^3 m/p)$ FrFTs of size $O(p)$ so its complexity is $O(n^3 m \log(p))$. Finally Step 5 computes $O(n^2 p)$ sums of $O(nm/p)$ elements and its complexity is $O(n^3 m)$. Therefore the total complexity if $n \gg m$ is $O(n^3 m \log(nm))$.

The space complexity of the algorithm above is $O(n^3 m)$. decomposed as follows: Step 1 computes m^2 DFTs of size n^2 so its space complexity is $O(n^2 m^2)$. Step 2 needs a $O(n^3 m)$ array to store all the sums. The same array is transformed in steps 3 and 4 so its space complexity is $O(n^3 m)$. If we assume that we want $O(p)$ planes ($p < mn$) in Step 5 the results are stored in a $O(pn^2)$ array. Therefore the total complexity if $n \gg m$ is $O(n^3 m)$ and is independent of the number p of desired planes. As it was explained in Section 1 the $O(n^3 m)$ space complexity makes the algorithm impractical for real plenoptic images.

3. A SPACE EFFICIENT VERSION OF THE FAST DFST

In this section we present a new technique to compute the Fast DFST that computes $O(nm)$ planes with the same global computational complexity of $O(n^3 m \log(nm))$ that presents the previous algorithm. This new algorithm is based on the following theorem:

Theorem

If $\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}}) = L_F^d(\hat{\mathbf{x}}\Delta x, \hat{\mathbf{u}}\Delta u)$ is a sampled lightfield over a grid G defined as: $\mathbf{x} = \Delta x \hat{\mathbf{x}}, \hat{\mathbf{x}} = -n_x \dots n_x, \mathbf{u} = \Delta u \hat{\mathbf{u}}, \hat{\mathbf{u}} = -n_u \dots n_u$, and $(1 - \frac{1}{\alpha}) = \hat{p}\Delta p, |\alpha|\Delta x \geq |1 - \alpha|\Delta u, \hat{p} = -n_p \dots n_p \in \mathbb{Z}, N_p = 2n_p + 1, n_p = 2n_x n_u$, then for $\mathbf{t} = \Delta x \hat{\mathbf{t}}, \hat{\mathbf{t}} = -n_x \dots n_x$, the Photography operator P_α^d can be written as:

$$P_\alpha^d[L_F^d](\alpha \mathbf{t}) = \frac{|\Delta \mathbf{u}|}{\alpha^2 F^2 |\mathbf{N}_x|} \left(\mathbf{F}_{N_x}^{-1, -1} \left(F_{N_p}^{-\left(\frac{\Delta u N_p \Delta p}{\Delta x N_x}\right)} \left[\sum_{\hat{\mathbf{u}}=-n_u}^{n_u} (\mathbf{F}_{N_x}^{1,1}[\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}})])(\hat{\mathbf{a}}, \hat{\mathbf{u}}) \delta(\hat{\mathbf{a}}, \hat{\mathbf{u}}, \hat{k}) \right] \right) (\hat{\mathbf{a}}, \hat{p}) \right) (\hat{\mathbf{x}}, \hat{p}) \quad (7)$$

$$\delta(\hat{\mathbf{a}}, \hat{\mathbf{u}}, \hat{k}) = \begin{cases} 1 & \hat{\mathbf{u}}' \hat{\mathbf{a}} = \hat{k} \\ 0 & \hat{\mathbf{u}}' \hat{\mathbf{a}} \neq \hat{k} \end{cases}$$

Transforms are made over variables not fixed. The resulting algorithm is presented below

Space- Efficient Fast DFST rendering algorithm

Input: Sampled lightfield $\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}})$ and N_x, N_u (both even).

$\hat{x}_i = -n_x \dots n_x - 1, n_x = N_x/2, \hat{u}_i = -n_u \dots n_u - 1, n_u = N_u/2$.

Output: $\hat{S}(\hat{\mathbf{x}}, \hat{p}) = \alpha^2 F^2 |\Delta x| \mathcal{S}^d[L_F^d](\mathbf{x}, \alpha), \mathbf{x} = \hat{\mathbf{x}}\Delta x$

$(1 - \frac{1}{\alpha}) = \hat{p}\Delta p, \Delta p = \frac{\Delta x}{\Delta u n_p}, \hat{p} = -n_p \dots n_p \in \mathbb{Z}$

Step 0

Pad $\hat{L}_F^d(\hat{\mathbf{x}}, \hat{\mathbf{u}})$ as in the DFST algorithm and obtain n_x^{ext} and the extended lightfield $L_F^{d;ext}(\hat{\mathbf{x}}, \hat{\mathbf{u}})$.

Compute $n_k^{ext} = n_x^{ext} n_u, N_k^{ext} = 2n_k^{ext} + 1$

Step 1

For every $\hat{\mathbf{u}}$ compute $S1(\hat{\mathbf{a}}, \hat{\mathbf{u}}) = (\mathbf{F}_{N_k^{ext}}^{1,1}[L_F^{d;ext}(\hat{\mathbf{x}}, \hat{\mathbf{u}})])(\hat{\mathbf{a}}, \hat{\mathbf{u}})$

Step 2

For every $\hat{\mathbf{a}}$

Step 2.1

Initialize $S2(\hat{k})=0$.

For every $\hat{\mathbf{u}}$ evaluate $\hat{k} = \hat{\mathbf{u}}' \hat{\mathbf{a}}$, and update:

$$S2(\hat{k}) = S2(\hat{k}) + S1(\hat{\mathbf{a}}, \hat{\mathbf{u}}).$$

Step 2.2

Compute $S3(\hat{p}) = F_{N_k^{ext}}^{\frac{N_k^{ext}}{N_x^{ext} n_p}} [S2(\hat{k})](\hat{p}), \hat{p} = -n_p \dots n_p \in \mathbb{Z}$

Step 2.3

For every \hat{p} store:

$$S4(\hat{\mathbf{a}}, \hat{p}) = S3(\hat{p})$$

Step 3

For every $\hat{\mathbf{a}}$

Compute $S5(\hat{\mathbf{x}}, \hat{p}) = (\mathbf{F}_{N_x^{ext}}^{-1, -1}[S4(\hat{\mathbf{a}}, \hat{p})])(\hat{\mathbf{x}}, \hat{p})$

Step 4

Build $\hat{S}(\hat{\mathbf{x}}, \hat{p})$ extracting from $S5(\hat{\mathbf{x}}, \hat{p})$ the $\hat{\mathbf{x}}$ values $-n_x \leq \hat{\mathbf{x}} \leq n_x - 1$ and finally divide $\hat{S}(\hat{\mathbf{x}}, \hat{p})$ by $(N_x^{ext} N_u)^2$.

If we assume that we want $O(p)$ planes ($p < mn$), the computational complexity of the algorithm above is $O(n^3 m \log(nm))$ decomposed as follows: Step 1 computes m^2 DFTs of size n^2 so its complexity is $O(n^2 m^2 \log(n))$. Step 2.1 computes $O(n^2 m^2)$ sums. Step 2.2 computes $O(n^2)$ FrFTs of size $O(nm)$ to obtain $O(p)$ elements so its complexity is $O(n^3 m \log(p))$. Therefore the total complexity if $n \gg m$ is $O(n^3 m \log(nm))$.

The space complexity of the algorithm above is $O(n^2m^2)$ decomposed as follows: Step 1 computes m^2 DFTs of size n^2 so its space complexity is $O(n^2m^2)$. Step 2.1 computes needs a $O(nm)$ array to store all the sums. Step 2.2 needs an $O(p)$ array and Step 2.3 needs a $O(pn^2)$ array. Finally, in Step 3 the results are stored in a $O(pn^2)$ array. Therefore the total complexity if $n \gg m$ is $O(\max(p, m^2)n^2)$. In the typical case that $p = m^2$ the space complexity is the same as the space complexity of the input plenoptic image.

4. 3D RECONSTRUCTION USING THE FOCAL STACK

Once the focal stack is built we will use Depth-from-focus (DFF) techniques [7] to recover depth. DFF is a 3D reconstruction method, based on applying a focus measure to a set of differently focused photographs. The use of a plenoptic camera allows us to obtain several focused images from a single lightfield image. A variety of focus measures have been proposed, in this paper we use a Laplacian based focus measure FM :

$$(FM [P_\alpha^d [L_F^d]]) (\alpha \mathbf{x}) = -|(P_\alpha^d [L_F^d] * D_{x_1}^2)(\alpha \mathbf{x})| - |(P_\alpha^d [L_F^d] * D_{x_2}^2)(\alpha \mathbf{x})|, \quad (8)$$

Convolving each photograph in the focal stack with smoothed Laplacian kernels in the x_1 and x_2 directions. This collection of laplacian convoluted focal stack photographs is the laplacian focal stack (LFS) [5]. These convolutions can be efficiently implemented as a product in the frequency domain modifying Step 1 in all DFST-based algorithms as follows:

Step 1

For every $\hat{\mathbf{u}}$ compute $S1_{x_i}(\hat{\mathbf{u}}, \hat{\mathbf{u}}) = (F_{N_x^{ext}}^{1,1} [L_F^{d,ext}(\hat{\mathbf{x}}, \hat{\mathbf{u}})])(\hat{\mathbf{u}}, \hat{\mathbf{u}}) (F_{N_x^{ext}}^{1,1} [D_{x_i}^2])(\hat{\mathbf{u}})$,

$$F_{N_x^{ext}}^{1,1} [D_{x_i}^2](\hat{\mathbf{a}}) \propto \exp(-(\hat{a}_1^2 + \hat{a}_2^2)/\sigma^2) \hat{a}_i^2, i=1,2.$$

The algorithm proceeds as before obtaining $\hat{S}_{x_i}(\hat{\mathbf{x}}, \hat{\mathbf{p}})$ $i=1,2$ and we finally compute the focus measure $FM(\hat{\mathbf{x}}, \hat{\mathbf{p}}) = -(|\hat{S}_{x_1}(\hat{\mathbf{x}}, \hat{\mathbf{p}})| + |\hat{S}_{x_2}(\hat{\mathbf{x}}, \hat{\mathbf{p}})|)$. We could greedily optimize the focus measure for each pixel independently: $\hat{\mathbf{p}}(\hat{\mathbf{x}}) = \arg \min_{\hat{\mathbf{p}}} FM(\hat{\mathbf{x}}, \hat{\mathbf{p}})$ but it gives better results to construct a prior favouring surface smoothness and to solve a regularized version of DFF. We use the LFS and the Markov Random Field (MRF) approach [8] to obtain the optimal depth minimizing an energy function E [5]. This energy function is composed of a data energy E_d and smoothness energy E_s , $E = E_d + \lambda E_s$, where the parameter λ measures the relative importance of each term. The data energy is the sum of the per-pixel data costs $E_d = \sum_{\hat{\mathbf{x}}} FM(\hat{\mathbf{x}}, \hat{\mathbf{p}}(\hat{\mathbf{x}}))$. To define the smoothness energy we use a 4-connected neighborhood system, so the smoothness energy E_s can be written $E_s = \sum_{\hat{\mathbf{x}}, \hat{\mathbf{y}}} V_{\hat{\mathbf{x}}, \hat{\mathbf{y}}}(\hat{\mathbf{p}}(\hat{\mathbf{x}}), \hat{\mathbf{p}}(\hat{\mathbf{y}}))$ where $\hat{\mathbf{x}}$ and $\hat{\mathbf{y}}$ are 4-connected pixels. We use $V_{\hat{\mathbf{x}}, \hat{\mathbf{y}}}(\hat{\mathbf{p}}(\hat{\mathbf{x}}), \hat{\mathbf{p}}(\hat{\mathbf{y}})) = \min(\mu, |\hat{\mathbf{p}}(\hat{\mathbf{x}}) - \hat{\mathbf{p}}(\hat{\mathbf{y}})|^\nu)$ and optimize the energy E using belief propagation [8].

5. EXPERIMENTAL RESULTS

In this section we present some experimental results from our space-efficient fast DFST. We have used several real plenoptic images shown on figures 3-6. Intensity fall-off effects on the edges of microlens were compensated using a histogram-based non-linear transform. Degraded pixels in the border of each microlens were discarded. For each plenoptic image we show the image focused on several planes and the 3D depth for the elements in the scene.



Figure 3. (Left and center) Subset of images from the focal stack obtained with our algorithm. (Right) 3D reconstruction image where depth is presented as a grey color. The size of the plenoptic image is $n=291$ and $m=11$.



Figure 4. (Left and center) Subset of images from the focal stack obtained with our algorithm. (Right) 3D reconstruction image where depth is presented as a grey color. The size of the plenoptic image is $n=291$ and $m=9$.



Figure 5. (Left and center) Subset of images from the focal stack obtained with our algorithm. (Right) 3D reconstruction image where depth is presented as a grey color. The size of the plenoptic image is $n=249$ and $m=9$.



Figure 6. (Left and center) Subset of images from the focal stack obtained with our algorithm. (Right) 3D reconstruction image where depth is presented as a grey color. The size of the plenoptic image is $n=512$ and $m=17$.

Comparisons of the results in Figures 3-4 with the DFST in [5] show the increment in depth resolution due to the change from $O(m)$ to $O(m^2)$ planes. In Figure 5 we show a plenoptic image taken with a different plenoptic camera than figures 3-4. In Figure 6 we show the results from a plenoptic image taken with a camera gantry with $n=512$ and $m=17$ leading to 0.22 Gb. plenoptic image that was also successfully processed with our algorithm. It can be seen that the algorithm is able to focus on any desired plane in the scene and also recovers the 3D depth.

6. CONCLUSIONS

We have presented a new fast algorithm that decreases space complexity of the Fast DFST from $O(n^3m)$ to $O(n^2m^2)$ maintaining its computational complexity. This leads in typical situations to a 60 times reduction in the memory requirements of the Fast DFST. This allows the computation of the focal stack of real plenoptic images in RAM memory which leads to a substantial reduction in the time. The use of depth from focus techniques allows us to recover depth using a single plenoptic image from the focal stack built with our algorithm. Future improvements will consist on porting the technique to GPUs and FPGAs.

ACKNOWLEDGEMENT

The authors wish to thank Ren Ng, Stanford C. G. Lab. and Paolo Favaro for their lightfield data.

REFERENCES

- [1] F. Ives. Patent US 725,567. 1903. 1
- [2] G. Lippmann. Epreuves reversibles donnant la sensation du relief. *Journal of Physics*, 7(4):821–825, 1908.
- [3] T. Adelson and J. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pages 99–106, 1992.
- [4] R. Ng, M. Levoy, M. Bredif, G. Duval, M. Horowitz, et al. Light field photography with a hand-held plenoptic camera. *Computer Science Technical Report CSTR*, Jan 2005.
- [5] F. Pérez, J.G. Marichal, and J.M. Rodríguez-Ramos, The Discrete Focal Stack Transform, *Proc. of the European Signal Processing Conference*, 2008.
- [6] F. Pérez, J. Philip Lüke, An $O(n^2 \log n)$ per plane fast discrete focal stack transform, *Optical 3D Measurement Techniques*, 2009.
- [7] S.K. Nayar Y. Nakagawa, Shape from focus, *IEEE trans. on PAMI* Vol. 16 n° 8 pp: 824 - 831, 1994.
- [8] D. Scharstein and R. Szeliski. A taxonomy and evaluation of dense two-frame stereo correspondence algorithms, *International Journal of Computer Vision*, 47(1/2/3):7-42, 2002.

3D RECONSTRUCTION OF SMALL SOLAR SYSTEM BODIES USING PHOTOCLINOMETRY BY DEFORMATION

Claire Capanna^{1,2}, Laurent Jorda¹, Philippe Lamy¹ and Gilles Gesquiere²

¹*Laboratoire d'Astrophysique de Marseille - 38 rue Joliot Curie, 13388 Marseille Cedex 13, France*

²*Laboratoire des Sciences de l'Information et des Systèmes - rue R. Follereau, BP 90178, 13637 Arles Cedex*

ABSTRACT

This article considers the problem of reconstructing the 3D shape model of asteroids and cometary nuclei from images obtained with an imaging system on board a spacecraft. We present a photogrammetry method based on the minimization of the chi-square difference between observed and synthetic images of the object by deformations of its initial shape. The minimization is performed using the so-called "limited-memory Broyden-Fletcher-Goldfarb-Shanno" algorithm. The deformations can be applied: (i) by modifying the coefficients of a spherical harmonic expansion in order to extract the global shape of the object, and/or (ii) by moving the height of the vertices of a triangular mesh in order to increase the accuracy of the global shape model and/or to derive localized topographic maps of the surface. This method has been tested on images of the asteroids Steins and Lutetia observed by the imaging system on board the Rosetta spacecraft of the European Space Agency.

KEYWORDS

3D reconstruction, optimization, deformation, synthetic images, photogrammetry

1. INTRODUCTION

There are several ways to reconstruct a three-dimensional surface from the grey levels measured in the pixels of an image. The stereo technique allows to build a network of control points from remarkable features identified at the surface of the object [8]. Stereo-photogrammetry, an extension of the former, is a powerful technique widely used in the past to reconstruct digital terrain models, for instance that of planets [3] or that of the nuclei of comets [6]. More recently, stereophotogrammetry using several views of the same area of the surface under different viewing and/or illumination conditions has been successfully applied to several bodies of our solar system, among them the asteroid Itokawa observed by the Hayabusa spacecraft [2].

We present here a new photogrammetry method in which we apply deformations to a triangular mesh in a non-linear optimization loop until the synthetic images resulting from the mesh best match the observed ones. The article is organized as follows: we present our method in Section 2, its application to asteroids Steins and Lutetia in section 3 and we finally conclude in Section 4.

2. 3D RECONSTRUCTION METHOD

Our 3D reconstruction method needs an input model described as a mesh of triangular facets. This model can be a sphere or a more refined model obtained with another reconstruction method.

2.1 Generation of Synthetic Images

Synthetic images are generated using a tool called OASIS (Optimized Astrophysical Simulator for Imaging Systems) [5]. OASIS calculates the position and the orientation of the object in the camera frame. It then performs ray-casting to determine which facets of the shape model are illuminated and in view of the camera.

For each such facet i and each image n , it calculates the bi-directional reflectance (BDR) $r_i^{(n)}$ following Hapke's model [4]. The intersections $\Omega_i^{(p,q,n)}$ between this facet and the pixel (p, q) are calculated in terms of solid angle. The signal $D_{pq}^{(n)}$ received by each pixel (in DN) is then estimated by summing the contributions of all facets in the field-of-view of this pixel:

$$D_{pq}^{(n)} \approx \frac{g S_{coll} t_e^{(n)}}{hc (R_h^{(n)})^2} \left(\sum_{i=1}^{M_{pq}} r_i^{(n)} \Omega_i^{(p,q,n)} \rho_i \right) \quad (1)$$

where g is the gain of the electronics, S_{coll} is the collecting surface of the telescope, $t_e^{(n)}$ is the exposure time of image n , $R_h^{(n)}$ is the heliocentric distance of the object (in astronomical units), h is Planck's constant, c is the light speed, ρ_i is a pre-calculated absolute calibration factor. Finally, the image is convolved by the point-spread-function of the instrument.

2.2 Deformations of the Shape Model

2.2.1 Deformation based on Spherical Harmonics

Spherical parametrization of a triangular mesh shape model has been developed in the last three decades. The spherical harmonic transform [10] is used to decompose the input model in the frequency domain. The coordinates \mathbf{R}'_k of the vertices are described as:

$$\mathbf{R}'_k = R_k \begin{pmatrix} \sin \theta_k \cos \varphi_k \\ \sin \theta_k \sin \varphi_k \\ \cos \theta_k \end{pmatrix} \quad (2)$$

where (θ_k, φ_k) defines the vertices of a pre-defined triangular mesh in a spherical coordinates system and R_k is given by:

$$R_k = \sum_{l=0}^{l_{max}} \sum_{m=-l}^{+l} C_{lm} Y_{lm}(\theta_k, \varphi_k) \quad (3)$$

$Y_{lm}(\theta, \varphi)$ are the real form of the spherical harmonic functions. In this representation, the parameters C_{lm} define the shape of the object. The number of coefficients depends on the degree l_{max} of the above expansion. Increasing its value allows us to get higher frequencies in the 3D representation of the object.

2.2.2 Deformation based on Vertex Offsets

In order to increase the accuracy of the spherical harmonic model or of the input model, we introduce another deformation scheme in which we directly modify the height of the vertices with respect to the initial mesh used as starting points in the optimization process. We calculate the vector \mathbf{N}_k normal to the surface at the vertex k by averaging the normal vectors \mathbf{N}_i of the V_k facets which share this vertex:

$$\mathbf{N}_k = \frac{\sum_{i=1}^{V_k} S_i \mathbf{N}_i}{\sum_{i=1}^{V_k} S_i} \quad (4)$$

where S_i is the surface of the facet number i .

The modification of the height H_k of the vertex k is applied in the direction \mathbf{N}_k . The coordinates \mathbf{R}'_k of the vertex after this deformation become:

$$\mathbf{R}'_k = \mathbf{R}_k + H_k \mathbf{N}_k \quad (5)$$

The coefficients H_k are initially set to zero and their value is modified during the optimization.

2.3 Optimization of the Parameters

2.3.1 Shape Model

We want to minimize the reduced chi-square between the pixel values $F_{pq}^{(n)}$ of the N observed images and those of the synthetic images given by Eq. (1):

$$\chi^2(P_k) = \frac{1}{N_p} \sum_{n=1}^N \sum_{pq} \frac{\left(F_{pq}^{(n)} - D_{pq}^{(n)}(P_k)\right)^2}{(\sigma_{pq}^{(n)})^2} \quad (6)$$

where N_p is the total number of pixels used in the minimization process, and $\sigma_{pq}^{(n)}$, the uncertainty on the observed pixel value, is given by:

$$\sigma_{pq}^{(n)} = g \sqrt{\frac{D_{pq}^{(n)} t_e}{g} + \sigma_R^2} \quad (7)$$

where σ_R is the readout noise. In Eq. (6), the free parameters (variables) P_k are the coefficients C_{lm} when the shape model is defined as an expansion in spherical harmonics (section 2.2.1) and the heights H_k when we modify the coordinates of the vertices (section 2.2.2).

In order to minimize the chi-square function (6), we use a non-linear optimization algorithm called “limited memory Broyden-Fletcher-Golbfarb-Shanno” (L-BFGS), a quasi-Newton optimization method [1]. It is well suited to large scale optimization problems and requires a limited amount of memory. A typical number of $\sim 50 - 100$ iterations of the algorithm is required before it converges to a stable value P_k of the free parameters. When the final parameters are reached, the coordinates \mathbf{R}'_k of the vertices form the final shape model.

The L-BFGS method requires the calculation of the partial derivatives at each iteration. These derivatives are calculated with the finite difference method, in which two chi-squares are calculated with the Eq. (6) for each partial derivative:

$$\frac{\partial \chi^2}{\partial H_l}(H_k) = \frac{\chi^2(H_{k'}, H_l + \epsilon_H) - \chi^2(H_{k'}, H_l - \epsilon_H)}{2 \cdot \epsilon_H} \quad (8)$$

where the index $k' = 1, \dots, l-1, l+1, \dots, N_v$, where N_v is the number of vertices.

We emphasize that a major part of the CPU time is used for the calculation of these derivatives. We will see in section 2.4 how these partial derivatives can be calculated much faster, thus allowing us to speed up considerably the optimization process.

2.3.2 Local Error Calculation

For each pixel (p, q) of the image number n , we calculate the residual value in units of the instrumental noise at the end of the optimization using the notation of section 2.3.1:

$$\mu_{pq}^{(n)} = \frac{F_{pq}^{(n)} - D_{pq}^{(n)}(P_k)}{\sigma_{pq}^{(n)}} \quad (9)$$

We “project” the value of the residuals from the pixels to the facets using the pixel–facets intersecting solid angles $\Omega_i^{(p,q,n)}$ calculated earlier in section 2.1:

$$\mu_i = \frac{1}{N \Omega_P} \sum_{n=1}^N \sum_{pq} \Omega_i^{(p,q,n)} \mu_{pq}^{(n)} \quad (10)$$

where Ω_P is the total pixel solid angle, and the second sum runs over the pixels (p, q) intersecting the facet number i . We then calculate the effect of a variation of the slope of each facet on the measured signal, normalized to its associated instrumental noise. The derivative $dD_i/d\varepsilon$ (in DN/°) of the signal with respect to a change of slope is obtained numerically by calculating the mean variation of the signal $D_i^{(n)}$ from Eq. (1) when the vector normal to each facet remains on a cone of axis \mathbf{N}_i - the normal to the surface of the facet – and of half cone aperture ε . The error ξ_i on the slope of each facet is deduced from μ_i , $dD_i/d\varepsilon$, and from the instrumental noise σ_i associated to the signal D_i (Eq. 7):

$$\xi_i = \frac{|\mu_i| \sigma_i}{dD_i/d\varepsilon} \quad (11)$$

We take a typical value of $\varepsilon \sim 1^\circ$ to estimate the value of the derivative $dD_i/d\varepsilon$. Knowing the error on the slope, the error on the heights (parameters H_k) can be easily deduced. The parameter ξ_i forms what we call the “slope error map”.

2.3.3 Additional Parameters

Additional parameters can be optimized in the same way if they have an impact on the synthetic images, for instance:

- the parameters which describe the BDR of the surface,
- the three angles describing the pointing direction and roll angle of the camera for each individual image,
- the three Euler angles describing the orientation of the object in space.

The accuracy in the reconstruction of these parameters by the space agencies is usually not sufficient to successfully achieve the optimization process. Therefore, we always need to perform iterative optimizations of the shape, of the pointing direction and roll angle of the camera.

2.4 Faster Calculation of the Partial Derivatives

2.4.1 General Principle of the Method

As explained in section 2.3.1, most of the CPU time during the optimization process is used to calculate the partial derivatives of Eq. (8). In this section, we describe a method that allows to calculate them faster, accelerating considerably the whole reconstruction process. The relationship (8) implies the calculation of two chi-squares, i.e. the calculation of $2N$ synthetic images for each partial derivative. We briefly describe in this section how we can perform the same task recalculating only the pixel values of the images which are modified when we move the height of a vertex from H_l to $H_l \pm \varepsilon_H$ (section 2.4.2) and to extract from these values the final partial derivative of the chi-square (section 2.4.3).

2.4.2 Calculation of Updated Pixel Values

The calculation of the pixel values modified by a change of the height of the vertex number l from its original value H_l to $H_l \pm \varepsilon_H$ is performed in the following steps.

1) We identify the facets $F_l^{(n)}$ using the vertex number l . We determine the nominal set of pixels $S_l^{(n)}$ intersected by the facets. We then determine a new set of pixels $S_{l,\pm}^{(n)}$ intersected by the facets after the height of the vertex number l has been modified. The set of pixels potentially modified by a displacement of this vertex is the union of both samples: $P_l^{(n)} = S_l^{(n)} \cup S_{l,\pm}^{(n)}$.

2) We re-determine if the facets intersected by the pixels $P_l^{(n)}$ are illuminated and visible from the observer taking into account the displacement of the vertex.

3) For all the facets $F_l^{(n)}$ which are both illuminated and in view from the observer, we re-calculate the BDR $r_l^{(n)}$ taking into account the new geometry.

4) The updated values $D'_{pq}{}^{(n)}(H_k)$ of the pixels $P_l^{(n)}$ are then calculated using the relationship (1).

We repeat this operation for all images ($n = 1..N$).

2.4.3 Chi-square Partial Derivatives Calculation

Let us introduce the contribution of the image number n to the global chi-square of Eq. (6):

$$\gamma^{(n)} = \sum_{pq} \Delta_{pq}^{(n)} \quad (12)$$

where:

$$\Delta_{pq}^{(n)} = \frac{\left(F_{pq}^{(n)} - D_{pq}^{(n)}(H_k) \right)^2}{(\sigma_{pq}^{(n)})^2} \quad (13)$$

The value of the function $\Delta'_{pq}{}^{(n)}$ corresponding to the updated pixel values $D'_{pq}{}^{(n)}(H_k)$ calculated in section 2.4.2 is calculated in the same way:

$$\Delta'_{pq}{}^{(n)} = \frac{\left(F_{pq}^{(n)} - D'_{pq}{}^{(n)}(H_k) \right)^2}{(\sigma_{pq}^{(n)})^2} \quad (14)$$

for $pq \in P_l$. We now call $A^{(n)}$ the set of all pixels of image number n . After modification of the height of a vertex, the pixels which remain unchanged belong to $C_l^{(n)}$. Using these notations, we have $A^{(n)} = C_l^{(n)} \cup P_l^{(n)}$. With these notations, the modified chi-square can be written as:

$$\chi^2 = \frac{1}{N_p} \sum_{n=1}^N \left[\sum_{pq \in C_l^{(n)}} \Delta_{pq}^{(n)} + \sum_{pq \in P_l^{(n)}} \Delta'_{pq}{}^{(n)} \right] \quad (15)$$

which can be rewritten:

$$\chi^2 = \frac{1}{N_p} \sum_{n=1}^N \left[\left(\gamma^{(n)} - \sum_{pq \in P_l^{(n)}} \Delta_{pq}^{(n)} \right) + \sum_{pq \in P_l^{(n)}} \Delta'_{pq}{}^{(n)} \right] \quad (16)$$

This relationship gives us the expression of the chi-square as a function of three parameters. The first parameter, $\gamma^{(n)}$, is calculated from the nominal image before the calculation of the partial derivatives. The parameters $\Delta_{pq}^{(n)}$ and $\Delta'_{pq}{}^{(n)}$ are recalculated knowing the sets of modified pixel $P_l^{(n)}$ of all images and the new pixel values $D'_{pq}(H_k)$.

The partial derivatives can be calculated with the finite difference of the new chi-square:

$$\frac{\partial \chi^2}{\partial H_l}(H_k) = \frac{\chi^2(H_{k'}, H_l + \epsilon_H) - \chi^2(H_{k'}, H_l - \epsilon_H)}{2 \cdot \epsilon_H} \quad (17)$$

This calculation is simplified since only the sums of contributions $\Delta_{pq}^{(n)}$ and $\Delta'_{pq}{}^{(n)}$ have to be calculated:

$$\begin{aligned} \frac{\partial \chi^2}{\partial H_l}(H_k) = \frac{1}{2 \cdot N_p \cdot \epsilon_H} \sum_{n=1}^N \left[\sum_{pq \in P_{l,+}^{(n)}} \left(\Delta_{pq}^{(n)}(H_{k'}, H_l + \epsilon_H) - \Delta_{pq}^{(n)}(H_{k'}, H_l - \epsilon_H) \right) \right. \\ \left. - \sum_{pq \in P_{l,-}^{(n)}} \left(\Delta'_{pq}{}^{(n)}(H_{k'}, H_l - \epsilon_H) - \Delta'_{pq}{}^{(n)}(H_{k'}, H_l + \epsilon_H) \right) \right] \quad (18) \end{aligned}$$

The gain in CPU time in the calculation of the partial derivatives between the relationships (8) and (18) is then given by the ratio between the total number of pixels of the images and the typical number of pixels in all the set $P_l^{(n)}$, i.e., typically 4 pixels per image. The gain is therefore of the order of $N_p / (4N)$.

3. APPLICATIONS TO ASTEROIDS STEINS AND LUTETIA

3.1 Observations

The Rosetta spacecraft [7] launched in 2004 by the European Space Agency is now on its way to meet its final target, the nucleus of comet P/Churyumov-Gerasimenko. During its cruise, Rosetta flew by two asteroids: Steins in September 2008 and Lutetia in July 2010. Images of these two asteroids have been acquired by OSIRIS, the imaging system on board Rosetta. OSIRIS includes a narrow and a wide-angle camera both equipped with the same 2048^2 pixels E2V 42-40 CCD detector offering pixel field-of-views of respectively 18.9 and 99.5 μ rad.

Rosetta flew by Steins at a minimum distance of 802 km, with a phase angle which varied from 38 to 0 and then increased to almost 140 out-bound. For the following analysis, we use a set of 8 geometrically calibrated images: one image acquired with the narrow-angle camera from a distance of 5235 km and seven images acquired with the wide-angle camera from a distance of 1120 km in-bound (see Fig. 1) to 865 km out-bound. These images correspond to a range of phase angles from 7 to 73.

Lutetia was flown by at a minimum distance of 3170 km, with a phase angle which varied from 10 to 0 and then increased to almost 140 out-bound. For the following preliminary tests, we use a set of 4 small images extracted from geometrically calibrated images acquired with the narrow-angle camera from a distance of 5200 km in-bound (see Fig. 3) to 3670 km out-bound. These images correspond to a range of phase angles from 26 to 110.

3.2 Steins Reconstruction

3.2.1 Reconstruction Strategy

The shape reconstruction of Steins is performed in two main steps. The first step consists in calculating a low-resolution spherical harmonic model. For this, we start from a sphere, for which $C_{00} = R_a = 2.7 \text{ km}$ and the other parameters are set to zero. We use the Hapke parameters [4] describing the BDR derived from a global photometric analysis. We begin optimizing the C_{lm} spherical harmonic coefficients of the shape model with $l_{max} = 2$. We iteratively optimize the shape and the Euler angles describing the pointing of the camera. When a stable solution is reached, typically after 3-5 iterations, we set the degree of the expansion to $l_{max} = 4$ and we repeat the cycle of shape and Euler angles optimizations until $l_{max} = 20$. The sampling of the shape model is given by a hierarchical triangular mesh [9] of level five built from an icosahedron. The resulting 20480 facets and 10242 vertices ensure that the pixels of the images contain typically a few facets. At the end of this step, we have a spherical harmonic model of the asteroid. The corresponding synthetic image is given in Fig. 1.

In a second step, we optimize the heights of the vertices in order to improve the spherical harmonic shape model calculated at the previous step. We proceed as before by iteratively optimizing the shape model and the Euler angles. After a total of 3 iterations, the final shape model of the asteroid is obtained (see Fig. 1). We keep the same sampling of the surface used to derive the spherical harmonic shape model

The results given in this section were obtained with the “slow” version of chi-square partial derivative calculation, which corresponds to the Eq. (8).

3.2.2 Final Shape Model

Table 1 summarizes our results. The mean residuals are expressed in units of the instrumental noise. The final mean slope error is calculated from the residuals (section 2.3.2). Note that, for the spherical harmonic optimization, only one iteration between pointing and shape optimization is needed.

Table 1. Summary of our results for spherical harmonic Optimization (SHO) and Vertices Height Optimization (VHO)

Parameters	SHO	VHO
Starting chi-square	28640	90
Final chi-square	90	37
Mean residuals	9.5	6.1
Mean slope error	12°	7°
Mean height error	20 m	10 m
Number of iterations	1	3
Total CPU time	30 hours	3 weeks

The CPU times are given for a dual-core 2 GHz Opteron CPU with 2 GB of RAM. The mean residuals decrease from 9.5 to 6.1 (in units of instrumental noise) from the spherical harmonic to the triangular mesh representation (see also Fig. 1).

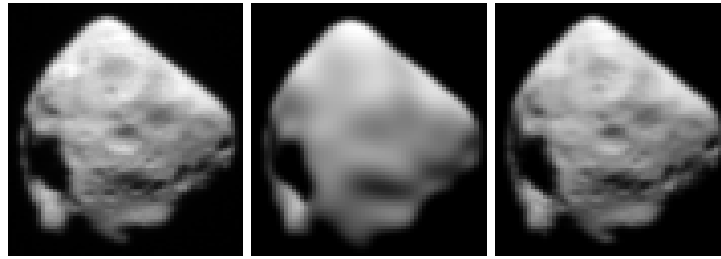


Figure 1. Illustration of the reconstruction methods for one of the eight images of asteroid Steins. Left panel: Observed image. Center panel: synthetic image calculated from spherical harmonic model. Right panel: synthetic image calculated from the final optimized model

3.2.3 Accuracy

We present in Fig. 2 the histogram of the values in the “slope error map” defined in section 2.3.2 for both the spherical harmonic model and the final triangular mesh model. For the latter, the histogram peaks at 7, which corresponds to 10 m in height, about 1/8th of the pixel resolution at closest approach.

Larger values of up to $\sim 30^\circ$ are however obtained in regions of the shape model which correspond to limbs or terminators on the images.

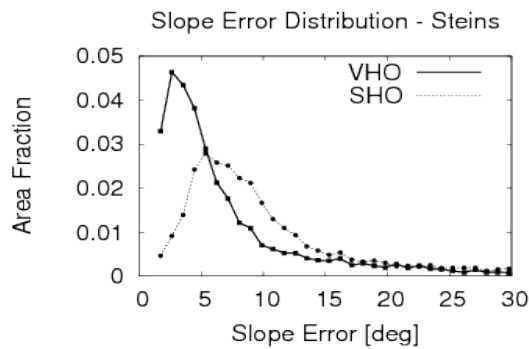


Figure 2. Slope Error Histogram of the models of Steins.

3.3 Preliminary Tests on Lutetia

Figure 3 shows the result of preliminary tests performed on a Digital Terrain Model of the asteroid Lutetia applying deformation on the height of the vertices. Our input model is a “maplet” (DTM) extracted from the Stereophotoclinometry model of R. Gaskell [2] calculated with his software LITHOS. Our method allows us to recover high-frequency information (right panel) compared to the initial DTM (middle panel). Craters and grooves are reconstructed with a more accurate depth.

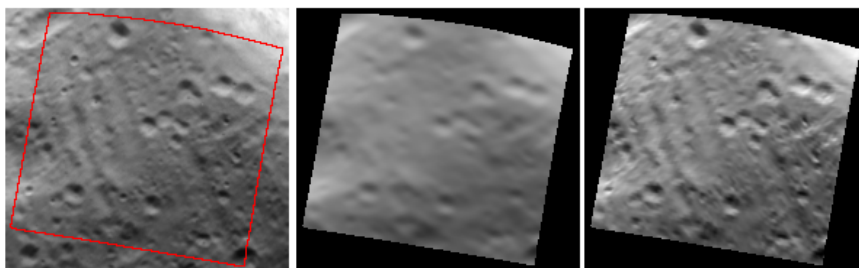


Figure 3. Illustration of the reconstruction method used on asteroid Lutetia. Left panel: Observed image. Center panel: synthetic image calculated from LITHOS model. Right panel: synthetic image calculated from the final optimized model

4. CONCLUSION AND FUTURE WORK

We developed and tested a new multi-image photogrammetry method based on deformations of a three-dimensional shape model, on images of the asteroid Steins obtained by the imaging system on board ESA's Rosetta spacecraft. The method also generates a map of the local topographic error deduced from the pixel residuals, both in slopes and in heights. In the case of Steins, the local averaged slope error amounts to 7° .

The "faster" version of the code for the calculation of the partial derivatives of the chi-square described in section 2.4 has been implemented in the code and is in the test and validation process. The first results indicate an overall gain in speed which is in agreement with the prediction, i.e., would correspond a factor of about 1000 for the example described in section 3, therefore reducing the total CPU time to 30 min.

In the near future, we intend to improve the speed and the robustness of the method by implementing it in a multi-resolution approach. We will also apply this technique to the Lutetia images in order to improve existing shape models of this asteroid. Finally, we will apply it to the images of the asteroid Vesta acquired by the DAWN spacecraft, and later on to the high-resolution images of the nucleus of comet 67P/Churyumov-Gerasimenko acquired by the scientific cameras of Rosetta.

REFERENCES

- [1] Byrd, R.H. Et al., 1994, Representation of quasi-newton matrices and their use in limited memory methods, *Mathematical Programming* , Vol. 63, No. 4, pp. 129–156, 1994.
- [2] Gaskell, R.W. et al., 2006, Landmark navigation studies and target characterization in the hayabusa encounter with itokawa, *AIAA, Astrodynamics Specialists Conference and Exhibit*.
- [3] Gwinner, K. et al., 2007, Derivation and validation of high resolution digital terrain models from mars express HRSC-Data, *Photogramm. Eng. Remote Sens.*
- [4] Hapke, B., 2002, Bidirectional reflectance spectroscopy 5. the coherent backscatter opposition effect and anisotropic scattering, *Icarus* , vol. 157, pp. 523-534.
- [5] Jorda, L. et al., 2010, OASIS: a simulator to prepare and interpret remote imaging of solar system bodies, *Proc. SPIE*, Vol. 7533, No. 753311.
- [6] Oberst, J. et al., 2004, The nucleus of Comet Borrelly: a study of morphology and surface brightness, *Icarus*, Vol. 167, pp. 70-79.
- [7] Schulz, R., 2009, Rosetta - one comet rendezvous and two asteroid fly-bys, *Solar System Research* , Vol. 43, No. 4, pp. 343-352.
- [8] Simonelli, D.P. Et al., 1993., The generation and use of numerical shape models for irregular solar system objects, *Icarus* , Vol. 103, pp. 49-61.
- [9] Szalay, A. et al., 2005, Indexing the sphere with the hierarchical triangular mesh, *Tech. Rep. MSR-TR-2005-123, Microsoft Research*
- [10] Wiebicke, H.J., 1989, A method for modelling the surface of irregular celestial bodies, *Astron. Nachr.* , Vol. 310, No.2, pp. 159–174.

POINT-BASED SIMILARITY ESTIMATION BETWEEN 2.5D VISUAL HULLS AND 3D OBJECTS

Konstantinos Moustakas, Georgios Stavropoulos and Dimitrios Tzovaras
*Centre for Research and Technology Hellas – Informatics and Telematics Institute
 6th km Charilaou-Thermi Road, P.O.Box 60361, 57001 Thessaloniki, Greece*

ABSTRACT

This paper presents a novel framework for point-based similarity estimation between 3D objects and 2.5D visual hulls. Initially, the protrusion map is estimated for both the visual hull that is generated by a range image and the 3D model that is followed by the extraction of the salient features that correspond to the highly protruding areas of the objects. Then, based on the concept that for a 3D object and a corresponding query range image, there should be a virtual camera with such intrinsic and extrinsic parameters that would generate an optimum range image, in terms of minimizing an error function that takes into account the visual hull and the salient features of the objects, when compared to other parameter sets or other target 3D models, matching is performed via estimating dissimilarity within the range image and salient feature space. Experimental results illustrate the efficiency of the proposed approach in benchmark datasets.

KEYWORDS

Range image, 3D object classification, protrusion map, salient features, visual hull.

1. INTRODUCTION

The problem of full 3D object similarity estimation, registration, search and retrieval has been successfully addressed by many researchers in the past, while excellent and extensive surveys can be found in (Tangelder, 2008) and (Bustos, 2005).

1.1 Related Work

Focusing on the problem of recognizing a 3D object when only a part of its shape is available as query few approaches have been presented in the past. Reeb Graphs are topological and skeletal structures that are used as a search key that represents the features of a 3D shape (Hilaga, 2001), (Bespalov, 2003). In (Biasotti, 2003) Reeb graphs that are obtained by using different quotient functions are obtained and highlight how their choice determines the final matching result. Other commonly used methods for 3D matching, that also support partial matching use *Local features* as described in (Gal, 2006) and (Shilane, 2007). Finally, partial matching, can be also achieved with the use of *model graphs* (Cicirello, 2001), (El-Mehalawi, 2003) or point-based approaches (Ruggeri, 2010), (Stavropoulos, 2010). In (Macrini, 2002), utilizing shock graph matching, indexing using topological signature vectors is applied to implement view based similarity matching more efficiently. Moreover, in (Chen, 2003) the light field descriptor is presented that is based on the concept that if two objects correspond, then they should also correspond from every viewpoint. A similar approach is presented in (Vranic, 2004), the so called "depth buffer". However, these approaches do not deal with partial matching. Germann et al. (Germann, 2007) initially precalculate a number of range images from different points of view. Shum et al. (Shum, 1996) map the surface curvature of 3D objects to the unit sphere with the use of a spherical coordinate system. By searching over a spherical rotation space, a distance between two curvature distributions is computed and used as a measure for the similarity of two objects. Finally sketch-based approaches have also been developed for the quick sketch of a partial primitive shape to be used as query in 3D search system (Moustakas, 2006), (Moustakas, 2009).

1.2 Motivation

The aforementioned approaches have a few drawbacks. In many cases, they use a priori information for registering the partial view with the complete 3D model, while they do not utilize the saliency information of the 3D objects. In the proposed method, salient features are generated based on the protrusion maps of 3D models. The protrusion maps provide a stochastic saliency-based representation of the 3D models that is seen experimentally to efficiently encode object shape and exhibit robustness to noise and occlusion. The proposed scheme is experimentally seen to be robust and efficient in benchmark datasets.

2. 3D OBJECT REPRESENTATION

The identification of similar areas is computationally very expensive if matching is performed by comparing the query range image with another range image extracted from the 3D model. Therefore, a subset of features for the range image and the 3D model should eventually be used that should be easy to handle and representative for each object. In the present paper, salient features are used, that lie in general in the most protruding areas of a 3D surface.

The 3D models of most databases are in general in various scales. In order to be able to easily compare the similarities between a query range image and a set of models, the models should be normalized to a common scale. Since the spherical coordinate system is used in the current scheme, the best choice would be to normalize each model to the *unit sphere* before extracting the salient points of the 3D models. It should be noticed that the database models are normalized to a common scale so as to ease the implementation of the matching algorithm, even if the normalization is not a crucial process.

2.1 Protrusion Maps

The developed method for salient feature extraction that correspond to sharp locally protruding areas of the object's surface (Moustakas, 2007), (Lin, 2004) is based on Hoffman and Singh's theory of salience (Hoffman, 1997). A brief description of the method follows.

Initially, the dual graph $G=(V,E)$ of the given triangulated surface is generated, where V and E are the dual vertices and edges. A dual vertex is the center of mass of a triangle and a dual edge links two adjacent triangles. The degree of protrusion for each dual vertex results from the following equation:

$$p(\mathbf{u}) = \sum_{i=1}^N g(\mathbf{u}, \mathbf{v}_i) \cdot \text{area}(\mathbf{v}_i) \quad (1)$$

Where N is the number of dual vertices in the entire surface, $p(\mathbf{u})$ is the protrusion degree for the dual vertex \mathbf{u} , $g(\mathbf{u}, \mathbf{v}_i)$ is the geodesic distance of \mathbf{u} from dual vertex \mathbf{v}_i and $\text{area}(\mathbf{v}_i)$ is the area of the triangle corresponding to \mathbf{v}_i .

2.2 Protruding Points

Using simple gradient based methods (i.e. steepest descent) all local maxima of the protrusion map $p(\mathbf{u})$ are obtained. Geodesic windows are then applied and only the global maxima inside the window are considered as salient. A geodesic window, GW , centered at the dual vertex \mathbf{u} is defined as follows:

$$GW_{\mathbf{u}} = \{ \mathbf{v} \mid \forall \mathbf{v} \in V, g(\mathbf{u}, \mathbf{v}) < \varepsilon \} \quad (2)$$

Where ε defines the window size. Consider now the generalization $GGW_{\mathbf{u}}$ of the geodesic window $GW_{\mathbf{u}}$ that refers to the union of all geodesic windows defined on the dual mesh that include \mathbf{u} . The generalized geodesic window is defined as follows:

$$GGW_{\mathbf{u}} = \bigcup GW_{\mathbf{a}} \mid \mathbf{u} \in GW_{\mathbf{a}}, \quad \forall \mathbf{a} \in V \quad (3)$$

Assuming the set of salient features S that is a subset of the set of dual vertices V , then a dual vertex \mathbf{s} is characterized as protruding

$$\mathbf{s} \in S \subset V \quad (4)$$

If and only if the following condition holds:

$$\exists GGW_s \text{ so that } p(\mathbf{s}) = \max(p(\mathbf{u})), \forall \mathbf{u} \in GGW_s \quad (5)$$

2.3 Processing Visual Hulls

In order to extract salient features from the range image, a 3D visual hull should initially be formed. The surface is created using only a subset of the points of the image, so as to reduce the redundancy and size of the triangulated surface to be generated. Features on the range image are selected, as the ones with maximum minimal eigenvalue in a predefined window. After the 3D surface is formed, the salient features are extracted in the same way they are extracted for the 3D models. **Figure 1** illustrates the salient features extracted from the 3D model of an Ant and from its 2D projected range image.

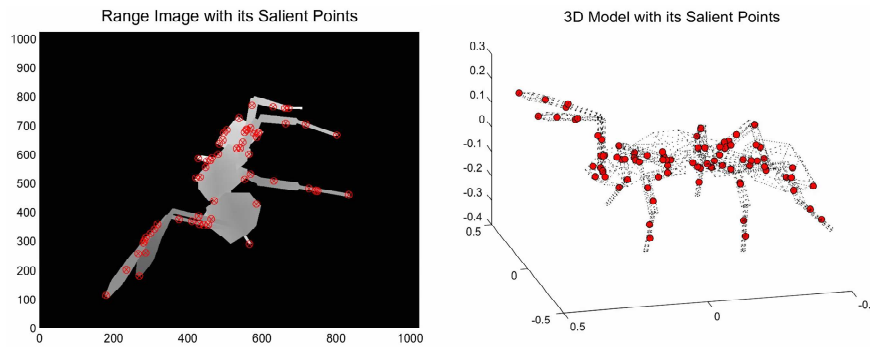


Figure 1. Ant model and the corresponding protruding points for a visual hull (left) and the full 3D model (right).

3. REGISTRATION AND MATCHING

The proposed framework is based on the assumption of a virtual camera, assumed to lie in the space of the examined 3D model. The search in the parameter space is performed for the set of camera parameters that capture a surface as similar as possible to the surface of the query range image.

3.1 Camera Model

Perspective projection using the standard pinhole camera model is adopted in the proposed method. The camera model consists of 10 parameters: resolution (r_w (width), r_h (height)), field of view (f_h (horizontal), f_v (vertical)), radius (ρ), longitude (φ), latitude (θ), roll (r) and z-axis offset (z_0 (yaw), z_φ (pitch)). From the aforementioned parameters, ρ , θ and φ define the position of the camera in space, while r , z_0 and z_φ define the orientation of the virtual camera. Finally, r_w , r_h , f_v and f_h define the internal structure of the camera.

Some of the camera parameters can be estimated prior to the matching process, in order to reduce the dimensionality of the parameter space and thus save computational power. The resolution of the camera is set equal to the resolution of the input image and the focal length can be explicitly estimated. Thus the parameters that need to be estimated are reduced to the following six (ρ , φ , θ , r , z_0 and z_φ)

3.2 Registration using the Visual Hulls

The simplest way to perform matching between the query visual hull and the 3D object is to create a visual hull for every camera in space, for every possible set of parameters and compare it with the query visual hull. Although this method would be in the ideal case perfectly accurate, it is computationally unacceptable.

The most efficient way to search in the parameter space is by using a *spherical coordinate system* in order to describe the camera position. The spherical coordinate system ensures that the virtual camera will "look" at the model which is placed in the center of the coordinate system.

Then, the range image of a particular view is generated using graphics hardware, in particular functions of OpenGL. After the range image is created, it is compared with the query range image by calculating the error for the current camera parameters using the following equation.

$$E(P_i, P_\theta) = \sum_{i,j=0}^{r_w \cdot r_h} \left(I(i, j) - \hat{M}_{P_i, P_\theta}(i, j) \right)^2 \quad (6)$$

Where I corresponds to the query range image, while M refers to the rendered view of the 3D object for a given set of intrinsic and extrinsic virtual camera parameters (P_i, P_e). The set of camera parameters (P_i, P_e) that minimizes equation (6) is considered the best match. It should be noticed that the range image error function includes all the pixels that belong both to the query and the database object's image, only if the common area of both images is larger than a predefined threshold that is set to be 0.75 with respect to the size of the query object. Thus, if less than the 75% of the query object match with the target then this set of camera parameters is considered as non-matching set.

3.3 Registration using the Protruding Points

Even if exhaustive matching is very accurate, it is also computationally very expensive. A way to accelerate the matching process could be to use only a number of feature points of the range image and the 3D model. In the context of the proposed method, salient protruding points are used.

If the virtual camera described in the Section 3.1 exists and the query range image corresponds to the 3D model, then if a set of salient points is extracted from the query range image and another one from the surface that the virtual camera captures, then these two sets of points should also have corresponding subsets of features.

A set of salient points is extracted for the query range image, and another one for the 3D model. The later is transformed for each set of camera parameters (P_i, P_e), in order to be able to project the salient points on the virtual camera plane. These sets of points are used to calculate the similarity between the query range image and the 3D model.

The salient points extracted from the range image (I_i^S) are, in general, not identical to the ones extracted from the 3D model (M_j^S). In order to register the two sets of salient points, the salient points (M_j^S) of the 3D model are projected on the virtual camera plane thus creating a set of 2D salient points ($\hat{M}_j^S(P_i, P_e)$). Each of the projected salient points is registered with a salient point I_i^S , when the Euclidian distance between them V_k , is smaller than a threshold τ_v :

$$V_k(I_i^S, \hat{M}_j^S(P_i, P_e)) = \infty \quad \text{if } V_k > \tau_v \quad (7)$$

and

$$V_k(I_i^S, \hat{M}_j^S(P_i, P_e)) = V_k \quad \text{if } V_k < \tau_v \quad (8)$$

Where τ_v is found empirically to be 3% of the resolution of the image.

Another constraint taken into account is that depth difference between the salient points d_k should also be smaller than a threshold τ_d :

$$d_k(I_i^S, \hat{M}_j^S(P_i, P_e)) = \infty \quad \text{if } d_k > \tau_d \quad (9)$$

and

$$d_k(I_i^S, \hat{M}_j^S(P_i, P_e)) = d_k \quad \text{if } d_k < \tau_d \quad (10)$$

A protruding point from the range image is registered with one from the 3D model only if both constraints described in equations (7)-(10) are satisfied.

More specifically the distance function E_S used to calculate the difference between the two sets of salient points, takes into account the 2D distance of the projected salient points compared, as well as their depth difference:

$$E_S(P_i, P_e) = \frac{1}{N_S} \sum_{k=0}^{N_S} \left(d_k(I_i^S, \hat{M}_i^S(P_i, P_e)) + V_k(I_i^S, \hat{M}_i^S(P_i, P_e)) \right) \quad (11)$$

Where N_S is the number of the registered salient points.

As in the case of exhaustive matching, the set of camera parameters that minimizes the function E_s is considered as the best match. In most cases, the actual number of salient points registered is significantly lower than the ones extracted in the original 3D model and the query visual hull. Therefore, at least 10 salient points have to be registered, in order to achieve correspondence between the query range image and the surface that the virtual camera captures.

3.4 Hierarchical Search

Even if the use of salient points reduces the computational complexity of matching full range images, the matching procedure remains computationally expensive when exhaustively searching in the parameter space. In the proposed framework a hierarchical approach is developed that reduces the computational complexity by several orders of magnitude.

The hierarchical search algorithm builds initially a coarse sampling of the 6D parameter space and evaluates for each sample the error function as described in Section IV. The sample that produces the minimum error is considered as best match and then the algorithm proceeds to the second layer of the hierarchy. In this layer the error function is evaluated around the local neighborhood of the “winning” sample and the new sample that produces the minimum error is considered as best match for the second layer of the hierarchy. This procedure is repeated until the final layer of the hierarchy is reached that corresponds to the maximum accuracy as previously described.

For the first level of the hierarchy the whole range image is used. After that, the search is done around the sample that gave the smallest error, with the use of salient features. This is done repeatedly until satisfactory accuracy in the parameter space is achieved. In order to achieve even better accuracy, for the last 3 levels of the hierarchy, instead of matching using salient points, the whole range image is used. This way any noise intruded by the salient points, is omitted, thus achieving increased accuracy (about 15%).

3.5 Accuracy Analysis

An important issue of the proposed method and of hierarchical methods in general, is to avoid the convergence of the algorithm in a local minimum of the function to be minimized. This issue has been carefully addressed by selecting the proper error function (range error or salient error) and by sampling the parameter space in such a way so as to minimize the possibility of convergence in a local minimum.

Figure 2 illustrates four diagrams that concern a single object of the database. In **Figure 2a** the range error is illustrated for all possible different values of θ and ϕ , while **Figure 2b** illustrates the cover factor rate that corresponds to the area of the joint support set for both the query and the target range image divided to the query image area. Notice that the global minimum can be clearly identified, while other local minima have much higher value. Moreover, the valley of the global minimum extends to more than 50° for both the ϕ and θ parameters. This behavior is observed for all examined objects of the database and for the remaining parameters of the parameter space. Therefore, in the proposed algorithm, the initial sampling of the parameter space for the first level of the hierarchy is chosen to be in $\pi/4$ intervals. Coarser sampling would inhibit the performance of the algorithm, while denser sampling does not increase significantly the accuracy of the results, while slowing down the process.

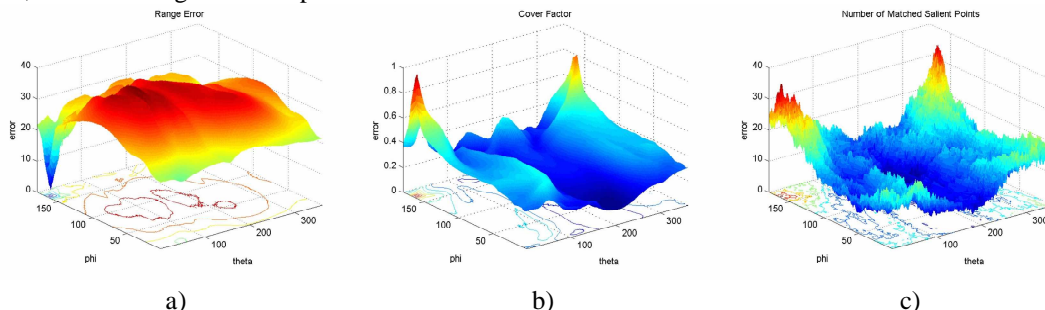


Figure 2. For an object of the database and for different values of θ and ϕ : a) Range error, b) cover factor rate (area of the joint support set for both the query and the target range images with respect to the query image area), c) number of salient pairs identified.

Figure 2c depicts the number of salient pairs that are identified in the query and the target range image. Figure 3 illustrates the same variables that appear in Figure 2, in the presence, however, of 20% occlusion. Notice that even if the number of salient pairs identified (Figure 2c) presents robustness to local minima and to occlusions as illustrated in Figure 3c the range error and the cover factor demonstrate increased robustness. Therefore, despite the increased speed of the calculation of the salient error, in the first level of the hierarchy, where convergence is of high importance, the range error is used.

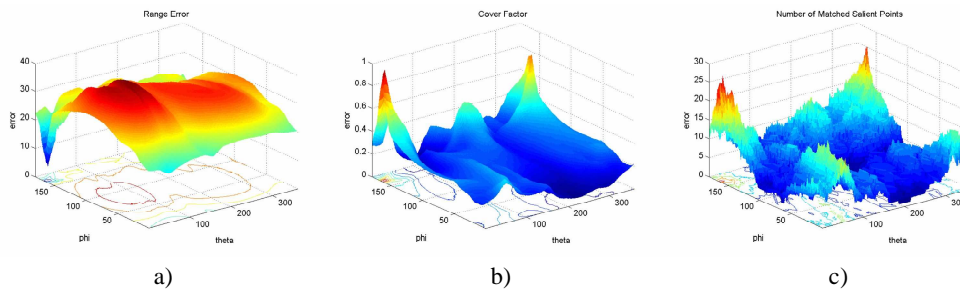


Figure 3. For an object of the database in the presence of occlusion (20%) and for different values of θ and ϕ : a) Range error, b) cover factor rate (area of the joint support set for both the query and the target range image with respect to the query image area), c) number of salient pairs identified.

4. EXPERIMENTAL RESULTS

The proposed method was tested on the 3D model database of the Watertight model Track of Shape Retrieval Contest '07. The database consists of 400 models organized in 20 categories. From the database, 400 range images were created from different views (one for each model), and entered as query in the matching algorithm. Each range image was compared with all the 3D models.

The following table describes the matching and retrieval efficiency of the proposed approach in terms of precision vs. recall, where precision is defined as the ratio of the relevant retrieved elements against the total number of the retrieved elements, and recall is the ratio of the relevant retrieved elements against the total relevant elements in the database. The first row of Table 1 refers to the precision of the proposed approach when the range image is used, while the second row refers to the precision, when 28dB Gaussian noise is added in the query range images.

Table 1. Precision of the proposed scheme for different recall values for clean – first row – and noisy – 28dB second row – range images.

Recall	5%	10%	20%	30%
Precision full	98%	89%	72%	59%
Precision noise	93%	82%	66%	52%

Moreover, Figure 4 illustrates the “precision vs. recall” curves for three different object classes. It is obvious that for classes with more representative shapes the matching scheme performs more efficiently.

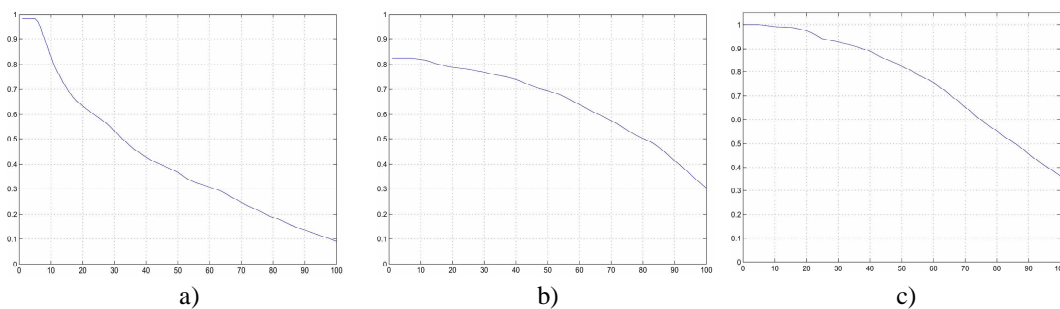


Figure 4. Precision vs. Recall curves for three different classes, namely a) animals, b) glasses, c) pliers

Finally, Figure 5 illustrates examples of the query range images – first column of the table – and the retrieved results. Notice that in the second column the result stems from a different class (bird instead of airplane) exhibits however similar geometry. In the final row the input is a range image generated by a 3D scanner.

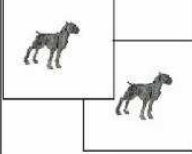

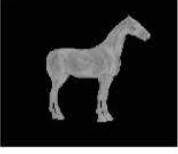

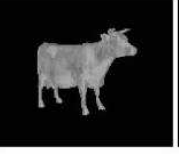




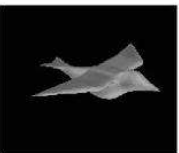


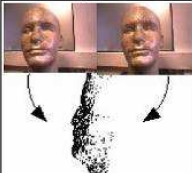


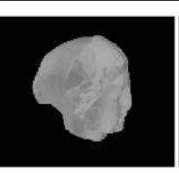

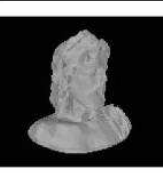
Query	Results				
					
					
					

Figure 5. Example of query range images (first column) and the retrieved results.

5. CONCLUSIONS

In this paper a novel method for point-based similarity estimation between 2.5D visual hulls and 3D objects has been presented. The proposed approach introduces a description of 3D objects using salient protruding features that provide a compact description of the underlying geometry. Matching is performed via a hierarchical view-based approach so as to reduce the computational complexity of matching pipeline. The approach has been tested in a benchmark dataset and is also seen to be robust to occlusion. Moreover, the salient protruding features descriptor could be potentially extended to articulated or deformable models utilizing the manifold surface of the underlying geometries that forms our major future research direction in the field of 3D object retrieval through range image queries.

ACKNOWLEDGEMENTS

This work was supported by the EU funded I-SEARCH IST STREP (FP7-248296)

REFERENCES

- Bespalov D., et. al., 2003, "Scalespace Representation of 3D Models and Topological Matching," In ACM Symposium on Solid Modeling '03, pp.208-215.
- Biasotti, S. et. al., 2003, 3D Shape Matching Through Topological Structures, DGCI, pp. 194-203.
- Bustos, B. et. al., 2005, Feature-based similarity search in 3D object databases, ACM Computing Surveys, Vol. 37, No. 4, pp. 345-387.
- Chen, D.Y. et. al., 2003, On visual similarity based 3D model retrieval, Computer Graphics Forum (EG 2003 Proceedings), Vol. 22, No. 3.

- Cicirello V. and Regli W.C., 2001, "Machining Feature-based Comparisons of Mechanical Parts," SMI 2001, pp.176-185.
- El-Mehalawi M. and Miller R.A., 2003, "A Database System of Mechanical Components Based on Geometric and Topological Similarity. part i: Representation," vol.31, no.1, Journal of Computer-Aided Design, pp.83-94,, January 2003.
- Gal, R. and Cohen-Or, D., 2006, Salient geometric features for partial shape matching and similarity, ACM Transactions on Graphics, Vol. 25, No. 1, pp. 130-150.
- Germann, M. et. al., 2007, Automatic Pose Estimation for Range Images on the GPU, Proc. 3D Digital Imaging and Modeling, 3DIM '07, pp. 81-90.
- Hilaga M., et. al, 2001, "Topology Matching for Fully Automatic Similarity Estimation of 3D Shapes," In SIGGRAPH 2001, pp.203-212.
- Hoffman D.D. and Singh M., 1997, "Salience of Visual Parts," Cognition, vol. 63, pp. 29-78.
- Lin M., et. al., 2004, "Visual Salience-Guided Mesh Decomposition," Proc. IEEE Int'l Workshop Multimedia Signal Processing (MMSP).
- Macrini D., et. al., 2002, "Viewbased 3-D object recognition using shock graphs," International Conference on Pattern Recognition (ICPR '02).
- Moustakas K., et. al., 2006, "MASTERPIECE: Physical Interaction and 3D content-based search in VR Applications", IEEE Multimedia, vol. 13, no. 3, pp. 92-100.
- Moustakas K., et. al., 2007, "SQ-Map: Efficient Layered Collision Detection and Haptic Rendering", IEEE Transactions on Visualization and Computer Graphics, vol. 13, no. 1, pp. 80 - 93
- Moustakas K., et. al., 2009, "3D content-based search using sketches", Springer International Journal on Personal and Ubiquitous Computing, vol.13, no.1, pp. 59-67.
- Ruggeri M., et. al., 2010, "Spectral-driven isometry-invariant matching of 3D shapes", International Journal of Computer Vision, Vol. 89, Numbers 2-3.
- Shilane P., and Funkhouser T., 2007, "Distinctive Regions of 3D Surfaces," ACM Transactions on Graphics, vol. 26, no. 2, Article 7.
- Shum, H.Y. et. al., 1996, On 3D Shape Similarity, Proc. IEEE Computer Vision and Pattern Recognition, pp. 526-531.
- Stavropoulos G., et. al., 2010, "3D Model Search and Retrieval from Range Images using Salient Features", IEEE Transactions on Multimedia, vol. 12, no.7, pp. 692-704.
- Tangelder, J.W.H. and Veltkamp, R.C., 2008. A survey of content based 3D shape retrieval methods. *In Multimedia Tools and Applications*, Vol. 39, No. 3, pp 441-471.
- Vranic, D., 2004, 3D model retrieval, Ph.D. thesis, University of Leipzig, Germany.

COMPUTATION AND VISUALIZATION OF SURFACE OF PROTEINS AND THEIR CHANNELS

Barbora Kozlíková, Igor Aleksandrowicz and Jiří Sochor
Masaryk University Brno, Czech Republic

ABSTRACT

In this paper we introduce our approach to the computation of molecular surface which is important for protein analysis. This challenge occupies the researchers for many years because of its complexity and many solutions have been proposed. Our approach comes from the generally known reduced surface algorithm (Sanner et al (1996)) but in our case this approach is innovated by using various simplifications and improvements, especially when dealing with specific problems. This is done with respect to the demands to utilize our algorithm not only for the computation of molecular surfaces but also for the surface detection of channels inside proteins. Our approach is namely used as a part of the CAVER Viewer application designed specially for the localization, visualization and further exploration of channels in protein molecules.

KEYWORDS

Surface, protein, channel, visualization

1. INTRODUCTION

Nowadays the biochemists may select from a wide choice of applications offering various types of protein visualization and analysis. Most of them provide users with different types of molecular representation. One of the most useful and popular displays is the visualization of the molecular surface. Using this method the biochemist can obtain the information about the overall shape and curvature of the surface. On the other hand, the inner structure of the molecule is completely hidden. This approach can be very useful mainly in cases when searching for some entrance paths leading from the outside of the molecule into its specific inner cavity.

In our research we are concentrated especially on searching for such paths which are called channels. These channels play a crucial role in the process of drug design. They are used as the transport paths for the small molecule of substrate. Such substrate enters the inside cavity of protein which is called an active site. For more information about channel definition and its computation see Medek et al (2007). For better illustration, Figure 1 shows simplified model of a molecule with one channel.

When searching for the molecular surface definition in the literature we can find various types of surfaces. Lee et al (1971) introduces so called solvent-accessible surface (SAS). This surface is defined by the sphere called a probe which rolls over the atoms of molecular surface. The SAS surface is then traced out by the center of the probe (see Figure 2).

Other type of the molecular surface is tightly glued to the bounding atoms and is called solvent-excluded surface (SES). It was introduced by Greer et al (1978) and computed by Connolly (1983). Our algorithm deals with this the SES type of surface. Both surfaces are outlined in the Figure 2.

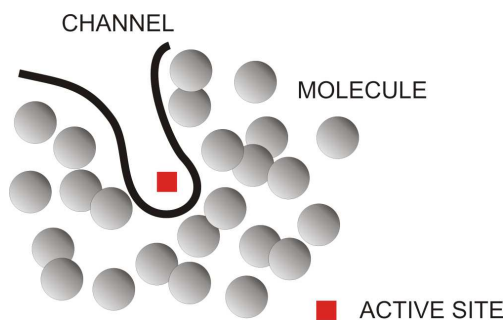


Figure 1. Definition of channel in the protein molecule

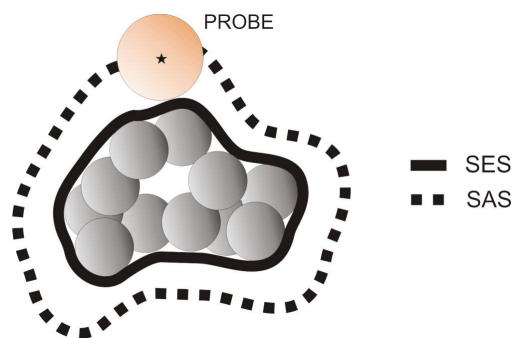


Figure 2. Definition of SES and SAS surface

2. RELATED WORK

Various approaches for surface construction can be taken into account. Some methods such as Liang et al (1998), Edelsbrunner et al (1994), use the alpha shapes approach. The main problem of alpha shapes is their complexity. In the worst case it runs in $O(n^2)$ time. This complexity led us to search for some other approach with better computational time.

Other method with the complexity in $O(n * \log n)$ is called reduced surfaces. It was introduced by Sanner et al (1996). From this solution various other algorithms come, such as Runge (1999), where the winged edge structure is used for solving problems with artifacts which appeared during the surface construction.

Zhang et al (2006) constructs the molecular surface using the implicit approximation with radial basis functions. This algorithm first computes multi-scale volumetric synthetic electron density maps and then the value of specific parameter has to be set. This parameter defines the level of approximation of the implicit solvation surface. In the next phase the modified dual contouring method extracts triangular meshes. Finally the mesh quality is improved using geometric flows. This approach is not suitable for our purpose because of several reasons. First, we have to compute electron density map which requires the knowledge processing of atom charges. Further the setting of the approximation parameter can lead to different results than we require. And the last reason, which is not measurable, is the resulting effect of the surface – it looks different from surfaces constructed by alpha shapes or reduced surfaces, which are generally accepted by biochemists.

3. ALGORITHM

As mentioned above, the core of our algorithm comes from the idea of algorithm published by Sanner et al (1996) and the later approach described by Runge (1999). The original algorithm for reduced surface computation was implemented as a part of the MSMS program and it consists of four phases.

In the first phase the skeleton of reduced surface is computed as the set of triangles and edges. Second phase processes the primitives created in the first phase and the analytical representation of the solvent-excluded surface is constructed. The problematic point of this phase is the presence of self-intersections. So the self-intersecting parts have to be processed in the third phase. When all problematic cases are solved, the last part of the algorithm generates triangles of the solvent-excluded surface, which can be subsequently visualized.

When designing the algorithm presented in this paper, we decided to come out from the reduced surface approach. Primarily we considered the complexity of the reduced surface algorithm – it has the best possible complexity among all known solutions of the molecular surface computation, which is $O(n * \log n)$.

However, our approach introduces several simplifications and diversities in comparison with the original algorithm. All the differences come out from the fact that we intend to use the algorithm for protein as well as channel surface computation and visualization. Our approach should also be as simple as possible because one of our main goals is the integration of the algorithm to the CAVER Viewer desktop application. CAVER Viewer is the tool for analysis and visualization of proteins which enables computation of channels in static

and also in dynamic case. When working with molecular dynamics we have to deal with huge amount of data (thousands of snapshots) and in each snapshot a predefined number of channels is computed. These channels have to be subsequently processed and the correspondence of channels between snapshots has to be determined (Medek et al (2008), Beneš et al (2009)). So it is evident that the algorithm for protein and channel surface computation must be very efficient both in time and space. To achieve this demand we had to focus on problems which appear during the surface reconstruction. This paper aims to describe our solution of these problems in detail because existing papers mostly just outline the exact solution of this part of the whole algorithm.

So our main contribution lies in the description of our solution of these artifacts when trying to resolve them at maximum rapidity and with minimal resources. Thanks to these demands we are avoiding the complex data structures such as winged edges are (used in Runge (1999)).

In the next section the principles of our approach to the protein and channel surface computation will be described.

3.1 Description of Algorithm

Before we start to describe our algorithm, we define some terms which are used further in the text.

- **RS triangle** – triangle of the reduced surface, it is defined by centers of three atoms which can touch the probe at the same time, see Figure 3.

- **RS edge** – edge of the reduced surface, it is defined by centers of two atoms which touch the probe at the same time. This edge is also an edge of the RS triangle, see Figure 3.

- **RS-free edge** – definition is the same as for the RS edge, but the difference lies in the fact that it does not form the edge of some RS triangle. The probe can roll around such edge and at the same time it touches two atoms. RS-free edge connects so called RS components, see Figure 4.

- **RS component** – set of RS triangles which does not contain a RS-free edge, see figure 4.

- **RS vertex** – atom on the vertex of some RS triangle or RS edge.

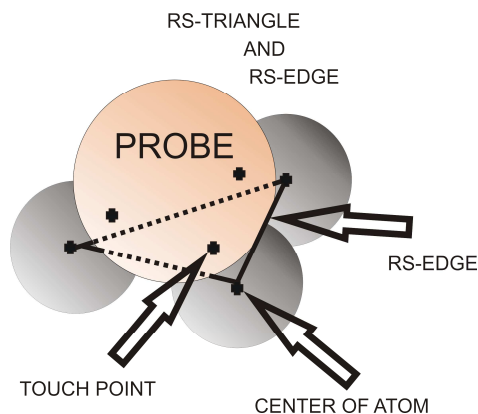


Figure 3. Definition of RS triangle and RS edge

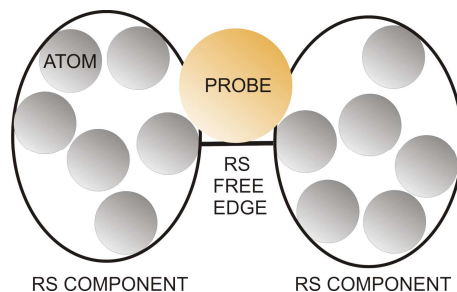


Figure 4. Definition of RS components and RS-free edges

Firstly we introduce the phases of the algorithm in pseudocode and then we will discuss them in detail.

SURFACE RECONSTRUCTION ALGORITHM

1. Reduced surface (RS) construction
 - a. Detection of neighbors for all atoms
 - b. RS-free edges and RS components detection
 - c. RS edges and RS triangles detection
 - d. **PROBLEMS:** multipoint touch, intersection of RS triangles
2. Generation of triangles
 - a. Triangles for all RS and RS-free edges (parts of torus)
 - b. **PROBLEM:** degenerated edges
 - c. Triangles for atoms
 - d. Triangles for RS triangles
 - e. **PROBLEM:** intersection of created plates
3. Sewing the triangles
 - a. Connecting spherical plates with contours
 - b. **PROBLEM:** long edges
4. Visualization of triangular mesh

3.1.1 Reduced Surface Construction

First of all the map of “neighbors” for all atoms is created. This relationship is defined as follows: when the probe in some position touches two atoms at once, these atoms are marked as neighbors.

In the next phase RS-free edges are detected. The map of neighbors is used in this process. For RS edges between each atom and each of its neighbors we have to verify if the probe can circle around such RS edge without touching some other atoms. If so, such edge is marked as an RS-free edge.

Afterwards the RS components are created. One RS component consists of atoms with specific attribute. This means that when rolling the probe over atoms of such component any RS-free edges are included in this component. So one molecule may consist of more RS components (separated by RS-free edges). An example of such components can be seen in Figure 4. The presence of many components is more often in channels.

As the result of this process we obtain the information about adjacent atoms, RS triangles, RS edges, RS-free edges and RS components. From this knowledge we can reconstruct the reduced surface of the whole molecule or channel.

Now for each component we have to define some starting atom for launching the reduced surface computation. The criterion for selection of such atom can be arbitrary, e.g. the atom with minimal x, y, z coordinates. In our algorithm we determine the starting atom as the atom in the maximal distance from the center of the whole molecule. Than we pose the probe into such position when it touches the starting atom and its center lies on the flowline between the center of the molecule and the center of the starting atom (see Figure 5).

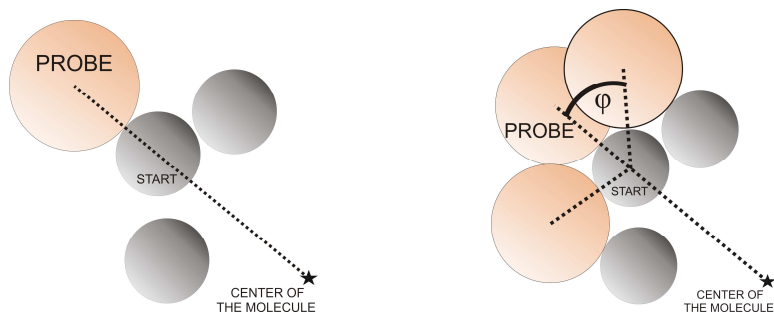


Figure 5. Starting position of the probe (left) and detection of the new position of probe (right)

Now we detect the first RS edge. We roll the probe around the starting atom until we reach some adjacent atom. This is performed for all atoms in the neighborhood of the starting atom and in each case the ϕ angle is computed. ϕ is the angle between the original position of the probe, the center of the starting atom and the new position of the probe after rolling around the adjacent atom (see Figure 5). From these potential new positions the one with the smallest ϕ angle is selected and thus the first RS edge is constructed. Similarly we detect the other RS edges and RS triangles. This process is stopped when any new RS triangles are found.

This phase generates first problematic situations. In some cases the probe can touch more than three atoms at once. In such case the RS polygon is constructed instead of the RS triangle. In order to follow the idea of triangular mesh construction we have to divide these RS polygons into RS triangles. This is reached by constructing a triangle fan inside each RS polygon.

Another problematic case occurs when for some triplet of atoms the probe creates two RS triangles (in two symmetric positions) which intersect. This is caused by the small size of the probe. This situation leads to unpleasant artifacts (see Figure 6). This event comes up when the probe rolls around some RS edge (between atoms A and B) of the first triangle (with vertices A, B, C) and stops after orbiting the round angle and touching the third atom (C) again. Solution of such case is simple. We replace such RS triangles by RS or RS-free edges (according to the situation) (see Figure 6). This forms the simplification in comparison with the original approach and the quality of results is sufficient for our purpose.

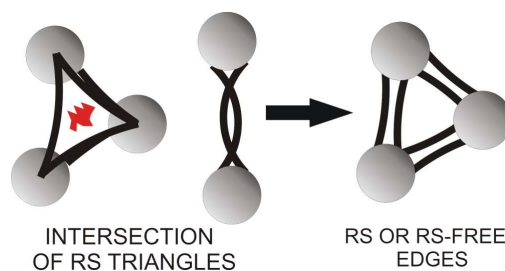


Figure 6. Problem with intersection of two RS triangles (left and middle – front and side view) and their replacement by three RS-free edges (right)

In this phase the reduced surface construction is complete. So we can continue to generate triangles from RS and RS-free edges.

3.1.2 Generation of Triangles

First triangles forming the faces for all RS edges and RS-free edges are generated. In fact we create parts of torus which is constructed by rolling the probe around two atoms. In this phase another self-intersections can occur. This is caused by the presence of so called “degenerated” edges (see Figure 7). In such edges the probe intersects the connecting line between two atoms forming the edge.

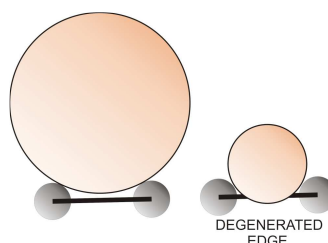


Figure 7. Common case (left) and degenerated edge (right)

This problem is solved by omitting the inner part of the edge which caused the self-intersection. We replace it by the arc which lies in the intersection between two adjacent RS triangles of such edge (see Figure 8).

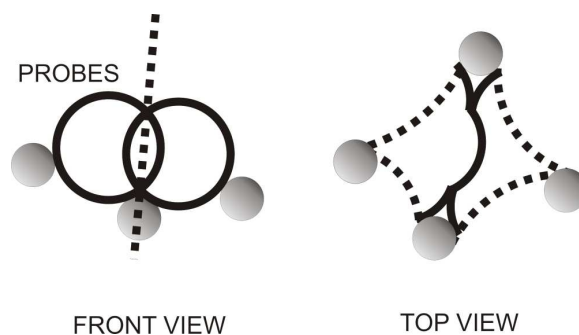


Figure 8. Solving the problem with degenerated edge

Subsequently triangles for each atom are generated. These triangles originate from the regular octahedron. Octahedron faces are uniformly divided into triangles and vertices of these triangles are projected to the same distance from the center of the atom. This distance is defined by the radius of the atom. The resulting plate of triangles is then cut by RS edges which correspond to this atom.

Very similar method is used for RS triangles, only triangles have the reverse orientation than in the previous atoms case. Triangles generated by RS triangles are cut by three planes. Each of these planes is defined by three characteristics – two atoms of the RS triangle and position of the probe in the RS triangle).

During the triangle mesh construction another problem occurs. It is possible that two RS triangles which are adjacent or belong to completely different parts of the reduced surface can intersect. In this case we have to cut the intersecting part of both plates which leads to creating borders (see Figure 9). These borders have to be subsequently stitched together. This is solved in the next phase of the algorithm.

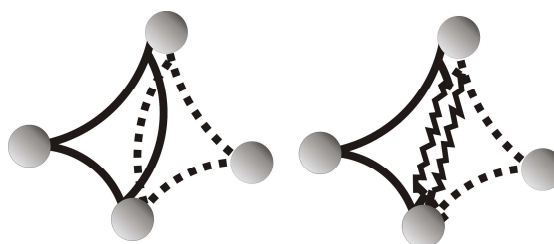


Figure 9. Solution of intersection of plates

3.1.3 Sewing the Triangles

After solving the most important problematic cases, the last phase follows – sewing together the generated triangles. Here the same principle for both cases (plates created from atoms and plates generated by the RS triangles) is used. We have to connect border vertices of plates with border vertices of adjacent plates. For the smooth sewing we can use the significant fact – plates of RS edges can be arranged in the manner that the border vertices of each plate are organized to the cycle. We term these cycles as **contours**.

For vertices of the spherical plates (originated from atoms) we cannot assume such property. Spherical plates are often cut by various planes so we usually cannot obtain the information about continuance of the border. Thus in this case we cannot create any contour and we have to cope only with the list of border vertices.

The actual sewing starts from some contour. We take a vertex of the contour and search for the nearest border vertex on the adjacent spherical plate. To be more specific, we keep two vertices of the contour – the current vertex C1 and the next vertex C2 and their two nearest border vertices on the spherical plate – S1 and S2. The selection is then based on test:

$$\text{distance } |C1, S2| < \text{distance } |C2, S1| \text{ then } S2 \text{ is added else } C2 \text{ is added}$$

If S1 and S2 vertices on the spherical plate are not adjacent we have to create a triangle fan between C1 vertex and vertices on the shortest path from S1 to S2 (such vertices are also part of the spherical plate).

When sewing one complication can occur. For some vertices on the contour there are any corresponding vertices on the spherical plate or vice versa. In such case we triangulate only vertices on the contour (or only on the spherical plate). Otherwise it can construct too long edges which is undesirable.

3.1.4 Visualization of Triangular Mesh

Now the triangular mesh is prepared for visualization. This phase is very straightforward and users can visualize the mesh according their needs. Our algorithm for surface computation produces the set of triangles which can be processed by the visualization algorithm. Of course, we can add supplementary information such as the corresponding atom for each triangle (useful for coloring of the surface according to atom types). Results of our visualization method, which takes all triangles of the surface and displays them with uniform coloring, can be seen in Figures 10 and 11.

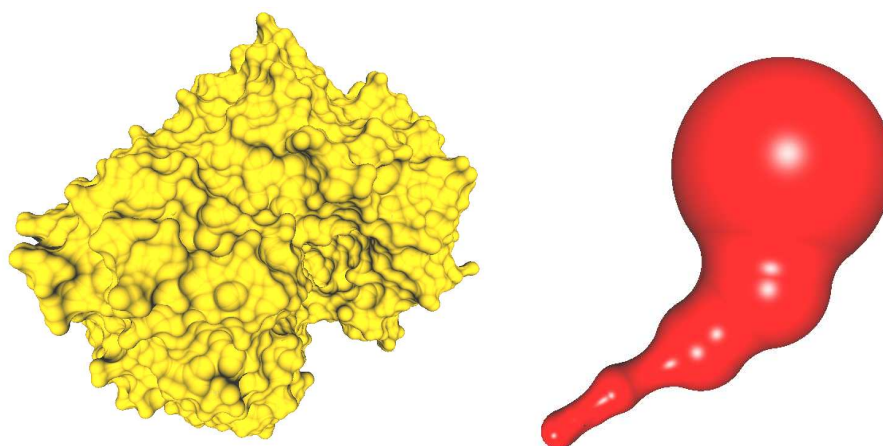


Figure 10. of 1GPE protein and surface visualization of its channel (CAVER Viewer)

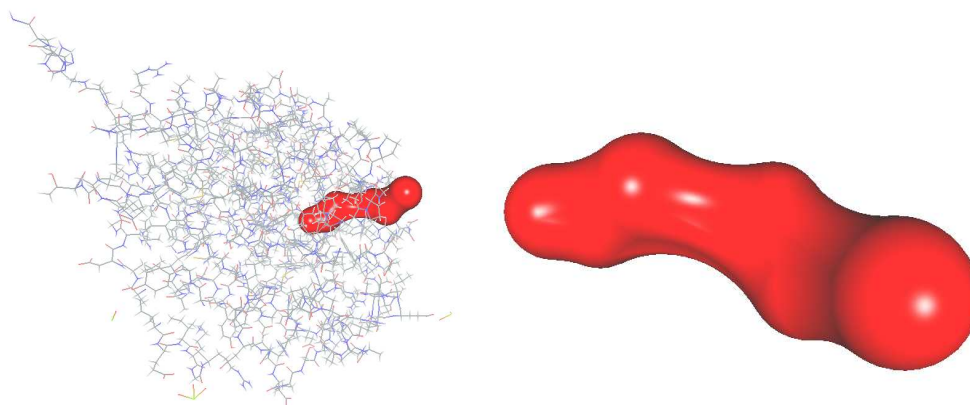


Figure 11. The channel inside 1GPE protein and closer view (CAVER Viewer)

4. CONCLUSION

In this paper we introduced our simplified and accelerated approach to the molecular surface computation. We started from the reduced surface algorithm and modified the basic approach for the purpose of channel visualization. We replaced the solutions of artifacts by more simple and elegant ones. In consequence the resulting implementation accelerated the surface computation significantly and now we are able to exploit it in the CAVER Viewer application where the response time plays a crucial role for visualization and interaction tasks.

In some degenerate cases the artifacts are still not completely solved but they are so uncommon that in our case their complete solution would slow down the performance with almost no added value. However they introduce an interesting direction for our research in this area. The most interesting field in protein analysis is the computation and visualization of the surfaces for molecular dynamics. We search for a method to compute the surface of a protein in each snapshot and then to visualize the dynamic changes using some suitable interpolation. This topic is a big challenge for the future research because of its time and space complexity.

ACKNOWLEDGEMENT

This work was supported by Ministry of Education of The Czech Republic, Contract No. LC06008 and by The Grant Agency of The Czech Republic, Contract No. P202/10/1435.

REFERENCES

- Sanner, M. F. et al, 1996. Reduced Surface: An Efficient Way to Compute Molecular Surfaces. *Biopolymers*, Vol. 38, pp. 305-320.
- Runge, D., 1999. Algorithms and Methods for the Visualization of Molecular Surfaces and Interfaces. *Master's thesis*, Humboldt University of Berlin, pp. 87.
- Medek, P. et al, 2007. Computation of Tunnels in Protein Molecules Using Delaunay Triangulation. *Journal of WSCG*, Vol. 15, No. 1-3, pp. 107-114 .
- Medek, P. et al, 2008. Multicriteria Tunnel Computation. *Computer Graphics and Imaging*. Innsbruck, Austria, pp. 5.
- Beneš, P. et al, 2009. Computation of Channels in Protein Dynamics. *Proceedings of IADIS International Conference Applied Computing*. Rome, Italy, pp. 251-258.
- Lee, B. et al, 1971. The Interpretation of Protein Structures: Estimation of Static Accessibility. *J. Mol. Biol.*, Vol. 55, pp. 379-400.
- Greer, J. et al, 1978. Macromolecular Shape and Surface Maps by Solvent Exclusion. *Proc Natl Acad Sci USA*, Vol. 75, No. 1, pp. 303-307.
- Connolly, M. L., 1983. Analytical Molecular Surface Calculation. *Journal of Applied Crystallography*, Vol. 16, pp. 548-558.
- Liang, j. et al, 1998. Analytical Shape Computation of Macromolecules: I. Molecular Area and Volume Through Alpha Shape, *Proteins: Structure, Function, and Genetics*, Vol. 33, pp. 1-17.
- Edelsbrunner, H. et al, 1994. Three-Dimensional Alpha Shapes, *ACM Transaction on Graphics*, Vol. 13, pp. 43-72.
- Zhang, Y. et al, 2006. Quality Meshing of Implicit Solvation Models of Biomolecular Structures, *Computer Aided Geometric Design*, Vol. 23, pp. 510-530.

SURFACE FEATURE SHARPENING: OBSERVATION MINUS CALCULATION (O-C) APPROACH

V. Savchenko¹ and M. Savchenko²

¹*Hosei University, 3-7-2 Kajino-cho, Koganei-shi, Tokyo, Japan*

²*Tokyo Institute of Technology, 2-12-1, O-okayama, Meguro-ku, Tokyo, Japan*

ABSTRACT

In this paper, we present an approach to recover sharp edges on a triangulated mesh. A region-growing partitioning technique is used to segment the unstructured noisy mesh that allows extracting and sharpening features on a surface triangular mesh. Features are created from boundaries between mesh segments and recovered through geometry prediction by applying a nonlinear root mean square procedure to find a 3D curve. This method can be useful in computer-aided design, reverse engineering, and other applications for feature detection on the surface and for sharp edges restoration.

KEYWORDS

Feature extraction, segmentation, triangular mesh, root mean square.

1. INTRODUCTION

In geometric modeling and reverse engineering applications such as reconstruction and modification of the shape of three-dimensional objects, sharp edges and corners are degraded on the resultant surface. The work in progress presented in the paper is motivated by the need to improve the quality of triangulated meshes with broken feature lines via their recovering. In fact, the approach presented in the paper can be applied to clouds of points. Let us also notice that main request from companies dealing with the solution of such problems is preservation of the overall surface texture, see an example shown in Figure 1, under the action of refinement in the neighborhood of the sharp edges.

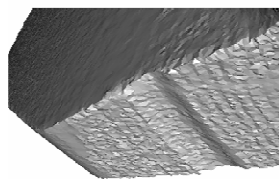


Figure 1. A fragment of the “Detail” model.

Feature lines, which can be defined as a local extrema of principal curvatures along corresponding principal directions of the original shape, however, can be destroyed by applying reverse engineering algorithms or during the polygonization of iso-surfaces.

The main feature extraction approach used in many algorithms, see, for example (Baker 2004), is the following. An edge can belong to a feature if it possesses the dihedral angle properties and the edge is close to the direction of Gaussian curvature. It is clear that if the triangulation is fairly coarse, then estimates of the dihedral angles, the principal curvatures, and their directions may not be accurate enough. A simple example of a local feature extraction according to normal deviations between triangular faces is given in Figure 2. As it can be seen from Figure 2 we have got a broken chain of line segments; finally, increasing the threshold of

the dihedral angle tolerance can lead to loss of edge direction. So what lessons can we take from this example, it tells us that we need to devise ways to knock ourselves out of local minima.

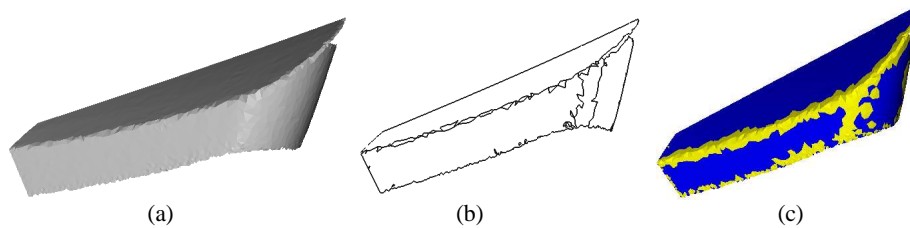


Figure 2. (a) The “H” model. (b) The feature detection according to estimates of dihedral angles. (c) The feature detection by the curvature estimation technique (Taubin 1995).

Our approach to features extraction and recovering sharp edges is based on the idea borrowed from methods which were called “beatification.” These methods, see for instance (Langbein et al 2001), seek similarities between features of objects; in our case we are seeking for reasonable solutions to establish criteria of similarities with a ruled surface - a surface S is ruled if through every point of S there is a straight line that lies on S . It was well noticed in (Mangan and Whittaker 1999) solutions are often application specific, depending quite heavily on one’s definition of “meaningful.” Here we suppose that sharp edges of a polygonal surface typically created by a CAD facet generator are similar to quadratic curves. Nevertheless, proposed approach is “meaningful” or applicable also for restoration of features of free surfaces as it shown in Figure 5(c).

2. RELATED WORKS

Estimation of surface features is a main part of the segmentation process and object recognition. A number of feature detection and mesh segmentation techniques have been proposed recently. In (Agathos et al 2007) an exhaustive overview of 3D mesh segmentation methodologies examining their suitability for CAD models is presented. In the paper, a categorization of the existing 3D mesh segmentation methods is proposed and the basic conclusions about different methods are drawn. The authors also present criteria and features used for each segmentation method. A review of mesh segmentation algorithms can be found also in (Shamir 2008). The paper (Attene et al 2006) provides a comparative study of the latest algorithms and evaluation results. The authors conclude that each algorithm has benefits and drawbacks and future research works on mesh segmentation will be useful. In the paper (Vieira and Shimada 2004), a new curvature based algorithm which segments the mesh into several regions is described. The authors extend the region growing algorithm to unstructured three-dimensional surface meshes. After seed vertex selection and sorting the vertices according to their filtered absolute curvatures, regions are growing in order of ascending curvatures. Estimation of surface features is a main part of the segmentation process and object recognition. Features are intrinsic properties of the 3D shape which include local geometry and topology (Bespalov et al 2004). The problem of feature decomposition is very important and many solutions were proposed for feature detection. The paper (Bespalov et al 2004) discusses the extension of a scale-space decomposition approach for feature extraction. Also, the authors discuss the performance of the technique used to extract features from CAD data in polyhedral representation. The work (Huang and Ascher 2008) describes a hybrid algorithm which while denoising regularizes triangle meshes on flat regions for further mesh processing preserves crease sharpness for faithful reconstruction. A clustering technique, which combines K-means and geometric a priori information, is suggested. When K-means does not give a very satisfactory vertex partition, the authors propose hierarchical K-means. Clustering repeats on the unresolved cluster until all sub-clusters consist of vertices with only one type of feature – corner, edge or non-feature. In (Jiao and Heath 2002) characterizations and techniques for detecting geometric features in surface mesh are presented. In the feature line extraction algorithm (Hildebrandt 2005), accurate method to estimate the discrete curvature is used. The approach is based on utilizing discrete differential operators on piecewise linear meshes which allow avoiding costly preprocessing steps such as surface fitting technique or constructing an implicit surface (Ohtake 2004). The paper (Demarsin et al 2005) presents an algorithm to extract features lines from a point cloud without curvature estimation. The algorithm uses a first order segmentation for receiving initial

information about the location of the feature lines. An approach based on surface interpolation (Daniels et al 2006) presents an algorithm to define a set of curves that are aligned along the feature edges of a point cloud. The approach is based upon the framework of Robust Moving Least Squares (RMLS) surfaces. Authors first select a number of potential feature points that are identified by the RMLS operator to be near to possible features. One issue with the RMLS is the inability to reconstruct smooth features. The errors produced by feature-insensitive sampling approaches are concentrated in what (Attene et al 2005) call chamfer triangles. Such triangles cut through the solid near sharp convex edges or through the solid's complement near sharp concave edges. The objective of the algorithm is to identify these triangles and to replace them with a finer triangle mesh portion that better approximates the sharp features of the solid. The approach to identify chamfer triangles is based on the initial identification of the smooth edges and on a succession of six simple filters. Each filter colors the edges, vertices, or triangles, based on the colors of their adjacent or incident elements. Then, retriangulation is applied by subdividing the chamfer triangles, including the corner ones. A new vertex in the middle of each chamfer edge and in the middle of each corner triangle is inserted. Experiments on a variety of meshes indicate that proposed algorithm is extremely fast and robust. The algorithm has limitations according to the remarks of authors. It may produce sharp edges where the original model has a feature that has been smoothed with a small-radius blend. Nevertheless, according to our opinion it is not a serious limitation. The algorithm shows very good results for sufficiently smooth surfaces, however, it was tested using a representation of sufficiently smooth surfaces and the filters can only sharpen edges that are generated by Marching Cubes algorithm. An algorithm presented in (Ohtake and Belyev 2003) relies on the underlying mathematical surface representation or the Hermite data (exact intersection points and normals) equipped with surface normals. A robust algorithm which does not rely on the identification of chamfered edges was proposed by (Wang 2006). The region embedding sharp features are firstly recognized by measuring the maximal normal variation around a vertex on a triangulated mesh. Then the sharp region is shrunk into its skeleton. An ideal position for a vertex minimizes the difference between its position and the smoothness signals - tangent planes. An objective function for minimization, which should minimize its distances to the tangent planes near the vertex, is employed. During region shrinking, the positions of newly generated vertices are predicted by the triangles existed in the non-sharp region. After that, vertices on the skeleton are repositioned to generate the geometry of single-line-wide sharp edges. The method efficiently produces a high quality recovery of sharp edges. However, it has also limitations discussed by the author. Limited by the nature of an incremental approach, the author assumes that the noises or errors are not exhibited on the non-sharp-feature region. This technique failed to sharpen features of sufficiently simple, according to our opinion, the "Isis" model (see Figure 9). The author noticed that similar to other sharpening algorithms, the feature that blends smoothly into a flat area may be miss-sharpened by the approach if the sampling rate is not able to generate enough great normal variations around aliasing regions.

This paper makes the following technical contributions. First of all, let us note that according to our knowledge there were no attempts to consider the problem of surface features (sharp edges) restoration as a navigation or motion detection problem. Our goal is the following: we are striving to develop an automatic, reliable, and simple in realization feature sharpening algorithm based on a motion determination of a "virtual" observer. The proposed approach looks attractive because of the simplicity - no surface approximation and no filtering in common image processing sense are needed. And finally, we present visual results to demonstrate the usefulness of the proposed approach.

3. BASIC APPROACH

The basic premise of our approach is that any reasonable surface is made up of several smooth regions where a tangent plane to the surface at a given point can be defined. In our method, we define a feature as a set of simultaneous line segments created by intersection of the tangent planes, see Figure 3, where the red line shows a simultaneous feature line. We consider the feature extraction and feature sharpening problem as a typical statistically-based navigation (filtering) problem: find a trajectory (curve) of a "virtual" observer with minimal deviation from a feature, see Figure 4. In turn, it helps also to find a set of mesh nodes satisfying given constraints such as sharp edges. The result of tracing the model is shown in Figure 5.7

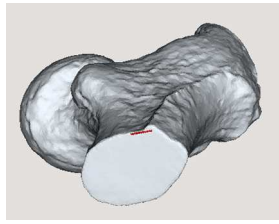


Figure 3. A simultaneous feature line segment at a seed point of the “Balljoint” model.

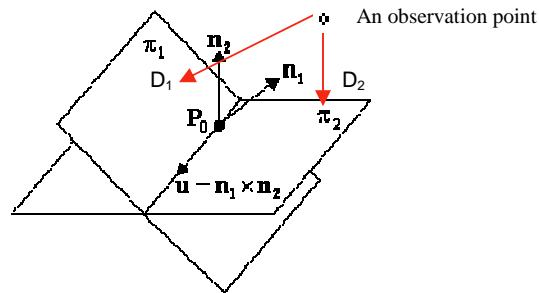


Figure 4. O-C schema. D_1 and D_2 measurements - distances between the observation point (point belonging to a trace) and tangent planes defined in the vicinity of the observation point. Vector \mathbf{u} defines the motion direction. \mathbf{n}_1 and \mathbf{n}_2 are normals to the tangent planes π_1 and π_2 .

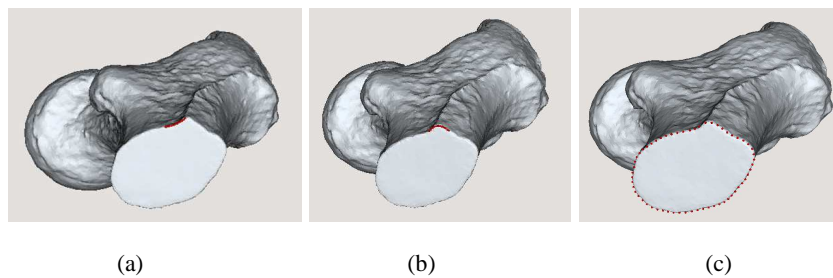


Figure 5. The “Balljoint” model, 68.5K triangles. (a, b) Trace points are shown for two trajectory steps. (c) Result of tracing the model. 234 trajectory steps (shown 117 steps). The time of processing is about 90 sec.

In the absence of a precise mathematical description of the underlying surface all information about surface properties such as smoothness and curvature must be inferred from the triangulation itself. Any surface is essentially random or chaotic in nature, as it was mentioned in (Baker 2004). However, it was noticed by many researchers that meshes appearing as result of triangulating scattered data often exhibit almost Gaussian distribution of, for example, triangle aspect ratios. Thus, our second premise is that such statistical characteristics as mean angle and range deviations can be used in the proposed algorithm.

In 3D computer vision, 2D views of the world are captured as images which are composed of pixel arrays that store the brightness of incident radiation. In our case, the “image” can be considered as a set of node’s coordinates projected onto a fitting tangent plane. The main assumption is that the 2D scene (part of the scene) can be extended to cover 3D scenes if the orientation of the image is known. For simplicity, we call planar surface segments 2D views. The problem considered here can be called as “the problem of angular and motion estimation of 2D views”. The motion of 2D views is defined in terms of the instantaneous vector of angular ($\omega_x(t)$, $\omega_y(t)$, $\omega_z(t)$) and translation velocity (V_x , V_y , V_z). In our case, we, in fact, can propose defining a movement of 2D views according to linear or higher degree approximation.

The basic steps of the algorithm are as follows:

1. Form a cluster of mesh vertices near a feature point;
2. Define two moving (tangent) planes;
3. Find an intersection of moving planes;
4. Estimate local normals and distances to moving planes;
5. Estimate calculated local normals and distances to the planes;

6. Apply nonlinear root mean square (RMS) procedure to find a 3D curve;
7. Find new predicted curve (trajectory) coordinates from estimated values;
8. Project a covariance matrix on the next step;
9. Repeat previous steps for the next step.

Below we describe the main steps (1 and 6) of the algorithm in detail.

Point clustering. Point (vertex) clustering is achieved by sequential grouping cells with minimal deviations of dihedral angles between cells and growing clusters normals (also minimal deviations of range data are used). It seems that the idea of sequential data grouping was proposed in (Ward 1963). An initial cluster is defined by two cells (sharing an edge) with minimal deviation between cells normals. A fragment of point clustering is shown in Figure 6.

For the first step, forming a cluster of mesh vertices is produced around a seed point selected by the user. For the next steps, clusters are selected around predicted trajectory coordinates belonging to the restored sharp edge.

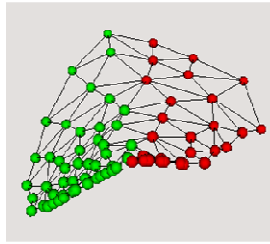


Figure 6. The “H” model. A fragment of point clustering.

RMS procedure. Let ξ, η, ζ be the fixed coordinate system and X, Y, Z is the connected with the solid coordinate system. Let γ, ψ , and ν be the direction angles of the fixed coordinate system in a coordinate system bound with the solid. We can assume that we know a sequence of values of the angular $\omega(t)$ and translation $V(t)$ velocities (here we consider the translation motion as the problem of approximating a spatial curve $C(x(t), y(t), z(t))$ by a quadratic spline curve). We need to construct a smooth matrix valued function $T(t)$ such that $T(t)$ is the orientation matrix for all t to describe the coordinate transformation of the image:

$$[X, Y, Z]^t = T[\xi, \eta, \zeta]^t,$$

where T is so called essential matrix:

$$\begin{pmatrix} 1 & \nu - \psi \\ -\nu & 1 & \gamma \\ -\psi & -\gamma & 1 \end{pmatrix}$$

If the t (time) dependency of 2D views is known, we may solve this problem by numerical integration of the system of differential equations for small values of the vector $\omega(t) = \{\omega_1, \omega_2, \omega_3\}$ and initial conditions γ_0, ψ_0, ν_0 :

$$d\gamma/dt \approx \omega_x - \nu\omega_y, \quad d\psi/dt \approx \omega_y - \omega_2\gamma, \quad d\nu/dt \approx \omega_y\gamma + \omega_z.$$

We assume that we have a sequence of orientation matrices in which every matrix is close to the previous one. Consider a variation of the “observer - object” vector $U = \{u_1, u_2, u_3\}$ defined in the fixed coordinate system. Then without taking into account translation components we can calculate a new “rotated” vector U' as follows:

$$u'_x = u_1 + \nu u_2 - \psi u_3, \quad u'_y = -\nu u_1 + u_2 + \gamma u_3, \quad u'_z = \psi u_1 - \gamma u_2 + u_3. \quad (1)$$

We suppose that we have linear variation of the angular coordinates:

$$\gamma = \gamma_0 + \frac{d\gamma}{dt} t, \quad \psi = \psi_0 + \frac{d\psi}{dt} t, \quad \nu = \nu_0 + \frac{d\nu}{dt} t.$$

Time dependent derivatives of the angular coordinates are follows:

$$d\nu/dt = \omega_2\gamma + \omega_3, \quad d\gamma/dt = \omega_1 - \omega_2\nu, \quad d\psi/dt = \omega_2 - \omega_3\gamma.$$

By substituting (1) (see above) we obtain u'_x as follows:

$$\dot{u}_x = u_1 + v u_2 - \psi u_3 = u_1 + (v_0 + \frac{dv}{dt} t) u_2 - (\psi_0 + \frac{d\psi}{dt} t) u_3$$

and similarly for \dot{u}_y and \dot{u}_z . According to this angular transformation we can define so called isochrones derivatives for time t_0 :

$$\begin{aligned} \partial \dot{u}_x / \partial \gamma_0 &= 0, \quad \partial \dot{u}_x / \partial \psi_0 = -u_3, \quad \partial \dot{u}_x / \partial v_0 = -u_2, \\ \partial \dot{u}_y / \partial \gamma_0 &= -u_3, \quad \partial \dot{u}_y / \partial \psi_0 = 0, \quad \partial \dot{u}_y / \partial v_0 = -u_1, \\ \partial \dot{u}_z / \partial \gamma_0 &= -u_2, \quad \partial \dot{u}_z / \partial \psi_0 = u_1, \quad \partial \dot{u}_z / \partial v_0 = 0. \end{aligned} \quad (2)$$

The partial derivatives of the space coordinates with respect to the unknown motion parameters $\{\omega_1, \omega_2, \omega_3\}$ are calculated as follows:

$$\begin{aligned} \partial \dot{u}_x / \partial \omega_1 &= 0, \quad \partial \dot{u}_x / \partial \omega_2 = \gamma_0 t u_2 - t u_3, \quad \partial \dot{u}_x / \partial \omega_3 = t u_2 + \gamma_0 t u_3, \\ \partial \dot{u}_y / \partial \omega_1 &= t u_3, \quad \partial \dot{u}_y / \partial \omega_2 = -\gamma_0 u_1 t - v_0 u_3 t, \quad \partial \dot{u}_y / \partial \omega_3 = -t u_1, \\ \partial \dot{u}_z / \partial \omega_1 &= -t u_2, \quad \partial \dot{u}_z / \partial \omega_2 = t u_1 + v_0 u_2 t, \quad \partial \dot{u}_z / \partial \omega_3 = -\gamma_0 t u_1. \end{aligned} \quad (3)$$

If we define vector of improved parameters \mathbf{q} as

$$\mathbf{q} = \{a_0, b_0, c_0, a_1, b_1, c_1, a_2, b_2, c_2, \omega^1_1, \omega^1_2, \omega^1_3, \omega^2_1, \omega^2_2, \omega^2_3, v^1_0, \gamma^1_0, \psi^1_0, v^2_0, \gamma^2_0, \psi^2_0\}$$

and measurement functions as F , then taking into account (2) and (3) we can easily calculate partial derivatives of 2D views $\partial F / \partial \mathbf{q}$, for example,

$$\partial F_k / \partial v = \partial F_k / \partial u_x \partial u_x / \partial v + \partial F_k / \partial u_y \partial u_y / \partial v + \partial F_k / \partial u_z \partial u_z / \partial v.$$

To estimate unknown parameters q_i for all $i = 1 \dots 21$, where common parameters a_0, \dots, c_2 are coefficients defining a spatial quadratic curve, we have to minimize the sum of the squared errors of measured functions and calculated. It leads to the solution of the linear system of equations:

$$\mathbf{A} \Delta \mathbf{q}^{(s+1)} = -\mathbf{B},$$

where $A_{l,m} = \sum_{k=1}^N P_k \partial B_k / \partial q_l \partial B_k / \partial q_m$ for l, m -th matrix element, $\Delta \mathbf{q}^{(s+1)} = \mathbf{q}^{(s+1)} - \mathbf{q}^{(s)}$, $s = 0, 1, 2, \dots$ is a number of iterations, $B_i = -\sum_{k=1}^N P_k \partial B_k / \partial q_i \xi_k$ for N corresponding observations, where ξ_k is the residual of k -th observation. $P_k = 1/\sigma_k^2$ is the weight of the observation for the given accuracy σ_k of the observation.

Calculating partial derivatives of the space coordinates with respect to the common parameters is straightforward and we don't give formulas for them.

4. IMPLEMENTATION AND EXPERIMENTS

In the preprocessing step, statistic characteristics such as mean depth and angular deviations of a mesh are calculated. Average length of the mesh edges, which is used as a "time" step for motion along the curve, is also calculated on this stage. Then, after seed vertex selection and clustering the vertices according to their filtered absolute deviations of vertex normal and range deviations, regions of feature extraction are selected along the trajectory defined according to the RMS procedure. In the current implementation, we use a batch algorithm; 17 observation points are selected for a current RMS step. The tracking of a feature in 3D space from images is a nonlinear estimation problem. The initial static parameters of the model are calculated from a seed point and all velocities are initialized to 0. For all shown examples one iteration step was used. Figure 7 shows the results of feature restoration of a decimated portion of the "Detail" model. It can be seen that we have successively used the proposed algorithm for feature restoration of extremely degraded mesh.

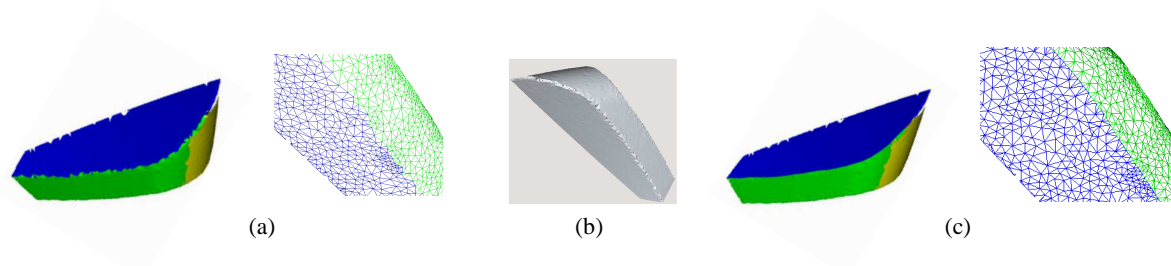


Figure 7. Feature restoration of the “H” model. (a) The initial model and the mesh fragment; (b) Trajectory points (in red); (c) The final mesh and the mesh fragment.

Figures 8 and 9 show experimental results of feature restoration and mesh remeshing for the “Detail” and the “Isis” models.

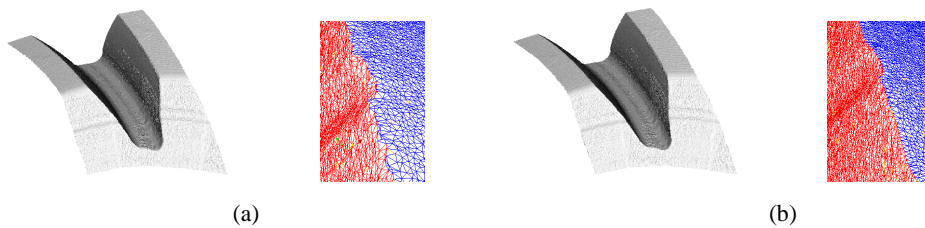


Figure 8. (a) The initial “Detail” model and the mesh fragment. (b) The model and the mesh fragment after feature restoration.

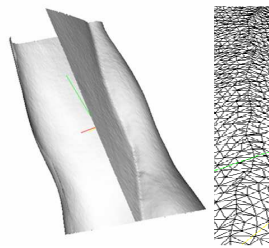


Figure 9. A portion of the “Isis” model and the mesh fragment after feature restoration.

5. CONCLUSION

The algorithm presented here is fairly simple and straightforward. We suppose that its true importance lies in the fact that it forms the foundation for algorithms for feature extraction and sharpening of complex meshes. The main limitation is that the segmentation process is time consuming. In the future, we are going to develop a parallel realization of the proposed approach. However, we suppose that sequential clustering algorithm can be greatly accelerated by using the hysteresis principle because of closely spaced points on the trajectory should expect a slight change in the separation of surface points. Question of implementing Kalman filtering that can drastically decrease processing time is also now under consideration. Let us notice that in spite of considerable improvement of the algorithm of sequential grouping cells based on the use dihedral angles and range deviations by the optimal prediction of a feature trajectory stable and robust surface mesh segmentation can not be guaranteed. In particular, we are still looking for an extension and improvement of the clustering technique. A technical question still waiting its solution is surface clustering for branching points (where two or more features appear). The inclusion of additional improvement parameters in the state vector is straightforward and slightly complicates the present algorithm. Optimal placement (control of the generated vertex positions) is also waiting its realization.

REFERENCES

- Agathos, A. et al, 2007, 3D Mesh Segmentation Methodologies for CAD applications, *Compute-Aided Design & Applications*, Vol.4, No. 6, pp.827-841.
- Attene, M. et al, 2006, Mesh segmentation: A comparative study, *Proceedings of the International Conference on Shape Modeling and Applications*, pp.14-25.
- Attene, M. et al, 2005, Sharpen&Bend: Recovering curved sharp edges in triangle meshes produced by feature-insensitive sampling, *IEEE Transactions on Visualization and Computer Graphics*, Vol. 11, No. 2, pp.181-192.
- Baker, T.J., 2004, Identification and Preservation of Surface Features, *Proceedings of International Roundtable*, pp.299-310.
- Bespalov, D. et al, 2004, Local feature extraction using scale-space decomposition, *Proceedings of ASME Design Engineering Technical Conferences, Computers and Information in Engineering Conference (DETC 2004-57702)*, ASME Press, pp. 501-510.
- Daniels, J., 2007, Robust Smooth Feature Extraction from Point Clouds, *Proceedings of SMI, USA*, pp. 123-136.
- Demarsin, K., 2005, Detection of feature lines in a point cloud by combination of first order segmentation and graph theory, Department of Computer Science, K.U.Leuven, Belgium, Report TW 440.
- Hildebrandt, K., 2005, Smooth Feature Lines of Surface Meshes, *Proceedings of Symposium on Geometry Processing*, pp. 85-90.
- Huang, H. and Ascher, U., 2008, Surface Mesh Smoothing, Regularization and Feature Detection, *SIAM Scientific Computing*, Vol. 31, pp. 74-93.
- Jiao, X. and Heath, M.T, 2002, Feature Detection for Surface Meshes, *Proceedings of the 8th International Conference on Numerical Grid Generation in Computational Field Simulations*, pp. 705-714.
- Langbein, F.C. et al, 2001, Finding Approximate Shape Regularities In Reverse Engineered Solid Models Bounded By Simple Surfaces, *Proceedings of the 6th ACM Symposium on Solid Modeling and Applications*, pp. 206-215.
- Mangan, A.P. and Whitaker, R.T., 1999, Partitioning 3D Surface Meshes Using Watershed Segmentation, *IEEE Transaction on Visualization and Computer Graphics*, Vol. 5, No. 4, pp. 308-321.
- Ohtake, Y. and A Belyev, A., 2003, Dual-prime mesh optimization for polygonized implicit surfaces with sharp features, *Proceedings of ACM Solid Modeling, Symposium*, pp. 171-178.
- Ohtake, Y., 2004, Ridge-valley lines on meshes via implicit surface fitting, *Proceedings of SIGGRAPH*, pp. 609-612.
- Shamir, A., 2008, A survey on mesh segmentation techniques, *Computer Graphics Forum*, Vol.27, No.6, pp. 1539-1556.
- Taubin, G., 1995, Estimating the tensor of curvature of a surface from a polyhedral approximation, *Proceedings of the 5th International Conference on Computer Vision*, pp. 902-907.
- Vieira, M. and Shimada, K., 2004, Segmentation of noisy laser-scanned generated meshes with piecewise polynomial approximations, *Proceedings of SME Design Automation Conference*, pp. 927-934.
- Wang, Charlie C. L., 2006, Incremental Reconstruction of Sharp Edges on Mesh Surfaces, *Computer-Aided Design*, Vol. 38, pp. 689-702.
- Ward, J. H. Jr., 1963, Hierarchical grouping to optimize an objective function, *Journal of American Statistic Association*, Vol. 58, No. 301, pp. 236-244.

ADAPTIVE CHARACTER MOTION SYNTHESIS BY QUALITATIVE APPROACH

Fangde Liu and Xiaosong Yang and Jianjun Zhang

National Center For Computer Animation, Media School, Bournemouth University

ABSTRACT

Adaptive motion synthesis normally involves complicate computation from physical simulation. However some biological researchers believe that the neural system takes little effort in motor control. Instead of planning motion from ground up, living creatures may only maintain or tweak some basic motion pattern for qualitative properties of complicate interaction between body and environment. Inspired by this idea, in this paper, we proposed a new method for adaptive motion synthesis based on the qualitative control theory. Adaptive motion control is achieved through enhancing the structural stability, rather than counteracting perturbations. Compared with current simulation method, our algorithm is more computational efficient and has the potential to be accelerated by GPU.

KEYWORDS

Character Motion Synthesis, Qualitative Dynamic, Physics Based Animation

1. INTRODUCTION

Human beings are very sensitive to motions. From the variety in motion details, humans can infer the changes in mental states, health conditions or even the surrounding environment. This makes Character Motion Synthesis (CMS) a difficult task. In industry, high quality motions are still majorly generated manually, animators need to tweak and set key frames for a large number of joints. To make things worse, it is very difficult to reuse these motion data. When the environment or the character changed, new motions have to be redesigned manually. To save animator from these tedious manual works, many researchers are trying to generate lifelike motions by simulating the dynamic effects between body, environment and the neural control system.

In Biology, lots of researchers have been working on the secret of motions for centuries and discovered some important motion features of live creatures:

Adaptive Natural motions are adaptive to the changes in the environment or body. For example, a human being can easily adjust its gait according to different terrains.

Agile The reaction of human and most animals is very fast. Even in a complicated changing environment, human can adapt their motions in real time.

Energy Efficient According to Darwin's Theory of Evolution, a natural motion should be energy efficient. Live creatures spent far less energy than we expected. An example is that the energy consumed by human walking is only 10% of that for a robot of the same scale.

Features above are very difficult to achieve by current CMS methods. In this paper, we will bring in the Qualitative Control Theory (QCT). From the viewpoint of QCT, a natural motion is in nature a **structural stable autonomous** system. Adaptation to different environment or characters will be produced automatically with very little control effort. All the above three natural motion features can be achieved from our system. It especially suits repetitive and low energy motion tasks which are most challenging for current researches. Besides, our approach is computational efficient and has the potential to be accelerated by GPU.

2. RELATED WORKS

2.1 Dynamic Motion Synthesis and Control

Dynamic Motion Synthesis synthesizes character motion through simulating the mechanics of the body. The most difficult task is to design system with functionality of biological neural systems. Early research applied classical PD controllers; first for locomotion (Raibert and Hodgins, 1991), later for different tasks like running, bicycling, vaulting and balancing (Hodgins et al., 1995). Limit Circle Control (LCC) (Laszlo et al., 1996) provided an alternative method for lower energy locomotion. However both methods need predefined motion trajectories and are not good at generating adaptive motions.

Because of large number of redundant DOFs, in most cases, motion solutions are not unique. Many optimization methods have been applied to choose the “best” motion. A popular idea is to minimize the energy cost V , such that

$$V = \int_{t_0}^{t_1} F_a(x)^2 dt$$

Where F_a is the active force generated by actuators like motors or muscles. This is introduced to CMS research as the **Spacetime Constraints** (Witkin and Kass, 1988), and serves as the foundation for many modern CMS researches. Jain et al. (2009) provided an example for locomotion; Macchietto et al. (2009) found a method for balance maintaining movement. Liu (2009) proposed a method for object manipulating animation.

The Spacetime method may adapt the motion. However it is in nature a variational optimization method and faces several problems.

Efficiency In many cases, it will take long time to find the "best" solution and there is no guarantee the optimal solution can be achieved. For complex body structures the computation will takes prohibitive long time (Anderson and Pandy, 2001). Optimization techniques like time window and multi-grid techniques were proposed by Cohen (1992) and Liu et al. (1994). Still only a few researches (Popović and Witkin, 1999) proposed Spacetime Constraint for full body dynamic animation.

Sensitive and Over Specific Current numeric methods are very sensitive to model accuracy and initial conditions. Precise model for both the environment and body have to be prebuilt. Liu (2005) points out that spacetime constraint methods only suit high energy motions like jumping and running; for low energy motion tasks like walking the results don't look natural. This is mainly because the muscle effects are neglected. Motions like heart beating, breathing, or motions of other animals such as the swimming of fish and jellyfish, flying of birds have not been synthesized with dynamic methods, mainly for the lack of a feasible dynamic model.

2.2 Biological Research

In biological research, motor control is an age old problem full of paradoxes. Motor control in nature is a complex process involving many chemical, electrical and mechanical effects. An immediate idea is Motor Control will involve complicated computation. However, the characteristics of biological neural systems are on the opposite (Glynn, 2003). Neural signal transmitting speed is very slow; and there is a long delay between neural signal firing and force generation in muscles. Besides, the neural signals are also noisy. The body structure and environment are nonlinear, noisy and time varying.

Current research evidences and common life experience show that motor control involves little control effort. Many experiments show motion can happen even without brain input. Despite the complexity of body structures and environment, the natural motor control strategy seems simple and involves very little computational work. For many animals, the neural structure active in motor control is the **Central Pattern Generator (CPG)** which generates rhythmic signals.

Many biological ideas provide space for an efficient motion adaptation. **Uncontrolled Manifold Hypothesis** proposed that some DOFs are not controlled and freely influenced by the environment (Latash, 2008). **The Equilibrium Point Hypothesis (EPH)** suggested that what the neural systems controls is not trajectory, but the final position. **The Impedance Control Hypothesis** (Hogan, 1985) refined the idea of EPH by providing an explanation of the functionality of the extra DOFs. Extra DOFs provide a way to

control the stability and admittance of final position according to the motion purpose. **Morphological Computation Theory** (Nishikawa et al., 2007; Pfeifer and Iida, 2005) believed that both the body structure and the environment play a crucial role. Basic motion patterns are generated by body and environment; the neural systems only maintain or tweak basic motion patterns.

3. QUALITATIVE CONTROL THEORY FOR CMS

Inspired by the biological research, in this paper we adopt a different strategy for motion adaptation. Perturbations are allowed to freely affect the motion, and motion control is applied to maintain the qualitative properties.

3.1 The Qualitative Control Theory

The Qualitative Control Theory is a mathematical description of the Morphological Computation Theory. In qualitative control theory the basic patterns of motion are called **motion primitive**. We propose that in mathematic viewpoint, motion primitives are **structural stable autonomous systems**.

3.1.1 Basic Concepts of Qualitative Dynamics

The configuration of system is described by state vector q , M is the state space which is a manifold, motion is a trajectory $q(t)$, which is usually represented by ordinary differential equation in the form (1).

$$\dot{q} = F_u(q) = F(q, u), q \in M \quad (1)$$

Where u is the control effort. F is determined by the system's natural property. If $u = 0$, systems of no control input are **autonomous systems**. For every point $q \in M$, F and u determines a derivative vector \dot{q} . All the vectors over the full space of M form the **vector field** V . $q(t)$ is the integral curve of \dot{q} , which is defined as flow Φ , all the flows form the **phase portrait**, which illustrates all the possible motions of the dynamic system.

On the phase plane, flows only intersect at some special position, **Fix Point** $\{q_e \mid H(q_e) = 0\}$ and **Limit Circle** $\{q \mid H(q(0)) = H(q(T))\}$, both of which are also called **equilibria**.

At each **equilibria**, the local space can be divided into three subspaces: **centre manifold**, **stable manifold**, and **unstable manifold**. They divide the state space into different regions, result in a cellular topological structure. In each region, all the flows converges to one **attractor**, the corresponding region is called **basin of attraction**.

3.1.2 Motion Adaptation and Structural Stability

A mechanical system can be stable without any control effort. This kind of stability is rough stability or structure stability (Andronov and Pontryagin, 1937) which is determined by the topology structure of the system (Jonckheere, 1997). Motion adaptation can be modeled as homeomorphism. Homeomorphic flows share the same topological structure, but with different shapes, which means motions are of different trajectories but qualitatively the same. Structure stable autonomous systems maintain its topology structure under perturbations, thus automatically generate adaptive motions that are qualitatively invariant. Qualitative control will preserve the three natural motion features for the following reasons.

Adaptive Different perturbations will generate different motions.

Efficient Motion will be generated passively and follow the least energy path.

Agile Topological structure can be manipulated and maintained with very little computation.

In our research, only the final motion is concerned. In mathematical viewpoint, only the attractors of flows are controlled, while the flow shape is not considered in motion control. According to the types of attractors, motion can be categorized into two groups.

Discrete Motion Such motions have fix point attractors. Typical motions include posture control and picking up motion of the arm.

Periodic Motion Such motion have periodic attractors, typical motion include walking, running and heart beating.

3.2 The New Control Scheme from Qualitative Control Theory

An animal's body and environment can be extremely complex. This usually leads to high dimensional manifolds with complicated topological structure. Many have asked the same question whether such complex system can be controlled with a simple method. Biology Research suggested that the motion is mainly controlled by the Central Pattern Generator (CPG). The existence of CPG is very common, from primitive animals like lamprey and fish, to high level animals like bird, mammal and human (Cohen, 1988). Motor control by CPG can be modeled as entrainment (González-Miranda, 2004). Entrainment is the phenomenon that two coupled oscillator systems synchronize. The phenomenon is universal, for some cases, stability can be enhanced and chaotic behavior can be suppressed.

In this section, we will discuss a new control scheme based on biological entrainment. The neural system forms one electrical oscillator; body and environment form the mechanical one. Mechanical oscillator can be controlled by the oscillation property of the neural system through entrainment.

3.2.1 The Structural Stability of Neural Oscillator

Equation (2) is the extensively studied neural oscillation model developed by Matsuoka (1985).

$$\begin{aligned}
 \tau_1 \dot{x}_1 &= c - x_1 - \beta v_1 - \gamma [x_2]^+ - \sum_j h_j [g_j]^+ \\
 \tau_2 \dot{v}_1 &= [x_1]^+ - v_1 \\
 \tau_1 \dot{x}_2 &= c - x_2 - \beta v_2 - \gamma [x_1]^- - \sum_j h_j [g_j]^- \\
 \tau_2 \dot{v}_2 &= [x_2]^+ - v_2 \\
 y_i &= \max(x_i, 0) \\
 y_{out} &= [x_1]^+ - [x_2]^+ = y_1 - y_2
 \end{aligned}
 \tag{2}$$

Where x and v are state variables, τ, c, β, γ are parameters of the oscillator.

Matsuoka oscillator is autonomous and adaptive; entrainments happen when it is coupled with different oscillators. But because of the nonlinear properties, its behavior has not been completely understood. Matsuoka (Matsuoka, 1987) explained the adaptive properties by investigating location of the roots of characteristic equation. Wilimas (Williamson, 1998) explained the properties in frequency domain.

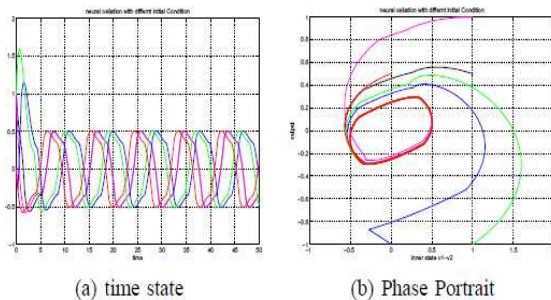


Figure 1. Matsuoka Oscillator

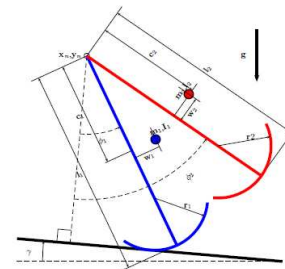


Figure 2. Passive Walker

Here we provide an idea based on structural stability analysis. The topology structure of neural oscillator is simple; it includes one attractive limit circle and one repelling fix point. All the simulations we carried out converged to the same limited circle. In most of the case, the flow will converge to the limit circle within one period. Features above are shown in Figure 1.

These properties are very valuable in CMS research. An intuitive idea is that we can make the motion structural stable by coupling body with Matsuoka Oscillator.

4. APPLICATION AND RESULTS

The basic idea of qualitative control for motion synthesis is boosting structural stability by coupling the mechanical oscillator with neural oscillator. Our approach can be applied to many motion tasks. In this section, we will discuss just one example in details, the bipedal walking. This is mainly because bipedal walking is one of the most challenging and common locomotion styles. Bipedal walking is unstable which makes it very difficult for adaptive gaits. While for artificial system, robust bipedal walking is difficult to achieve. Many control method has been tried, but none of them shows comparable performance with human walking.

In dynamic research, natural looking gaits can be generated by passive method (McGeer, 1990a, 1990b), Passive walkers can walk down a slope without any effort, but the stabilities are very fragile.

From the Qualitative Control Theory, the reason behind passive walking is that there is a limit circle for the dynamic interaction between body and ground. The fragile stability means the basin of attraction is small. To generate adaptive walking gait, we plan to boost the stability of the passive walking machine by neural oscillation entrainment.

4.1 2D Passive Walking Model

The mechanical model we adopted is illustrated in Figure 2. Passive walking is a hybrid dynamic system. We separate the motion into two phases described by Equation (3, 4).

Leg Swing Phase During the swing phases, we suppose that one leg is fixed on the ground, the arc foot makes the passive dynamic walker rolling without sliding (Wisse and Schwab, 2005).

$$\begin{bmatrix} \bar{M} & D^T \\ D & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ F_c \end{bmatrix} = \begin{bmatrix} \bar{F} \\ \ddot{D} \end{bmatrix} \quad (3)$$

Heel Strike Phase We suppose the heel strike the ground in a short time, the angular momentum is preserved, as described by Equation(4) (Wisse and Schwab, 2005)

$$\begin{bmatrix} \bar{M} & D^T \\ D & 0 \end{bmatrix} \begin{bmatrix} \dot{q}^+ \\ f_c \end{bmatrix} = \begin{bmatrix} \bar{M}\dot{q}^- \\ 0 \end{bmatrix} \quad (4)$$

Where \dot{q}_+ is the state variable after the collision, \dot{q}_- is the state variable before the collision.

In the Equations (3, 4), \bar{M} is the inertia matrix, $q = [x_h, y_h, \phi_1, \phi_2]$, D is the constraint equation. F_c is the constrained force, and f_c is the constrained impulse.

4.2 Adaptive Walking Motion

The input of neural oscillator is defined by the difference angle between the two legs. Neural output will drive the biped walker; torque is applied at the hip joint. After adding the neural control, the equation of the dynamic system is as below.

$$\begin{bmatrix} \bar{M} & D^T \\ D & 0 \end{bmatrix} \begin{bmatrix} \ddot{q} \\ F_c \end{bmatrix} = \begin{bmatrix} \bar{F} \\ \ddot{D} \end{bmatrix} + \begin{bmatrix} \bar{U} \\ 0 \end{bmatrix}$$

$$U = [0, 0, 1, -1] * G_{out}$$

Passive Walking When the passive walker walks down a slope, there is an equilibrium condition when the heel strike energy lost is equal to the potential energy input. Because there is no extra control energy input, such motion is the most energy efficient. Figure 3 shows the gait of the passive walker. After coupling the neural oscillator, the basic pattern is not changed as shown in Figure 4.

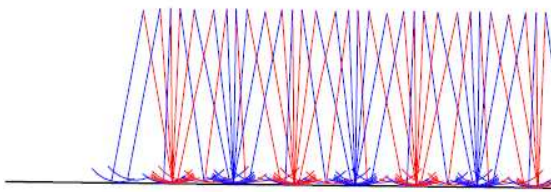


Figure 3. Stable Passive Walking Gait

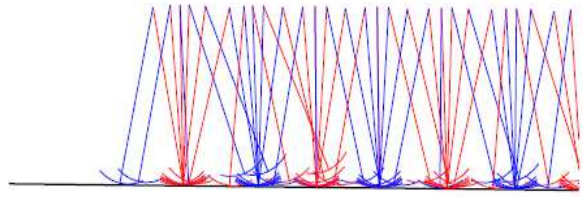


Figure 4. Walk down Slope with Neural Control

Walking On Plane However, the passive walker can't walk on plane. The step size will decrease after each step. Finally it will stop or fall over as illustrated in Figure 5.

After coupled with the neural oscillator, this walking machine can walk on plane, and exhibits gait similar to the passive walker. Figure 6 shows the gait. From Figure 7(a) and Figure 7(b), we can see that the gait converged to a stable limit circle.

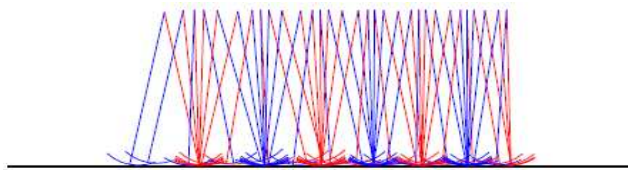


Figure 5. Passive Gait Can't be Maintained on Plane

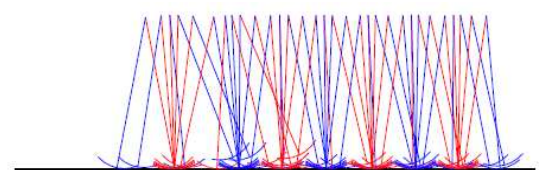


Figure 6. Gait on Plane with Neural Control

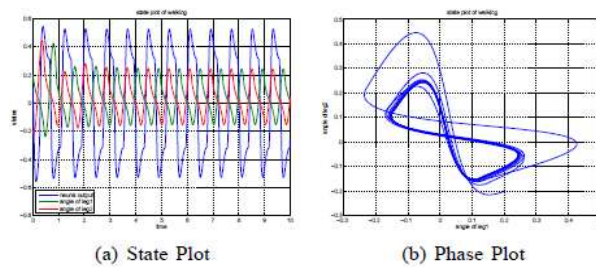


Figure 7. Gait on Plane with Neural Control

To verify the structural stability, we introduce a variety of perturbations to the passive walker.

Different Initial Condition The original passive walker is not very stable. A slight change in initial condition will result in walking failure. While after coupled with neural oscillator, a different initial condition can still lead to a stable gait, as show in Figure 8. This means the basin of attraction has been enlarged.

Walking On Different Slopes Even walking down a steeper slope, stable gaits can still be maintained, as shown in Figure 9. An important discovery is that although the walkers can walk down steeper slopes, it cannot walk up slope, no matter how control parameters are tweaked. We think that this is mainly because the proper limit circle does not exist anymore when walking up slope. Involving the upper body into this structure may help to solve this problem.

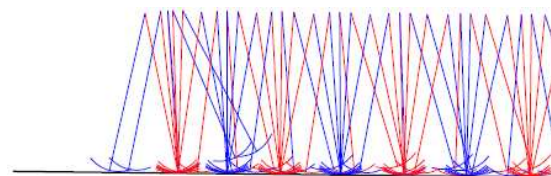


Figure 8. Gait of a Different Initial Condition

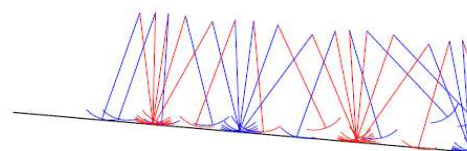


Figure 9. Gait on a Different Slop

Leg Mass Variation Mass of one leg is added up to 150% and the gait is still maintained. The step length and swing period of the two legs are different, this gait is similar to that with a crippled leg, see Figure 10.

Leg Length Variation Leg length is set to 1/8 shorter and the gait is maintained, see Figure **Erro! A origem da referência não foi encontrada.** .

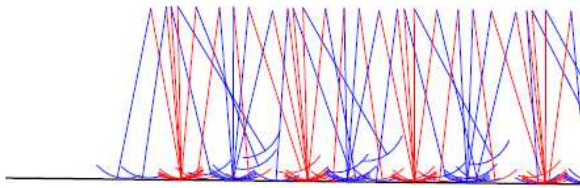


Figure 10. Gait of Different Leg Mass

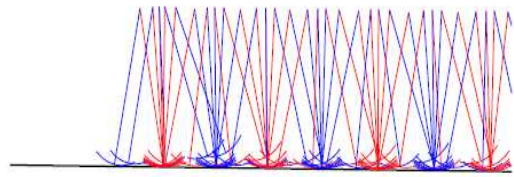


Figure 11. Walking with shorter Legs

5. DISCUSSIONS AND FUTURE WORK

Qualitative Control Theory can synthesize motion with adaptive behavior while keeping the qualitative properties. It provides a new method to synthesize adaptive motion efficiently. Since very little computation involved in each controller, compared with traditional optimization based method, this method can generate motions in real-time. And most importantly, our method is parallel in nature, in future; most of the computational burden of our method can be shifted to GPU. This will make our algorithm generating agile motions even with very complicated environment and involving whole body structures.

However since we bring in a new theory into the motion synthesis area, many works need to be done in the future. For example, our current model only involves the lower body structure, upper body and more joints will be considered in our future design. To proof the robustness, we will need experiment on more complicate terrain and perturbations.

More Central Pattern Generators are needed for different kinds of motions. And how to turn the CPG parameters for the animator purpose are still open. These topics will be covered in the future research.

ACKNOWLEDGMENT

We want to thank Richard Southern for the invaluable discussion and advice about organization of the paper. The knowledge of mechanics and questions of Jian Chang also help to make the paper more clear.

REFERENCES

- F.C. Anderson and M.G. Pandy. Dynamic Optimization of Human Walking. *Journal of Biomechanical Engineering*, 123: 381, 2001.
- AA Andronov and LS Pontryagin. Rough systems. In *Doklady Academy Nauk SSSR*, volume 14, pages 247–250, 1937.
- A.H. Cohen. Evolution of the vertebrate central pattern generator for locomotion. *Neural control of rhythmic movements in vertebrates*, pages 129–166, 1988.
- Michael F. Cohen. Interactive spacetime control for animation. In *SIGGRAPH '92: Proceedings of the 19th annual conference on Computer graphics and interactive techniques*, pages 293–302, New York, NY, USA, 1992. ACM. ISBN 0-89791-479-1. <http://doi.acm.org/10.1145/133994.134083>.
- I. Glynn. *An anatomy of thought: The origin and machinery of the mind*. Oxford University Press, USA, 2003.
- J.M. González-Miranda. *Synchronization and control of chaos: an introduction for scientists and engineers*. Imperial College Pr, 2004.
- Jessica K. Hodgins, Wayne L. Wooten, David C. Brogan, and James F. O'Brien. Animating human athletics. In *SIGGRAPH '95: Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pages 71–78, New York, NY, USA, 1995. ACM. ISBN 0-89791-701-4. <http://doi.acm.org/10.1145/218380.218414>.
- N. Hogan. Impedance control-An approach to manipulation. I-Theory. II-Implementation. III-Applications. *ASME, Transactions, Journal of Dynamic Systems, Measurement, and Control (ISSN 0022-0434)*, 107, 1985.

- Sumit Jain, Yuting Ye, and C. Karen Liu. Optimization-based interactive motion synthesis. *ACM Trans. Graph.*, 28 (1): 1–12, 2009. ISSN 0730-0301. <http://doi.acm.org/10.1145/1477926.1477936>.
- E.A. Jonckheere. Algebraic and differential topology of robust stability. Oxford University Press, USA, 1997.
- Joseph Laszlo, Michiel van de Panne, and Eugene Fiume. Limit cycle control and its application to the animation of balancing and walking. In *SIGGRAPH '96: Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 155–162, New York, NY, USA, 1996. ACM. ISBN 0-89791-746-4. <http://doi.acm.org/10.1145/237170.237231>.
- M.L. Latash. Neurophysiological basis of movement. Human Kinetics Publishers, 2008.
- C. Karen Liu. Dextrous manipulation from a grasping pose. In *SIGGRAPH '09: ACM SIGGRAPH 2009 papers*, pages 1–6, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-726-4. <http://doi.acm.org/10.1145/1576246.1531365>.
- Cheng-Yun Karen Liu. Towards a generative model of natural motion. PhD thesis, Seattle, WA, USA, 2005. Chair-Popovic, Zoran.
- Zicheng Liu, Steven J. Gortler, and Michael F. Cohen. Hierarchical spacetime control. In *SIGGRAPH '94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, pages 35–42, New York, NY, USA, 1994. ACM. ISBN 0-89791-667-0. <http://doi.acm.org/10.1145/192161.192169>.
- Adriano Macchietto, Victor Zordan, and Christian R. Shelton. Momentum control for balance. In *SIGGRAPH '09: ACM SIGGRAPH 2009 papers*, pages 1–8, New York, NY, USA, 2009. ACM. ISBN 978-1-60558-726-4. <http://doi.acm.org/10.1145/1576246.1531386>.
- K. Matsuoka. Sustained oscillations generated by mutually inhibiting neurons with adaptation. *Biological Cybernetics*, 52 (6): 367–376, 1985.
- K. Matsuoka. Mechanisms of frequency and pattern control in the neural rhythm generators. *Biological Cybernetics*, 56 (5): 345–353, 1987.
- T. McGeer. Passive dynamic walking. *The International Journal of Robotics Research*, 9 (2): 62, 1990a.
- T. McGeer. Passive walking with knees. In *Proceedings of the IEEE Conference on Robotics and Automation*, volume 2, pages 1640–1645. Piscataway, NJ, USA: IEEE, 1990b.
- K. Nishikawa, A.A. Biewener, P. Aerts, A.N. Ahn, H.J. Chiel, M.A. Daley, T.L. Daniel, et al. Neuromechanics: an integrative approach for understanding motor control. *Integrative and Comparative Biology*, 2007.
- R. Pfeifer and F. Iida. Morphological computation: Connecting body, brain and environment. *Japanese Scientific Monthly*, 58 (2): 48–54, 2005.
- Zoran Popović and Andrew Witkin. Physically based motion transformation. In *SIGGRAPH '99: Proceedings of the 26th annual conference on Computer graphics and interactive techniques*, pages 11–20, New York, NY, USA, 1999. ACM Press/Addison-Wesley Publishing Co. ISBN 0-201-48560-5. <http://doi.acm.org/10.1145/311535.311536>.
- M.H. Raibert and J.K. Hodgins. Animation of dynamic legged locomotion. *ACM SIGGRAPH Computer Graphics*, 25 (4): 349–358, 1991.
- M.M. Williamson. Exploiting natural dynamics in robot control, 1998.
- M. Wisse and A.L. Schwab. First steps in passive dynamic walking. In *Proceedings of the International Conference on Climbing and Walking Robots*. Springer, 2005.
- Andrew Witkin and Michael Kass. Spacetime constraints. In *SIGGRAPH '88: Proceedings of the 15th annual conference on Computer graphics and interactive techniques*, pages 159–168, New York, NY, USA, 1988. ACM. ISBN 0-89791-275-6. <http://doi.acm.org/10.1145/54852.378507>.

ANIBOT - ANIMATION ROBOT FOR A LIFE

Masashi Furukawa¹, Michiko Watanabe², Ikuo Suzuki¹ and Masahito Yamamoto¹

¹*Hokkaido University, Nishi 9 Kita 14 Kita-ku Sapporo, Japan*

²*Kitami Institute of Technology, 165 Kouen-cho Kitami, Japan*

ABSTRACT

Anibot stands for Animation Robot. It is a concept we have utilized to do research for A-Life. Anibot has virtual sensors, controllers, and actuators, and it evolves itself to acquire suitable behaviors in a given environment obeying the physics law. The acquired behaviors are visualized as animation short movies. The concept of Anibot is not only applicable as a simulator for A-life but also for mechanical designing. This study aims to model Anibot, establish its environment, and acquire its behavior. Some demonstrations prove the concept of Anibot is quite useful to do research for A-life and make an animation.

KEYWORDS

Animation Robot, A-Life, Artificial Neural Network, Physics Modeling, and Evolutionary Computing.

1. INTRODUCTION

Studies on A-life have attracted since the end of 1980. Reynolds [Reynold, C. W., 1987] proposed “bird-oid” objects generally called as “boids”, which is a model of “flock of flying birds”. Boids assumed three forces for the crowd, such as separation, alignment, and cohesion. Separation is the steer force to avoid crowding local flockmates, alignment is the steer force towards the average heading of local flockmates, and cohesion is the steering force to move toward the average position of local flockmates. He successfully made the short movie film of a flock of flying birds. However, the boid only consider the gravity, which is used to define the banking behavior. He did not take the air drag, buoyancy, and lifting force into consideration. Defined three forces were only used for flocking. The boid realized the flock but did not realized how it can fly, because forces given by an environment (air or water) were not calcullllated.

Terzopoulos and et al. [Terzopoulos, D., 1994] made short movie films of the artificial fish, which seemed to be a real fish. Their fish is modeled by springs and sensors, and its motion is determined by a few rules and by solving the second order differential equation. Their work was so impressive at that time. However, their model does not include the water drag and buoyancy given as forces of the environment as well as C. Reynolds.

Sims et al. [Sims, K., 1994] exemplated various artificial lives. They propose a mainframe work for the A-life to acquire its proper behavior. They adopted an artificial neural network (ANN) to control a virtual creature behavior, and an evolutionary computation method is utilized to give ANN output signals for the proper behavior. Their model is simulated based on physics modeling and its result is immediately animated as a movie. Although the creature behavior they showed seems to be real, it is not clear if they considered the environment forces. They proposed an important framework for A-life research, in which their creatures are activated by using physics modeling. His work has been still followed up [Chaumont, N., 2007].

On the other hand, several methods have been developed to calculate the forces given by environments, especially fluid effects in the field of mechanical engineering and computer graphics. These methods essentially solve the Navier-Stokes differential equation. In most of cases, the finite element method (FEM) and the finite difference method (FDM)[Carlson, M., 2004] are employed to accurately solve the differential equation by digitizing a space and object into small voxels. Because the number of voxels is huge, they consume a lot of computing resources. It is said that it takes four to seven minutes per frame for making animation. Also, the moving particle system method (MPS) [Premoze, S., 2003] has been recently used for the fluid simulation. As well as the FEM and FDM, the MPS is a time-consuming method because it needs a

lot of particle elements to solve the fluid effect. These time-consuming methods are not suitable for doing research for A-life and simplification is needed to be able to do simulation in any reasonable amount of time since we need on-line simulation.

This study proposes Anibot, which stands for Animation Robot, to do research for A-life and visualize its behavior as animation movies. The purpose of Anibot is to acquire suitable behaviors meeting the given task. Anibot has virtual sensors, controllers, and actuators, and it evolves itself to acquire suitable behaviors in a given environment. Both Anibot and its environment are modeled to obey the physics law, using physics modeling. Resultant behaviors are instantly visualized by computer graphics animation. There are many applications of Anibot in the field of Computer Aided Design (CAD), Computer Aided Engineering (CAE), Computer Graphics (CG), and Robotics.

The rest of this paper includes the followings: Section 2 describes a research object and a concept of Anibot. Section 3 introduces physics forces, which is not defined in opened physics modeling systems such as ODE [Open Dynamic Engine] and Physx [NVIDIA], into Anibot environments. Section 4 shows some Anibot applications. In Section 5, this study is summarized as a conclusion.

2. MODELING ANIBOT

Sims [Sims, K., 1994] proposed a basic idea that several types of a virtual creature can evolve themselves. His framework is based on physics modeling. The virtual creature is composed of some geometric primitives, joints with some freedom degree and a brain. The brain consists of neurons, which are defined by some arithmetic functions unlikely the sigmoid function. At most three signals are input to neurons. The best creatures are evolved by use of evolutionary computation. Anibot has a similar concept proposed by K. Sims. However, Anibot includes modeling itself and its environment, simultaneously. How Anibot is modeled and what Anibot can do is described in the following.

2.1 Anibot Modeling

Anibot is designed to perform a given intention, planning, task, complicated task (behavior composed), and simple behavior (behavior simple) as shown in Figure 1. We have been climbing the second lower peak now [Yamamoto, M., 2009, Furukawa, M., 2010]. Anibot itself is modeled by a body, sensors, actuators, and controllers. Also, its environment is modeled at the same time. Modeling the environment is described the next subsection. The body is composed of geometrically defined rigid primitives and joints. Two primitives are connected by a joint. Physical properties are added to primitives such as a density, a coefficient of friction, and a coefficient of repulsion. We introduced two types of joints. The one is a general joint, which connects two rigid primitives with some freedom degree. Another is a spring joint. The spring joint is used to model muscles and supple creatures. When the spring joint is not used for modeling, we call it rigid modeling. When the spring joint is used, we call it elastic modeling. An actuator is realized by generating forces to joints. A controller is attached with the actuator. Unlike K. Sims model, we adopt a layers artificial neural network (ANN) and sometimes a recurrent artificial neural network. Sensor signals are input to the ANN and the ANN outputs signals to the corresponding actuator. Weights of the ANN are explored for achieving the given Anibot behaviors (tasks) by use of a real number coded genetic algorithm (RCGA) or a particle swarm optimization method (PSO). We call the resultant evolved ANN EANN. A fitness function is designed so as to achieve the given tasks. It is usually calculated by accumulating moving (grand locomotion, swimming, or flying) distance, which is obtained by Anibot simulation during some time steps.

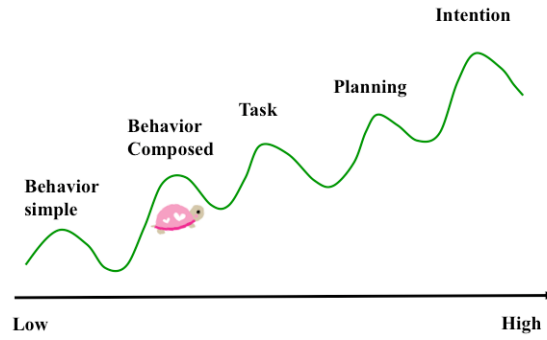
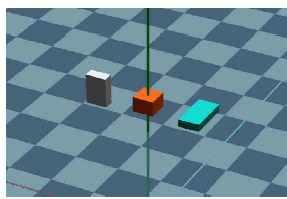
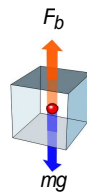


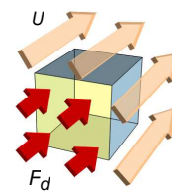
Figure 1. A purpose of Anibot modeling



(a) Floating object by buoyancy



(b) Buoyancy



(c) Water drag

Figure 2. Buoyancy and water drag

2.2 Environment Modeling

An environment plays an important role in Anibot. Anibot's environment is a physical space like a real world on the earth. Anibot is affected by some physical forces such as the friction, gravity, viscosity, buoyancy, water drag, and air drag. A conventional physics modeling system only offers the friction force and the gravity. A virtual creature cannot move on the ground, fly in a sky, and swim in a water without forces given by its environment like a real creature. For this reason, we implement artificial forces, which have effects with Anibot as the buoyancy, water drag, and air drag.

2.2.1 Water Environment

We implement two artificial forces given by the water environment with Anibot. One is the buoyancy and another is the water drag. It is easy to find the center of gravity, using the physics modeling system. We add the artificial force to the center of gravity of Anibot as the buoyancy. The buoyancy is written by

$$F_b = -\rho Vg \quad (1)$$

Where F_b is the buoyancy, ρ_a is the density, V is the volume, and g is the gravity acceleration.

An empirical equation has been used for estimating the water drag in the fluid engineering and it is expressed by (2),

$$F_d = \frac{1}{2} \rho C_w S U^2 \quad (2)$$

Where F_d , S , U , and C_w are the water drag, the representing surface area of the body facing on the moving direction, the relative velocity, and the a drag coefficient, respectively. In applying (2) to the rigid object, the water drag is calculated for only the representing surface. We divide the representing surface into n -pieces of surface (n sub-surfaces) to calculate the water drag more practically. Then, equation (3) is used to calculate the water drag F_{d_i} for the i -th sub-surface.

$$F_{d_i} = \frac{1}{2} \rho C_w s_i u_i^2 \quad (3)$$

Where s_i and u_i are the i -th sub-surface area and relative velocity, respectively. The artificial force F_{d_i} is added at the center of the i -th sub-surface.

2.2.2 Air Environment

Two artificial forces are implemented with Anibot as well as the water environment for the purpose of flying. The one is the buoyancy, which is also written by (1). We can neglect the buoyancy in the air because it is too small to have an effect with flying. The other is the air drag. We use the same equation as (2) to calculate the air drag by replacing a drag coefficient, which is written by

$$F_d = \frac{1}{2} \rho C_a S U^2 \quad (4)$$

Where C_a is the drag coefficient. We divide the representing surface into n -pieces of surface, and then equation (5) is used to calculate the space drag F_{d_i} for the i -th sub-surface.

$$F_{d_i} = \frac{1}{2} \rho C_a s_i u_i^2 \quad (5)$$

2.3 Actuators and Controllers

A Joint works as an actuator with some degree of freedom. As described previously, two types of actuators are prepared. One is the joint which connects two rigid objects with some degree of freedom. Another is a spring which also connects two rigid object. In case of the spring, the degree of freedom is one dimension, namely expansion and shrinking, along one direction. A controller is implemented with the actuator. A layer type of artificial neural network (ANN) is employed for the controller. Environment information through sensors and inner states of Anibot is set to the input layer. The number of hidden layer units (neurons) is determined by an ad-hoc way. The last layer of ANN outputs signals, which are transferred to the actuator.

2.4 Evolving Anibot

Anibot evolves itself to acquire proper behaviors for a given purpose. A real number coded genetic algorithm (RCGA) is employed for evolution. RCGA adjusts a set of weights (synapses) defined between ANN layers. To achieve the purpose, the fitness, which is a function of the energy consumed by Anibot for achieving a given task, is set as the accumulated distance (height) to a given position and/or the posture estimation measured from a given standard posture. The fitness is formalized by

$$Fitness = f(E, D, P) \quad (6)$$

E , D , and P are the energy, the accumulated distance, and the posture estimation, respectively.

3. DEVELOPMENT OF GUI FOR ANIBOT

To do research for A-life using Anibot on a standard platform, we have developed the GUI based physics modeler system, named TIPS-PM standing for Technical Information Processing System Physics modeler. We can model Anibot without physics modeling coding by use of TIPS-PM. Features of TIPS-PM is described in this section.

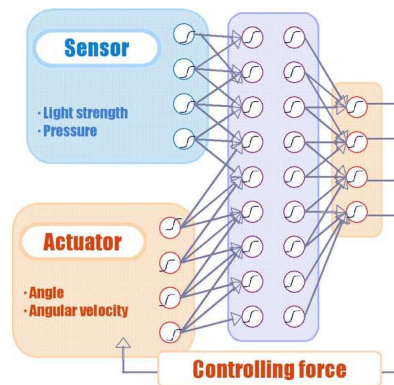


Figure 3. Actuator and controller. The sigmoid function is used for units

3.1 Geometric Modeling

3.1.1 Geometric Modeling

Simple geometric rigid object primitives are prepared for geometric modeling of Anibot. Such primitives are Cuboids, Sphere, and free-formed mesh objects. Figure 4 (a)-(b) shows these primitives. We can define, transform (deform), translate, and rotate primitives, interactively. Physics properties such as the density, the repulsion coefficient, and the friction coefficient, can be defined for primitives as well. Unlike geometric modelers developed for 3D CAD systems, geometric operations such as Union, Subtraction, and Negation are not used to construct complicated geometry. Instead we simply define joints to connect two primitives with some degree of freedom. Two types of joints can be defined. One is a rigid joint which plays a role in hinge, pivot, and coupling with no mass. Another is a spring which is defined by the mass, the natural length, and the elastic coefficient. The joint is also defined interactively.

3.1.2 Editing and Saving

As editing functions, TIPS-PM has copy and save functions. It is possible to copy primitives, jointed primitives, and modeled Anibot. Modeled Anibot is saved in the form of XML data format. It is possible to copy controllers attached with joints described below as well. Once modeled Anibot is saved, we can simulate model's dynamics motion anytime by use of a physics-modeling engine.

3.2 Environment Modeling

3.2.1 Gravitational Field

We adopt PhysX offered by NVIDIA as a physics-modeling engine. Since Physx has a function to generate the gravity acceleration at the center of primitive, Anibot is automatically modeled in the gravitational field. TIPS-PM can adjust the gravity acceleration so that it is possible to make the gravitational field in the moon by setting it as one sixth.

3.2.2 Water and Air Environment

When PhysX calculates model's motion dynamics, it divides primitives into tetrahedrons automatically and triangular meshes are modeled on the surface on primitives. When Anibot moves toward some direction, PhysX returns the relative velocity u_i of Anibot. We can easily extract the representing sub-surface, whose area is s_j in (3) and (5). The underwater environment is constructed by artificially adding the force (3) to the representing sub-surface, where the inner product of its normal vector and the relative velocity is positive.

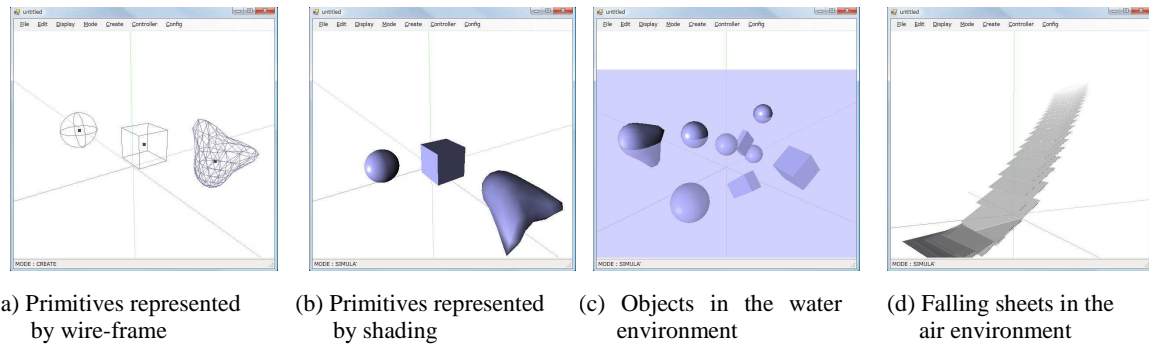


Figure 4. Primitives and environments

The air environment is constructed by artificially adding the force (5) to the representing sub-surface in the same manner. The buoyancy is artificially added at the center of primitives as well. It is possible to neglect the buoyancy in the air environment. Three environments, on the ground, in the underwater, or in the air, can be selected by specifying the environment menu of TIPS-PM. Examples for water and air environments are shown in Figure 4 (c)-(d).

3.3 Sensors and Controllers

3.3.1 Implementing ANN as Controller

Some primitives are selected and specified as sensors. Two types of sensors, the photo-sensor and the touch-sensor, are prepared. It is possible to implement a layered artificial neural network (ANN) with a joint. By interactively specifying the joint, we define the ANN as a controller of Anibot. The numbers of input units, hidden units, and output units can be specified. Sensor information is assigned to the input units. Default values are set to weights between units.

3.3.2 Learning ANN

TIPS-PM cannot implement the fitness function of RCGA for learning the ANN yet. It has the modeling mode and physics dynamic motion simulation mode, and two modes can be switched alternatively. In the simulation mode, model's dynamic motion is visualized as an animation. The model made by TIPS-PM is saved in the form of XML data. When we apply RCGA to controllers to acquire adequate motion, we need to code the fitness function by C++. Modeled primitives, joints, and controllers can be translated into C++ coding by reading the XML data stored in TIPS-PM. Then, the fitness function is defined to realize a given task in C++ coding and RCGA is executed in the same coding. Resultant weights of ANN obtained by RCGA are rewritten to the XML data. This XML data is read by TIPS-PM and results are visualized as the animation by the simulation mode of TIPS-PM. So, we use TIPS-PM as a standard platform for modeling Anibot to do research for A-life.

4. APPLICATION OF ANIBOT

This section presents some applications of Anibot. Also, TIPS-PM applications to Physics education are demonstrated.

4.1 Acquisition of Behavior

Based on the concept of Anibot, many experiments are done for virtual creatures to acquire adequate behavior. These creatures are modeled by use of TIPS-PM. The fitness, which plays a role in a given task, is coded in C++. Some of them are demonstrated in this subsection.

4.1.1 Walking, Crawling, and Jumping

Walking, crawling, and jumping (hopping) behaviors are acquired by use of the friction force between creatures and the ground. Some rhythmic patterns are realized for walking and crawling. EANNs are assigned to legs' joints. Sensor information, neighbor joints' states, and legs' states are assigned to the input layer of EANN. EANN outputs leg's angular velocity.

Figure 5 and 6 show four legs virtual creature's walking and multi-legs creature's crawling. Figure 7 shows the virtual creature jumping like a grasshopper.

4.1.2 Flying and Swimming

Experiments to acquire flying and swimming behaviors are done by use of artificial forces given by the fluid environment described in 2.2.

When a virtual creature has wings, flying is realized by that the flapping-down wing energy is larger than the flapping-up wing energy [Furukawa, M., 2010]. We can search for adequate wings' angular velocities for flapping up and down, which are set to outputs of EANN. Using the same principle, we found that a virtual creature like a fish can swim. Figure 8 shows acquired flying behaviors for the creature with several plural wings. Figure 9 shows acquired swimming behaviors for the fish-shape like creature. Figure 10 shows the butterfly model, which successfully acquires a flying behavior [Ooe, R., 2010].

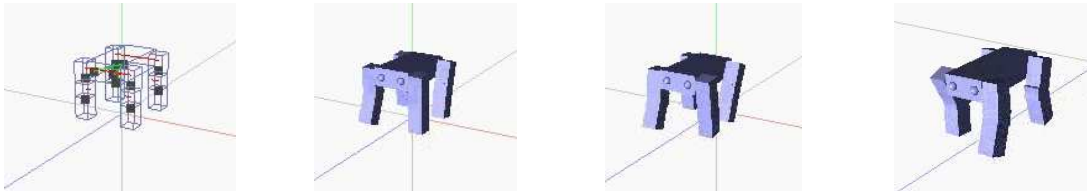


Figure 5. Walking four legs Anibot

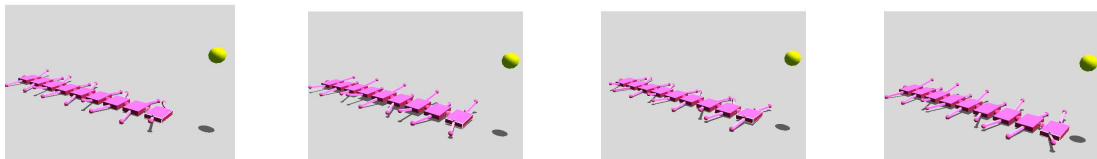


Figure 6. Crawling plural legs Anibot



Figure 7. Jumping Anibot like a grasshopper

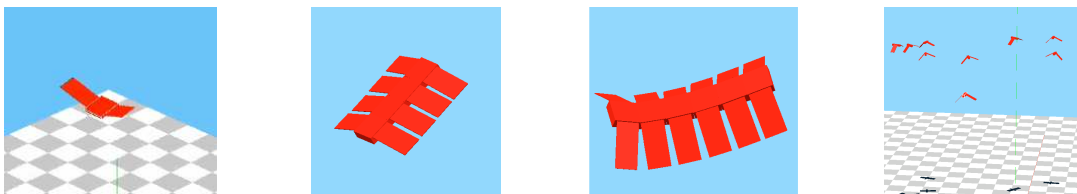


Figure 8. Flying Anibot with two, eight, twelve wings and flying Anibots in flock

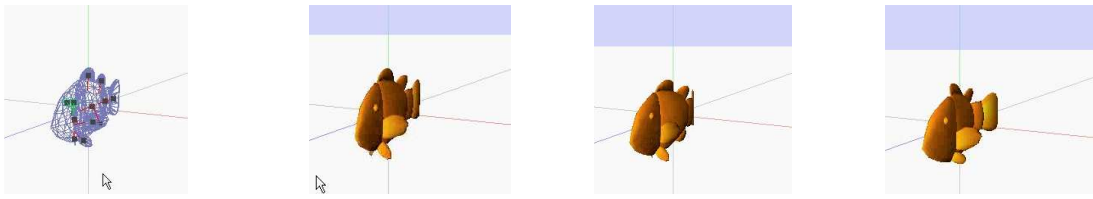


Figure 9. Swimming Anibot like a fish



Figure 10. Butterfly model visualized by texturing

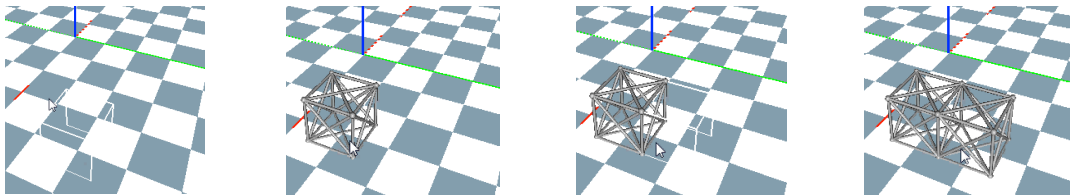


Figure 11. Elastic modeling

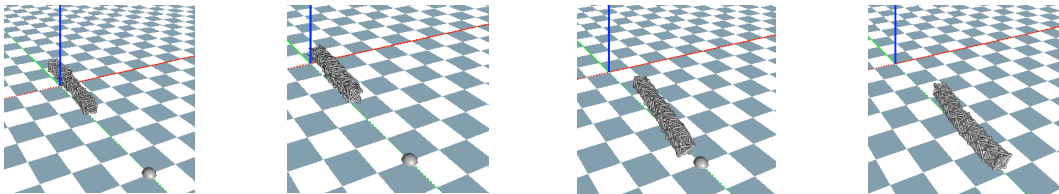


Figure 12. Crawling Anibot by elastic modeling

Anibots shown in Figure 5~10 are modeled by rigid primitives and joints. We can model Anibot by use of rigid primitives and spring joints as well. Figure 11 shows Anibot behaviors by elastic modeling. It consists of spring units. At first, the spring unit is modeled. It is copied and the copied unit is connected to the first spring unit. This procedure is done over. Anibot learns to crawl and reach a light source position by expanding and shrinking springs. Figure 12 shows the resultant behaviors after Anibot evolved itself.

4.2 Educational Application

TIPS-PM is capable of applying Physics education. It is possible for high school students to study and experiment elementary dynamics in simulation without learning any program coding. Physics experiments in the real world request an ideal environment. It is difficult to make the ideal environment. However, the simulation makes the ideal environment with ease, even though on the moon. We apply TIPS-PM to model instances in the physics textbook. A strobe function is added to TIPS-PM for tracing object motion. Figure 13 (a)~(d) show four instances, to which TIPS-PM is applied, referred to the Physics textbook. We taught high school students how to utilize TIPS-PM. After almost an hour, they use TIPS-PM perfectly. Figure 14 shows the simulation done by some student using TIPS-PM.

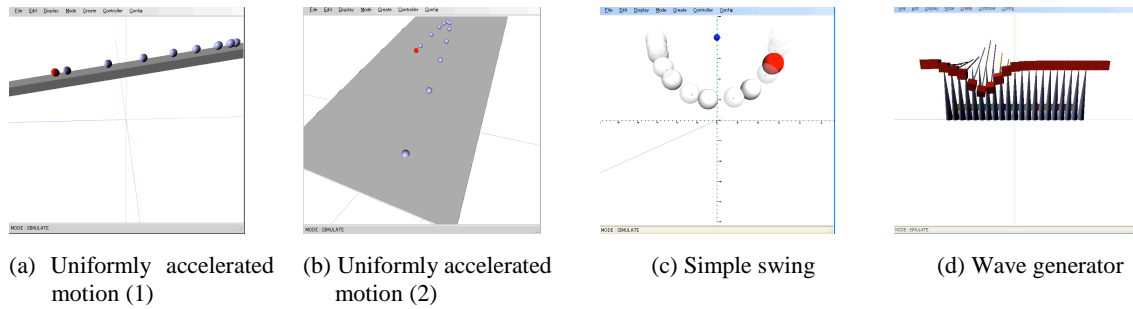


Figure 13. Simulations for instances in Physics textbook

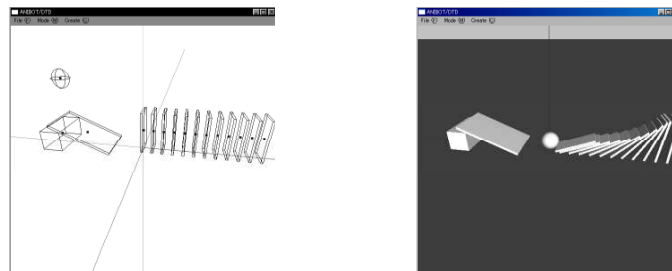


Figure 14. Domino simulation modeled by student

5. CONCLUSION

A concept of Anibot is proposed to do research for A-life. Anibot has sensors, actuators, and controllers. It evolves itself by acquiring adequate behaviors. The framework shown by Anibot is well-known. However, Anibot can behave in the virtual environment, where the gravitational force, air drag force, and water drag force work. The physics engine we used is Physix. As it does not offer air and water drag forces, we present the concise computation method for these forces, using the empirical equation referred from the fluid engineering textbook.

Based on the Anibot concept, we developed the standard Physics modeling tool TIPS-PM. TIPS-PM has capability of modeling Anibot. It has functions for simple geometric modeling, physics modeling, Anibot motion simulation, and making its behavior animation. It is possible to implement artificial neural network as controllers with Anibot by use of TIPS-PM. Several demonstrations verify that Anibot can acquire various types of behavior. Pictures in this paper can be referred to [Movies, 2011].

TIPS-PM is an easy tool that simulate physics phenomenon. As an example, it can be applied to Physics education. Some results are shown, which models and simulates instances referred in Elemental Physics textbook.

Out future work is to achieve much more sophisticated behaviors, such as ones based on planning and intention, for Anibot.

ACKNOWLEDGEMENT

This work is partially supported by Japan Grant-in-Aid for Scientific Research (B), No. 22360099. We appreciate our Anibot project member, Yoneda, K., Iwate, K., Nakamura, K., Ooe, R., Umemura, K., and Ojiri, K., who made impressive animation short movies for this studies.

REFERENCES

- Brooks, R., 1985. A Robust layered control system for a mobile robot. *A.I. Memo 864*, Massachusetts Institute of Technology Artificial Laboratory
- Carlson, M. et al., 2004. Rigid fluid: Animating the interplay between rigid bodies and fluid. *Proceedings of ACM SIGGRAPH*, Los Angeles, USA, Vol. 23, pp 377-384
- Chaumont, N., et al., 2007. Evolving virtual creatures and catapults. *Artificial Life*, Vol. 13, No. 2, pp 139-157
- Furukawa, M., et al., 2010. Behavior Composed for Artificial Flying Creature. *Joint 5th International Conference on Soft Computing and Intelligent Systems and 11th International Symposium on Advanced Intelligent systems (SCIS & ISIS 2010)*, Okayama, Japan, pp. 861-867
- Furukawa, M., et al., 2010. Artificial Flying Creature by Flapping, *2010 IEEE International Conference on Systems, Man and Cybernetics (SMC2010)*, Istanbul, Turkey, pp. 624-629
- Harvey, I. et al., 2005. Evolutionary Robotics: A new Scientific Tool for Studying Cognition. *Artificial Life*, Vol. 11, No. 1-2, pp79-98
- Iwadate, K., et al., 2011. Behavior Emergence of Virtual Creature Living in Complex Environments, *Proceedings of the Sixteenth International Symposium on Artificial Life and Robotics (AROB 16th '11)*, Oita, Japan, GS2-5
- Movies, 2011. <http://junji.complex.eng.hokudai.ac.jp/researches/physics-modeling/movies/CGVCVIP> (accessible from July 1st)
- Ooe, R., et al., 2011. A Physics Modelling of Butterfly's Flight Control by GA and ANN and its Over-Evolution Problem. *Proceedings of the Sixteenth International Symposium on Artificial Life and Robotics (AROB 16th '11)*, Oita, Japan, GS2-3
- Open Dynamics Engine, <http://www.ode.org/>
- Pfeifer, R., 2005. New Robotics: Design Principles for Intelligent Systems. *Artificial Life*, Vol. 11, No. 1-2, pp 99-120
- Physx, <http://developer.nvidia.com/object/physx.html>
- Premoze, S., et al., 2003. Particle-based Simulation of Fluid. *ACM SIGGRAPH/Eurographics Symposium on Animation*, San Diego, USA, Vol. 22, No. 3, pp401-410
- Reynolds, C. W., 1987, Flocks, herds and schools. *A distributed behavioral model*, *Computer Graphics*, Vol. 22, No. 4, pp 25-34
- Sims, K. 1994. Evolving virtual creatures. *SIGGRAPH'94: Proceedings of the 21st annual conference on Computer graphics and interactive techniques*, Orland, USA, pp 12-22
- Terzopoulos, D. et al., 1994. Artificial fishes: Autonomous locomotion, perception, behavior, and learning in a simulated physical world, *Artificial Life*, Vol. 1, No.4, pp 327-351
- Yamamoto, M., et al., 2009. Autonomous Animated Robots: *International Journal of CAD/CAM*, Vol. 9, No. 1, Online
- Yoneda, K., 2010., Emergence of Autonomous Behaviors on Elastic Circular Robot Consisting of Modular Units - Planning for a Task with Composite Behaviors-, *Proceedings of International Conference on Advanced Mechatronics 2010 (ICAM 2010)*, Osaka, Japan, 1A2-3

SIMFOR: TOWARDS A COLLABORATIVE SOFTWARE PLATFORM FOR URBAN CRISIS MANAGEMENT

Jean-Christophe Chambelland, Romain Raffin, Brett Desbenoit and Gilles Gesquière
LSIS laboratory, UMR CNRS 6168 - ESIL Case 925163, Avenue de Luminy, F-13288 Marseille cedex 09, France

ABSTRACT

Risk and crisis management has become an important topic for public and private organizations. Nowadays, an important problem concerns the management of disasters (flooding, fire, nuclear explosion ...) in urban environments. To face such events, different actors like firefighters, mayors, or policemen have typically to deal with a wide amount of data, each at different decision levels. Aggregating various data from various sources to prepare the scenario is then crucial in order to create a credible realistic environment. The project SIMFOR has been initiated to develop a "serious game" oriented towards risk and crisis management in a complex urban context. This game will provide a 3D multi-actor training platform, which will help the players to improve both the prevention and the reaction to critical situations. Several actors can be managed by AI based on multi-agent. Phenomena like fire propagation or transportation net can be simulated by physical models. The first part of this communication will be devoted to the problematic of risk and crisis management. It notably situates the potentiality of "serious gaming" in such problem, which involves to aggregate data in an interoperable way using ISO standards. Part two concerns the SIMFOR serious game prototype. It presents the software architecture and a game session concerning a car fire scenario.

KEYWORDS

Urban crisis management, serious gaming, data interoperability, ISO standards, OGC standards

1. INTRODUCTION

The study of risk and crisis management has become an important topic for most of public and private organization. Nowadays, a particularly important problem concerns the management of disasters (natural, chemical, or nuclear) as well as terrorist attacks in urban environments. While several strategies have been investigated, it remains difficult or impossible to organize most of such situations in real life; several risks are often coupled in a common crisis scenario which is complex to correctly simulate using computed environment. This paper addresses this problem in the framework of the SIMFOR project. Funded by the French government, SIMFOR aims at developing a "serious game" oriented towards risk and crisis management in a 3D urban context.

This document is composed of four parts. The first part (section 2) is devoted to the project context and the project goal. It notably proposes a list of needs like heterogeneous data management, physical simulations, artificial intelligence (AI), real-time multi-actor rendering involved in such project. The second part (section 3) is devoted to an overview of classical strategies used in the domain of risk and crisis management, and to a brief state of the art regarding the needs exposed in the first part. Part three (section 4) presents the actual architecture of SIMFOR, discusses the significant software parts of the game and the data flow involved. Part four (section 5) illustrates a game session in a car fire scenario.

2. CONTEXT AND GOAL

Risk and crisis management is a topic concerning a wide range of structures such as primary school, nuclear power plant, or city halls. In dozens of countries, such structures have to develop risk and crisis emergency plans which have to be regularly tested and updated. A common way to assess these plans is to repeat exercises in real life. Scenarios involved in the test plans usually bring together a large number of trades and

actors which complicate the synchronization of actors. The example of a risk/crisis scenario in a school is a good illustration. Such a scenario will lead actors such as teachers, the mayor and his team, emergency service, and police to act together in a same issue. In practice, it is really difficult to gather all the actors and trades which take place in the scenario. Moreover, studying risk shows that it is often unrealistic to consider isolated risk at a time: a forest fire will involve cutting off electricity in some large areas which may entail critical problem in neighboring hospitals.

The SIMFOR project aims at handling all these aspects in a single software platform. The goal is to design a 3D multi-actor training platform which must help people to improve the prevention and the reaction to a wide set of urban critical situations (flooding, fire, buildings collapse, nuclear accident, etc.). This platform will not be devoted to a few numbers of experts belonging to a specific domain. It will be compatible with a wide set of scenarios coming from various trades. The relative software will allow real or AI-driven players to progress relatively to their ability, and should allow users to fit into different roles within the same scenario. To enhance realism, the platform will also be linked with a wide number of physical models (fire, water flooding, etc.). The multi-player aspect is crucial within the game because it conditions the exchanges between actors. Moreover, even if generic environments can be useful for basic training, the platform must allow designing new environments from a devoted editor which would be able to deal with realistic heterogeneous remote data. This editor, named *environment editor*, will allow aggregating a wide amount of data to prepare new simulations. The scope of SIMFOR is very large, and a single communication does not allow to give deep details about all the aspects involved in SIMFOR (multi-actors, physical simulations, IA, 3D rendering, heterogeneous data merging, etc.). In this way, mostly the overall architecture of the platform and the problem of data exchanged involved within the environment editor is aimed in this paper.

3. PREVIOUS WORK

3.1 From a Real Life Experience to a Serious Gaming Based Approach

The study of risk and crisis management has become an important topic for most of public and private organizations. Risk and crisis generate an operation pipeline requiring the identification of the risk, the development of strategies to reduce it, the creation of directives to put these strategies into effect, and also some processes to handle crisis if preventive actions fail. Several specific methodologies have been proposed for modeling such operation pipelines ([Kara-Zaitri, 1996]), but there are basically three ways to assess their practical reliability: real-life experience, computer-aided simulation, and more recently, serious gaming based approach.

For most of small-scale crisis scenario (e.g. a fire in a building), real-life experiences ([Militello et al, 2004]) can be efficient, because they are a good way to validate plans and test preparations for a forthcoming emergency. However, the latter are in general very expensive, time consuming to organize and to execute, and also impossible to restart at an intermediate point, in order to try an alternate action or plan. Moreover, for large-scale crisis situations such as natural disaster (flooding, earthquake), real-life experiences are known to be very difficult or impossible to train because they require too much people with different responsibility level and critical resources to participate.

Computer-aided simulations (CAS) represent a significant improvement to real-life simulations ([Kleiboer, 97], [Simonovic and Ahmad, 2005], [Matejcek et al., 2006], [Fiedrich and Burghardt, 2007]). They are easier and less expensive to execute repeatedly, to test different approaches and decisions, and can be successfully applied to large-scale simulations including very complex rules to drive the actions and reactions covered by the simulation. However, if one set of rules can finely model a phenomenon, CAS are often limited to one kind of event and thus do not integrate the crucial multi-actor aspect.

In recent years, a new computer-based approach allowing handling real-life problems ([Stolk et al., 2001], [Haferkamp and Kraemer, 2011]) has emerged. This approach, called "Serious Gaming" (SG) has become popular in many fields such as education, business, safety, or tourism ([Kankaanranta and Neittaanmaki, 2009]). Basically, SG is founded on the union between simulation and video games technologies. This union is not new for many big organizations (army has used them as a part of training and planning for decades), but

the rise of SG-based planning and training is due to the recent breakthrough of computer games software and hardware. For less than ten years, SG has become a growing tool exploiting the advantages of both real-life and CAS to offer high level simulation capabilities such as real-time human interaction during the runtime and realistic environment. Nowadays, SG technologies allow combining the experience of a real-life simulation, the complexity and ease of reproduction of the CAS, while generally providing a 2D or 3D interactive graphics user interface. Another major contribution of SG is the potential multi-actor support that allows incorporating communication and cooperation of players inside the simulation. Most of all, SG brings a ludic way to handle problems with can be toilsome to approach with classical real-life or CAS. Largely inspired from entertainment games technologies (e.g. www.pjb.co.uk/games-resources-examples.htm), recent technologies for developing SG ([Iuppa and Borst, 2009], [Kankaanranta Neittaanmaki, 2009]) support 2D/3D geometries and textures, 3D sound, physical simulation, AI, network traffic, environment, and multi-players. Most of SG are based on so-called SG-engines which also integrate real-world constraints and contextual data. Some of them are able to track player behavior and assesses their ability, or to provide instant replay.

A lot of games have been developed for training, but a few of them are focusing on risk management. For instance, the serious game Moonshield by KTM is related to risk management. However, this game is only single player. The company EMI has developed a simulation package used for firefighters training but the solution is very focused on the forest fire prevention and requires the participation of specialists to prepare the scenario and the operating theater. The company Infoterra has developed a tool for risk management and crisis situations named RISKFRAME. This tool is based on a GIS covering the crisis cycle (prevention, during and after the crisis). However, it does not offer a 3D visualization module and is not coupled to a training system. This limits the disposal of the actors. Consequently, creating a SG-based training in risk management with a comprehensive tool integrating multi-actor, real personalized environments, scalable and realistic scenarios, and network gaming remains a challenging topic.

3.2 Preparing Operation Theater

Even if generic environments can be useful for basic use, a serious game for training must allow designing new playgrounds from a devoted editor. This one would be able to deal with realistic heterogeneous remote data. Importing, representing, and exchanging heterogeneous data is then an important problem. In the platform described previously, three types of data are mainly involved: geospatial, BIM (Building Information Model) and CAD (Computer-Aided Design) data. Using these data to prepare the environment game implies to take into account interoperability to facilitate their access and agglomeration process. Interoperability may be defined as "the ability of two or more systems or components to exchange information and to use the information that has been exchanged". In a world where software vendors have implemented various products tailored to the needs of specific communities and/or customers, standardization is the simplest and most efficient solution to interoperability problems ([Zhao, and Di, 2010]).

Many organizations, industry consortia and specific groups are involved in standards development activities related to urban matters: ISO/TC 211 - Geographic Information/Geomatics and the Open Geospatial Consortium (OGC) work on standards for geospatial information and services ([Sample et al., 2007], [François et al., 2010]); the building SMART consortium (formerly International Alliance for Interoperability, IAI) focuses on developing standards for the construction and facility management industries; The Web3D Consortium is concerned with standards for 3D data exchanged over the Internet. Taking advantage of standards that allow joint exploitation and combination of various geospatial and CAD data is a requirement for developing an interoperable platform, for which there is an increasing demand in many user communities. It will bring an easy way to prepare operation theaters and can be focused on particular virtual cities.

Some well-known standards in GIS, based on fully-described XML formats, are GML and CityGML ([Kolbe, 2009]) and WMS/WMS web services. GML (Geography Markup Language) is an XML grammar defined by OGC to express geographical features. It serves as a modeling language for geographic systems as well as an open interchange format for geographic transactions on the Internet. The ability to integrate all forms of geographic information is key to the utility of GML. CityGML is a common information model for the representation of sets of 3D urban objects, implemented as an application schema for GML3. It defines

the classes and relations for the most relevant topographic objects in cities and regional models with respect to their geometrical, topological, semantic and appearance properties. Generalization hierarchies between thematic classes, aggregations, relations between objects, and spatial properties are included. This thematic information goes beyond graphic exchange formats and makes it possible to employ virtual 3D city models for sophisticated analysis tasks in different application domains like simulations, urban data mining, facility management, and thematic inquiries ([Hagedorn and Dollner, 2007]). WMS (Web Map Service)/WFS (Web Feature Service) are standard protocols for serving georeferenced maps/entities over the Internet that are generated by a devoted server. Those standards have been considered as really interesting tool to ensure the game interoperability, because they allow managing most of the geographical, architectural, and semantic data needed for the game.

These standards are obvious to aggregate in an automatic way the heterogeneous data and to ensure the life of our application. Nevertheless, it is necessary to prepare the data to use them in a real-time environment. For instance, using cityGML file of several hundred of Mb based on XML is not realistic. This requires to prepare data in an automatic way to create a spatial data infrastructure (SDI) devoted to computer games. This SDI will bring an innovative way to prepare a new environment. The following part discusses this topic, which is one of the main components of SIMFOR.

4. SIMFOR

4.1 Towards an Interoperable Collaborative Platform

The goal of SIMFOR leads to a set of functional specifications which raise several research and technical points, conditioning the success of the project. The first one concerns the design of a 2D/3D urban environment from both quantitatively and qualitatively realistic data. This involves to manage the import, the representation, and the exchange of heterogeneous data such as geographical, architectural (BIM), and semantic data with well-suited technologies. The second point concerns the simulation of various crisis phenomena (e.g. fire, flood) and means integrating physical models into the game. The third point concerns the management of computer controlled agents and raises the problem of integrating AI processes into the scenarios. This part made by an other team of the SIMFOR project, will let to achieve cognition and agents communications. It will not be presented there. The last point in relation to the success of SIMFOR concerns the 3D interactive visualization of the game environment, and involves to be able to display a complex realistic 3D scene and to interact with some elements in real-time. This point is closely linked to the first one because it highly depends on the way the data are structured and communicated to the display environment. The state of the art exposed in section 3 allowed us to explore the panel of potential tools suitable for designing the SIMFOR platform. An overview of SG technologies has shown us that actual tools seem able to fulfill most of the technical points conditioning the success of the project. However, the huge amount of geographical, architectural, and semantic data required for the SIMFOR platform shows the lack in actual SG technologies, which are not directly compliant with geospatial and CAD standards. The SIMFOR platform has been designed to integrate this aspect through the general architecture illustrated on Figure 1.a. In its actual form, SIMFOR is composed of three logical parts: the resource, settings, and runtime parts.

The resource part encompasses all the data which are needed for the game. These data concern scenario files (files describing what type of crisis situation the players will have to face), entity files (describing real-life objects that will appear in the scenario), and environment files (describing the geographical and architectural environment of the scene). Scenario files and entity files are XML files structured with non-standard proprietary grammars. Environment data can be stored in various CAD and geospatial ISO or *de facto* standards. The file formats supported are for example: MIF/MID, 3DS, shapefile, COLLADA, or CityGML.

The settings part relies on an editor which allows configuring a game session from a graphical user interface (GUI). This editor permits a direct import of a scenario and entities from their corresponding files. It also allows importing environment files from OGC WFS /WMS requests and GML answers through a Normalized Data Exchange Services (NDES) exposed on Figure 1.b. This service (described in the next part)

allows serving various files formats in GML files to the editor which is able to interpret the results and integrate the data in a 3D scene.

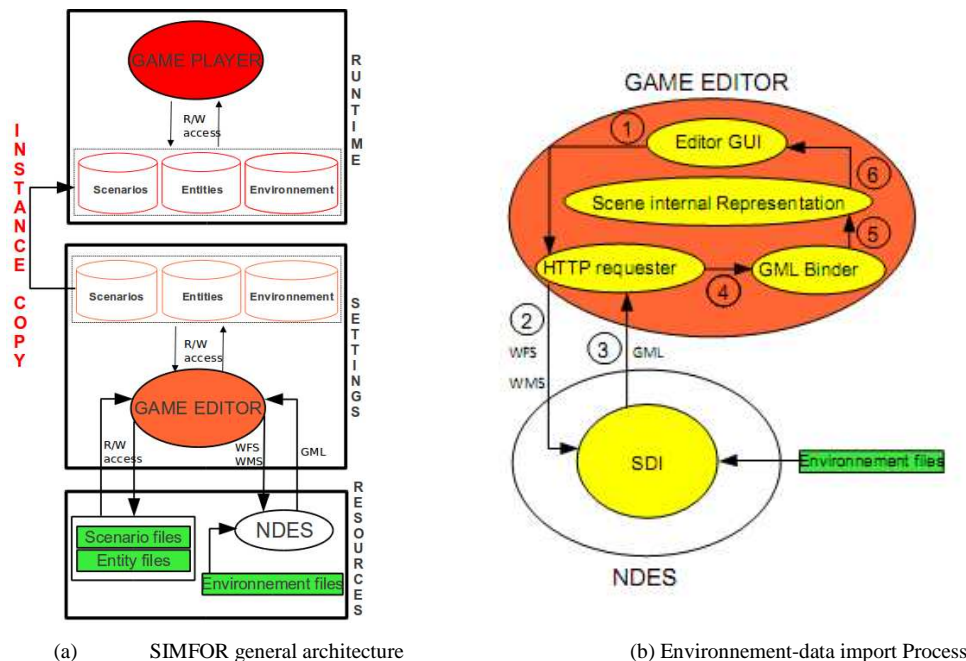


Figure 1. SIMFOR main components

Once the scene is completed, an export of its components (scenario, entities, and environment data) to well-structured databases is achieved. The databases are duplicated in each actor world to ensure data coherence and avoid the modification of the initial structure of the simulation. The data transfer latency during the simulation is also excluded by duplication. These databases are then the support of the runtime part which relies on the game player. This last provides a GUI from which the user can join a game session. The runtime uses the information stored in the databases to make the game progress.

4.2 Resources

Scenario and entities are specified using XML grammars. Instance files are used to define the disaster type, actors, and means involved in the scenario. The environment files are served by the NDES. In its actual version, the NDES is based on an opensource spatial data infrastructure (SDI) as illustrated on Figure 1.b. This SDI contains a comprehensive geospatial software package with implementations of OGC Web Services like WMS and WFS, and various tools for geospatial data processing and management (analysis tools for example). It allows serving a wide range of file formats and notably CityGML files which is a major advantage for SIMFOR.

The NDES simply encapsulates this SDI and is considered has a remote entity which can be acceded by the game editor from standardized requests such as those provided for instance by WFS and WMS. In return, the NDES is able to provide geographical data to the editor in a standard GML form.

4.3 Settings

The import of data in the game editor is achieved from direct files reading for scenarios and entities, and from WFS/WMS requests coupled to GML answers for environmental data. For this purpose import procedures of massively used formats has been achieved as well as GML analyzing tools. To insure spatial data coherence, data has been expressed in WGS84 projection.

GML analyze has been based on a binder which allows mapping XML grammars to C++ classes. This technique is more suited for wide grammar than its homologue SAX or DOM approaches based on XML

instance analyzing, and provides an easy way to read and write XML instances based on arbitrary grammar. The operation pipeline allowing the editor to import environment data is proposed on Figure 1.b. The simulation designer selects an entity he wants to integrate into the scene (1), the underlying WFS/WMS HTTP request is sent to the NDES through a devoted requester (2), the NDES send back to the requester the GML corresponding to the selected entity (3), the GML flow is sent to the binder which produces a C++ object (4), the C++ object is linked to the internal representation of the scene (5), finally the entity is displayed in the editor GUI (6).

The scene editor has been developed as a plugin of Pxviewer, software from Pixxim partner, devoted to urban environment design. The scene edition is based on a layer system which allows gathering components of the scene by their functional aspects. More significant layers are Digital Elevation Model (DEM), building, and road layers which are the base of most of crisis scenarios. For each layer, the environmental components are automatically proposed to the game designer by requesting the NDES on its serving capabilities. The user can then choose objects he wants to integrate into the scene by activating devoted check boxes on the GUI. The designer has the possibility to move elements and enrich the semantic aspect of the scene objects with classical picking functions.

Once the setting part is achieved, the designer has to export the scene towards the game player to launch a game session. For this purpose, a set of export procedures has been implemented in the editor to transfer the C++ objects (DEM, buildings, roads, etc.) of the scene to spatial relational database tables. Spatial databases are optimized to store and query data that is related to objects in space, including points, lines, polygons or more complex entities. Tables are suitably structured to contain minimal geometric and semantic data.

Each relevant entity of the scene can be geometrically represented by its bounding box, its unique identifier and a point representing its location on the scene.

4.4 Game Player

Last module of the project, the runtime is based on a GUI from which a user can launch or join a game session. In its actual version, the SIMFOR player GUI relies on the Delta3D (D3D) game engine (www.delta3d.org/). D3D is an open source engine which can be used for games, simulations, or other graphical applications. In addition to basic graphical components, D3D provides a variety of tools such as the "Simulation, Training, and Game Editor" (STAGE), a BSP Compiler, a particle editor, a stand-alone model viewer, and a "High Level Architecture" (HLA) Stealth Viewer. Furthermore, Delta3D has an extensive architectural suite that is integrated throughout the engine. This suite includes frameworks such a "Dynamic Actor Layer" (DAL) for actor proxies and properties, signal/slot support for direct method linking, a "Game Manager" (GM) for actor management, pluggable terrain tools for reading, rendering, and decorating terrain, and high-level messaging for actor communication.

In its actual form, the Game player GUI provides both a first and third person control of an avatar that the user choose at the beginning of the game. The GUI provides some communication functions (phone, fax, radio) which allow a player to communicate with other players during the game session and a 2D map of the simulation area to know where he is on the overall scene. The information used during the game session are dynamically updated from the spatial database. The spatial capabilities of the databases are used for answering to geometrical constraints involving the scene entities.

5. EXPERIMENTAL RESULTS

This part illustrates a simple use-case of SIMFOR process: a car fire event taking place at Lunel (small size town). Basically, this scenario involves a car which starts to burn on a parking and a firefighters crew who have to stop the fire in the shortest delay. This simulation also includes a witness which contacts the fire department and the mayor of Lunel. The disaster type, actors, and means involved in the scenario have been respectively specified by XML files similar to those given on Figure 2.

```

<disasters>
  <disaster
    label="CarFire"
    type="fire"
    origin="natural"
    state="initial"
    location="parking1"
    ...
  />
  ...
</disasters>

<actors>
  <actor
    name="John..Smith"
    id="firefighter1"
    skill="good"
    function="firefighter"
    mesh="firefighter.xml">
    ...
  </actor>
  ...
</actors>

<means>
  <vehicule id="01"
    mesh="car.ive">
    ...
  </vehicule>
  ...
</means>

```

Figure 2. XML files for disaster type, actors, and means involved in the scenario

The listings define that in the Lunel car fire scenario, the fire is classified of "natural" origin and takes place on "parking1". The firefighter "John Smith" is involved (cause of his "good" skill) to stop the fire started on vehicle "01". The environment data of the Lunel town, gathered via the environment editor, are shown on Figure 3.a. The car which is burning appears on Figure 3.b. This image uses the "Firefighter" view of the simulation. Each character, via his own specific view, can act on the simulation, depending on its responsibility rank (city mayor, medical rescue, public organization head, etc.). Other risks, for instance explosion of inflammable material, can be added to this scenario.



Figure 3. SIMFOR use case : car fire event in Lunel

Several types of entities have been exported from the editor (cars, trees, buildings, streetlights, etc.) and structured in a spatial database table. Each entity is linked to a unique identifier and several geometric (e.g. absolute position, 2D hull, etc.) and semantic data. It also describes its state, which indicates through an integer value if this entity is safe, under fire, or completely burned. Those states are linked to specific Uniform Resource Locator (URL), pointing on precalculated IVE files (www.openscenegraph.org/projects/osg) locally stored on the player machine. These IVE allow updating the graphical rendering of the scene entities according to their relative state in the database.

6. CONCLUSION

"Serious Games" seems to be particularly appropriate for training people to function in complex situations and as members of a team. SIMFOR has been initiated in this goal, as a multiplayer interactive game allowing to train individuals and teams in order to improve their skills. A promising prototype has been developed to allow players to train with basic environments and scenarios. Many aspects of the project

remain under construction. One important part concerns the file formats which are not yet supported by the SDI used in the NDES such as CAD data, and thus which can't be served in GML files. Until now, these data are loaded into the game editor via specific import methods, which should be replaced in a near future. This point involves investigating the possibility of transferring/converting various CAD data into the GML formalism.

An interesting way can be to use interoperable CAD formats, also based on XML, and perform a dictionary translation (U3D, 3DXML). This led up to geometric translations, as for example, parametric splines (usually Nurbs) to polylines. Despite the ISO19107 definitions (ISO19107, 2003) of geometric objects, there is some lacks in the usage of parametric curves and surfaces. A recurrent problem is also to express these geometries in different space representations (Euclidean, spherical, ellipsoidal, and geodetic). The communication loss, sometimes critical in such crisis situation have not been yet implemented. Standard HLA communication would permit spooling and dead-reconning of messages.

ACKNOWLEDGEMENT

SIMFOR is funded by the French Industry ministry on the period 2009-2011. The LSIS lab thanks the industrial partners of the project Pixxim (<http://www.pixxim.fr>) and SII (<http://www.groupe-sii.com/fr>) for their support to this communication.

REFERENCES

- Fiedrich, F. and Burghardt, P., 2007. Agent-based systems for disaster management. *In Communications of the ACM-CACM*, Vol. 50, No. 3, pp. 41-42.
- Francis, A. et al, 2010. Geometric data structures and analysis in GIS: ISO 19107 case study. *Proceedings of the 5th ISPRS International Conference on 3D GeoInformation (XXXVIII-4/W15)*, pp. 115-120.
- Haferkamp, N. and Kraemer, C., 2011. Training disaster communication by means of serious games in virtual environments. *In Entertainment Computing*, article in press.
- Hagedorn, B. and Dollner, J., 2007. High-Level Web Service for 3D Building Information Visualization and Analysis. *Proceedings of the 15th ACM International Symposium on Advances in Geographic Information Systems (ACM GIS), Seattle*.
- ISO TC 211, ISO 19107 Geographic information -- Spatial schema, ISO, 2003.
- Iuppa, N. and Borst, T., 2009. *End-to-End Game Development*. Focal press.
- Kankaanranta, M. and H., Neittaanmaki, P. (Eds.), 2009. Design and Use of Serious Games, Intelligent Systems, Control and Automation: Science and Engineering. Springer.
- Kara-Zaitri, C., 1996. Disaster prevention and limitation: state of the art; tools and technologies. *In Disaster Prevention and Management*, Vol. 5, No. (1), pp. 30-39.
- Kleiboer, M., 1997. Simulation Methodology for Crisis Management Support. *In Journal of Contingencies and Crisis Management*, Vol 5, No. 4, pp 198-206.
- Kolbe, T., H., 2009. Representing and Exchanging 3D City Models with CityGML. *In Lecture Notes in Geoinformation and Cartography, 3D Geo-Information Sciences*, pp. 15-31.
- Matejcek, L. et al., 2006. A GIS-based approach to spatio-temporal analysis of environmental pollution in urban areas: A case study of Prague's environment extended by LIDAR data. *In Ecological Modelling*, Vol. 199, No. 3, pp. 261-277.
- Militello, L., G. et al, 2004. Information flow during crisis management: challenges to coordination in the emergency operations center. *In Cognition, Technology & Work*, Vol. 9, No (1), 25-31.
- Sample J., T., et al., 2007. *Geospatial Services and Applications for the Internet*. Springer.
- Simonovic, S. P., and Ahmad, S., 2005. Computer-based Model for Flood Evacuation Emergency Planning. *In Natural Hazards, Vol 34, No. 1, pp 25-51*.
- Stolk, D., et al., 2001. Gaming and multimedia applications for environmental crisis management training.. *In Computers in Human Behavior*, Vol 17, No. (5-6), pp. 627-642.
- Zhao, P., and Di, L., 2010. *Geospatial Web Services: Advances in Information Interoperability*. IGI Global.

A FRAMEWORK FOR IDENTIFYING USER SUITABILITY TO VISUALISATION APPROACHES

Christopher Moir and G Stewart Von Itzstein
University of South Australia

ABSTRACT

Visualisation is a key approach to providing quick understanding of complex concepts. Presently very few visualisation systems attempt to understand the user in order to support the selection of the most appropriate visualization for the user and the task at hand. This paper will convey the requirements for an approach to address this deficiency. We will then introduce a diagnostic model that can enable a visualisation system to aid the recommendation of effective visualisations to the user. The components of the model will be described and the underlying empirical and theoretical support will be described.

KEYWORDS

Visualisation, Diagnostic Model, Eye Movements, Self-Efficacy

1. INTRODUCTION

Visualisations enhance our ability to make informed decisions by visually presenting information in a format that allows us to readily comprehend trends and patterns. As the rate of data acquisition and storage increases, our need to quickly and efficiently interpret this information grows. Yet, while we have become very good at collecting information we are not nearly as good at effectively using this information. To some extent we have already gathered an extremely large number of visualization strategies that facilitate rapid comprehension of information for a particular task and user. Unfortunately there is no real method of knowing what the right visualization is for a particular user or knowing if the user truly understands what they see.

What makes a visualisation effective is still not universally agreed upon. Some researchers argue that a visualisation is effective when it matches the structure of the data (Cleveland 1985; Mackinlay 1986; Wattenberg and Fisher 2004). Others point out that a visualisation is effective when it directly supports the viewer's task (Casner 1991; Domik and Gutkauf 1994; Nowell, Schulman et al. 2002). Another view is that the interplay between the viewer's mental representation of a visualisation, and the visualisation as an external representation, determines its effectiveness (Lowe 1993; Scaife and Rogers 1996; Hegarty 2004; Trafton, Trickett et al. 2005; Tversky, Agrawala et al. 2007). Finally, other researchers posit that a viewer's working memory capacity, domain knowledge, experience and visual-spatial skills also influence whether or not a visualisation is effective (Kosslyn 1989; Petre and Green 1993; Lowe 1994; Lohse 1997; Freedman and Shah 2002; Allen, Miller Cowan et al. 2006).

Our view is the user's perceptual capabilities and understanding of the meaning of the spatial arrangements and the entities (the shapes, symbols or icons) in a visualisation will significantly influence that visualisation's effectiveness. A viewer's cognitive load is likely to be reduced if they can readily comprehend the meaning of a spatial arrangements and entities in a visualisation. It is reasonable to conclude that a visualisation that is suited to a particular data type and task is still unlikely to be effective if the viewer cannot:

1. Detect and discriminate between the entities in a visualisation.
2. Understand the intended meaning of the spatial arrangements and entities in the visualisation.

Neisser's Perceptual Cycle (1976) further supports this view. According to Neisser, perception of a visual scene is a constructive process in which a person's cognitive schema (existing knowledge) directs their exploration of a scene. So it seems unlikely that an individual will accurately comprehend the information

that a visualisation is displaying without the knowledge required to do so. For these reasons we argue that a visualisation's effectiveness is influenced by the viewer's perceptual abilities and their knowledge about how a visualisation represents the data.

If one accepts the view that the effectiveness of a visualisation is affected, at least in part, by the viewer's perceptual abilities and knowledge about the visualisation, then it is logical to conclude that a visualisation system should include information about the user's perceptual abilities and their knowledge about particular visualisations. In this paper we present a high-level description of our User Model for effective visualisation that records the user's perceptual abilities and knowledge about different visualisations. To ensure that the User Model contains an accurate representation of the user capabilities and knowledge, we also propose a Diagnostic Model whose function is to update the state of the User Model as more evidence about the capabilities and knowledge of the user become available.

Our User and Diagnostic Models were designed to extend the Imago Visualisation Framework (Vernik, Von Itzstein et al. 2007) to include our view of what makes a visualisation effective for a particular user. Imago is a framework for prototyping, evaluating and deploying visualisation approaches. In this paper we will describe the User and Diagnostic Models that we have developed for Imago, along with the empirical results we have obtained that support our Diagnostic model.

The contributions of this research are:

1. Designed a User Model to enhance the effectiveness of the Imago Visualisation Framework
2. Implemented and tested a Diagnostic Model for building and updating the User Model.
3. Conducted an experiment to establish if self-efficacy beliefs correlate with reality.
4. Conducted an experiment to establish if there is a correlation between eye movements and understanding of a specific visualisation.

This paper is structured as follows: Section 2 provides a short overview of the Imago visualisation environment. Section 3 details the high level design of our proposed user model for effective visualisation. Section 4 describes how our proposed user model acquires information about a viewer. Finally conclusions are specified in Section 5 followed by future work in Section 6.

2. THE IMAGO VISUALISATION ENVIRONMENT

The Imago environment (Vernik, Von Itzstein et al. 2007) comprises several key features which combine to support the characterization, prototyping, evaluation and transitioning of information visualisation approaches for particular contexts of use. The pertinent features of the environment are: the semantic model and the evaluation framework. The system has been designed so that information could be captured on how and when various views are used by users through data provided by biological and neuro-physiological sensors.

The Imago environment draws from, extends and implements the Reference Model for Visualisation (RM-Vis) concepts (Gouin, Evdokiou et al. 2002). The core element of the Imago environment is a Semantic Model that provides the central knowledge base of domain contexts, descriptive aspects, viewpoints, and visualisation approaches. For example, the environment supports the modeling and capture of domain contexts based on the roles and activities undertaken by users. Support is also provided for capturing user viewpoints, which define what needs to be described in relation to particular tasks being undertaken within a domain context. Given this knowledge of user requirements, the system is able to suggest various views that support their specified viewpoints. User and team characteristics can be captured in the model. These characteristics include physical, cultural, interpersonal and cognitive attributes. The semantic model also supports the integration of evaluation and instrumentation data in relation to user contexts and associated visualisation approaches.

The semantic model stores the various semantic relationships, evaluations results, and instrumentation data. This provides the basis for the intelligence of the system. The central concept in Imago is the viewpoint that relates the views to a particular tasks being performed within a domain context and the aspects that need to be described. Views are the actual visualisations provided by the environment. The remaining Imago concepts (roles, activities, descriptive aspects, enhancements, interactions, representations, users, teams, evaluation, and domain contexts) serve to establish relationships between viewpoints and actual views in

order to allow the user to effectively decide on appropriate visualisations based on their particular situation. For more information on the Imago platform consult Vernik (2007).

3. THE USER MODEL

To have a coherent domain context you need the ‘who’, ‘where’ and ‘why’ of a given situation in order to find an appropriate visualisation. The designer of a visualisation implementation model must gather the required information to fulfill these requirements. Imago is one such implementation model. However, Imago is limited in its ability to identify which visualisation is the most effective one to display for a particular viewer. While Imago’s semantic network does contain a user component, it is not implemented at a detailed enough level to capture the capabilities of each individual viewer. Imago (see Vernik, Von Itzstein et al. 2007) implicitly assumes that all viewers have the same perceptual abilities and knowledge about each visualisation. By associating a viewpoint and a visualisation approach together in the model, Imago has bound the same visualisation approach to a viewpoint for every user. This is contrary to the view that no single visualisation is the most effective for all individuals (cf. Golemati, Halatsis et al. 2006).

Our user model overcomes this limitation of Imago’s semantic model by enabling the viewpoints and visualisation approaches to be associated on a viewer-by-viewer basis. This, in turn, enables Imago to provide visualisations that:

1. Are suited to the user’s domain context.
2. Fulfill the user’s specific task requirements.
3. The user can comprehend.

Each visualisation approach has an associated set of declarative and procedural knowledge. Declarative knowledge is knowing basic facts or concepts, while procedural knowledge is knowing how to do something (Ashcraft 1994). An example of declarative knowledge is recognizing that glyph rotation represents the amount of oxygen needed for oxidation in Lie, Kehrler et al.’s (2009) diesel particle filter visualisation. Knowing how to locate the maximum y value in a line graph is an example of procedural knowledge. One’s comprehension of a visualisation is more effortful without this declarative and procedural knowledge.

Each visualisation approach in Imago has certain knowledge that the viewer needs in order to successfully comprehend it. This is illustrated in *Figure 1* by an arrow from the visualisation approaches to the declarative and procedural knowledge schemata. The function of the User Model is to store the schemata that the viewer possesses (see *Figure 1*). The User Model enables Imago to match the schemata that the viewer possesses with the schemata needed to process the visualisation. If the viewer possesses the necessary knowledge to comprehend a visualisation then Imago may use this approach. If the viewer does not possess the necessary knowledge then Imago should avoid using this visualisation approach.

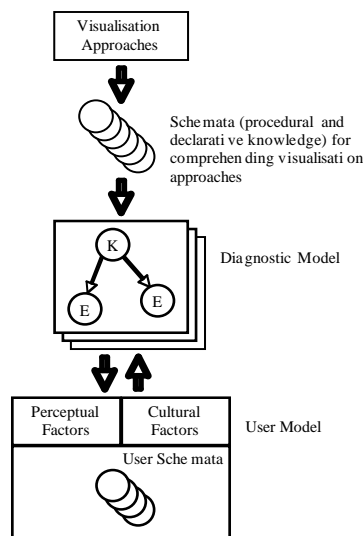


Figure 1. Our proposed user modeling addition to the Imago implementation framework

Perceptual and cultural factors that may influence a viewer's comprehension of a particular visualisation are incorporated in the User Model. For example, if a viewer suffers from colour blindness then any visualisation that relies heavily on the viewer distinguishing between different colours may be ineffective. The use of colour is also culturally dependent (Lakoff 1987; Ware 2000) so a visualisation that uses colour to present information may be ineffective for viewers from cultures that don't share the common meaning for particular colours.

We include the perceptual and cultural factors in the user model and not the diagnostic model as we felt these factors represent capabilities of the user and not of the diagnosis process. However, the Diagnostic Model may need access to these factors when determining if viewer possesses certain knowledge. For this reason we have shown arrows in both directions between the Diagnostic Model and the User Model in Figure.

4. THE DIAGNOSTIC MODEL

The function of the Diagnostic Model is to determine whether a user possesses the declarative and/or procedural knowledge needed to comprehend a visualisation. The accuracy of the User Model is dependent on the accuracy of the Diagnostic Model. The accuracy of the Diagnostic Model is in turn a function of the amount of evidence that can be provided in the model. Fortunately there are a large number of evidentiary events about the viewer which are available to form a probabilistic model, including but not limited to: pupil size, blink rate, skin conductivity, body temperature, blood flow rate and eye movements (Ikehara and Crosby 2005). Whilst some of these may give clues to whether the user possesses the knowledge required to comprehend a visualisation, the Diagnostic model needs several clues in order to get a significant set of evidence to either prove or disprove the existence of this knowledge. Whilst the more evidence the more accurate the diagnostic model we have selected two. Of these two we selected one explicit and one implicit evidence factor. Firstly, we selected self-efficacy beliefs that rely on the user stating either directly or indirectly their belief in their own abilities to perform specific visualisation tasks. Secondly, an implicit evidentiary factor was considered, that of eye movements. We have found support that both measures are suitable for inferring if a viewer possesses a particular knowledge about a visualisation.

Self-efficacy beliefs are an individual's conviction in his or her ability to successfully perform the behaviour necessary to produce a desired outcome (Bandura 1977). Self-efficacy beliefs relate to an individual's performance expectations for a task (Zimmerman 2000), such as: "How confident are you that you can identify low pressure regions in this weather map?" The task-specific focus of self-efficacy expectations distinguishes it from other more general cognitive constructs like self-esteem and self-concept, for individuals may have high self-efficacy expectations about their ability to perform a task even when they hold that particular ability in low esteem (Zimmerman 2000).

A person's self-efficacy beliefs can affect their behaviour and the choices they make (Bandura 1977; Pajares and Miller 1994). Individuals with high self-efficacy expectations are more motivated, persistent and expend more effort as they perform a task. Individuals with higher efficacy expectations are more likely to persevere with a task for longer when they encounter difficulties. This makes self-efficacious individuals more likely to complete a task, and this success in turn reinforces their high efficacy beliefs (Schunk 1991). In contrast, individuals with low self-efficacy expectations choose to avoid situations that they believe exceed their capabilities. They may give up on challenging tasks more easily, and in doing so "will retain their self-debilitating expectations and fears for a long time" (Bandura 1977).

Many studies have reported an association between a person's self-efficacy beliefs and their task performance. In an analysis of 114 studies of self-efficacy from the psychology, business, management and organizational behaviour domains, Stajkovic and Luthans (1998) found a moderately strong correlation between self-efficacy and work related performance. Multon et al (1991) performed an analysis of 36 self-efficacy studies from elementary school, high school, college and non-school settings. They calculated that self-efficacy beliefs accounted for 14% of the variation in students' academic performance. To investigate the relationship between self-efficacy and persistence, Multon et al analyzed 18 studies of students' self-efficacy expectations and academic task persistence. They found that self-efficacy beliefs accounted for 12% of the variation in students' academic persistence levels. In sum, both Stajkovic and Luthans', and Multon et

al's meta-analyses support for claim that an individual's self-efficacy beliefs are related to their task performance in both work and academic settings.

As there have been no studies on self-efficacy related to visualization task performance, we performed a correlation study of self-efficacy beliefs and visualisation task performance. This experiment investigated whether an individual's self-efficacy beliefs predict their performance in a visualisation-based task. In our study we presented 12 volunteers with 8 different visualisation tasks to perform on a computer. The volunteers' ages varied from 30 – 65 years. Some participants reported that they had prior experience with using the types of visualisations in this study while others reported no such prior experience. The visualisations and text were rendered on a 17-inch computer screen. None of the participants reported any difficulty reading the text and graphics on the screen, or with using our test program.

We asked each participant for his or her self-efficacy beliefs about a particular visualisation task prior to them performing each task. We developed a computer program to:

1. Record each participant's self-efficacy beliefs for a particular visualisation-based task.
2. Present the participant with a visualisation and the description of the task they had to perform.
3. Record the participants' response to each task.

Participants selected an option from a list of possible answers for each task. We did not provide a 'not sure' option, so the participants were forced to select an answer. We did provide participants with the opportunity to indicate if they guessed the answer immediately following each task. Our program recorded participant responses electronically for analysis.

Each participant provided 2 self-efficacy ratings per visualisation task. We asked participants to rate their level of agreement or disagreement with task-specific statements like: *'I am able to determine the company whose stock's closing price is less than its opening price'*. Participants responded using an 8-item Likert scale with levels ranging from 'very strong agree' to 'very strongly disagree'. An even number scale was chosen to discourage participants from simply selecting the middle level. We phrased each pair of questions so that one was the logical negation of the other. An acquiescent participant will respond in a positive manner to each question, giving rise to contradictory self-efficacy ratings. Overall, less than one third of the self-efficacy ratings were contradictory, which we removed from our experimental analysis.

Our study employed several different types of visualisations, including: 2-dimensional line graphs, pie graphs, scatter plots, Marey's train schedule in Tufte (1983), a novel 3-dimensional stock visualisation, texture maps, symbol maps, network diagrams and Sawant et al.'s (2007) glyph weather map. Participants were familiar with some of these visualisations, for example line graphs and pie charts, but were unfamiliar with several others, such as the novel 3-dimensional stock visualisation or Marey's visual train schedule. Each participant had to perform several different tasks using both familiar and unfamiliar visualisations. We randomized the order in which we presented the tasks between participants to control for possible temporal effects.

Our study examined the hypothesis that an individual's self-efficacy beliefs about a visualisation-based task influence their success in that task. If higher self-efficacy results in greater task motivation and perseverance, then it is likely that an individual with higher self-efficacy will experience greater success in our visualisation tasks than someone with lower self-efficacy. We predicted participants with higher task self-efficacy would successfully complete the visualisation-based task, while participants with lower task self-efficacy would be unsuccessful in completing the task.

Our experimental results provide support for our hypothesis that self-efficacy beliefs are related to task performance. We found that for 9 of the 12 participants, their self-efficacy beliefs correctly predicted their task performance in more than 50% of the tasks. For 2 out of the 12 participants, their self-efficacy beliefs correctly predicted their task performance in less than 50% of the tasks. One participant's self-efficacy beliefs correctly predicted their task performance in exactly 50% of the tasks. We defined a correct self-efficacy rating as a rating that predicts future task performance: if an individual's task self-efficacy is low and their performance is poor, or if their self-efficacy is high and their task performance is also high, then we labeled their self-efficacy rating as correct. We defined an individual's self-efficacy rating as incorrect if their self-efficacy rating does not predict their task performance. A one-tailed binomial sign test of the null hypothesis that self-efficacy is unrelated to visualisation-task performance found that the probability of obtaining our results is less than 5% ($p = 0.033$). When averaged across all 12 participants, self-efficacy beliefs correctly predicted task performance in 73% of the visualisation tasks. Our results provide sufficient evidence to reject the null hypothesis and accept the alternative hypothesis that self-efficacy is related to visualisation-task performance.

Our experimental results also suggest that low self-efficacy beliefs are more accurate predictors of task performance than high self-efficacy beliefs. We found that for the 8 participants that reported low task self-efficacy ratings, their self-efficacy beliefs correctly predicted their task performance in more than 50% of the tasks. A sign test using the null hypothesis that self-efficacy is unrelated to visualisation-task performance found that the probability of obtaining our results is less than 1% ($p = 0.0039$). We found insufficient evidence to reject the null hypothesis when we conducted a sign test of the predictive validity of high self-efficacy beliefs ($p = 0.25$). These results suggest that participants are accurate in their predictions of poor task performance, but they can over-estimate their task performance in situations where they expect to succeed.

In summary, we found that an individual's low self-efficacy beliefs are associated with his or her visualisation task performance. However, we did not find any evidence to suggest that an individual's high self-efficacy beliefs are as strongly related to their overall task performance. Our findings suggest that in some – but not all – situations a person's self-efficacy beliefs are a valid and potentially useful predictor of their performance in a visualisation-based task.

4.1 Eye Movements

Recent research suggests that eye movements may provide a way to infer aspects of a viewer's processing of a visualisation. Grant and Spivey (2003) report that an individual's locus of visual attention is linked to their cognitive processing during a spatial reasoning task. Grant and Spivey recorded the eye movements of 14 subjects attempting to solve Dunker's radiation problem. They found that the participants who successfully solved Dunker's radiation problem made more eye movements that mirrored the solution to the problem. Participants who were unsuccessful in solving Dunker's radiation problem did not make as many of these eye movements. Grant and Spivey interpreted their findings as evidence that the participant's eye movements reflect their cognitive processing. Thomas and Lleras (2006) show that there is a close link between an individual's eye movements and their cognitive processing. In a study involving 99 subjects, Thomas and Lleras manipulated the scan patterns (sequence of eye fixation locations) of the participants as they attempted to solve Dunker's radiation problem. Each participant was assigned a particular eye scanning pattern that they were required to reproduce for a 4 second period every 30 seconds. Thomas and Lleras found that participants were most successful when their eye movements replicated the solution to the radiation problem. Thomas and Lleras argue their results are evidence that not only can a person's cognitive processing influence their eye movements, but a person's eye movements can also influence their cognitive processing.

A viewer's scan patterns may also reflect their cognitive processing of a visualisation. One could deduce whether a viewer was trained to 'read' a visualisation on the basis of their eye movements as they view the visualisation. To investigate this claim further, we undertook an eye tracking study using the novel botanical tree visualisations from Kleiberg, E., H. van de Wetering, et al. (2001). The remainder of this section details our experiment, and the diagnostic model that we have developed.

The tree visualisation in Kleiberg, E., H. van de Wetering, et al. (2001) represents the files and folders on a computer's hard drive using a botanically inspired tree structure (see Figure 1). Individual files in a folder are represented as 3-dimensional cones on a sphere and sub-folders are represented as branches. The main trunk of the tree represents the root directory on a hard drive. This visualisation was ideally suited to our study, as participants required knowledge that is not easily deduced in order to successfully comprehend it. For example, it is not obvious that the height of each 3-dimensional cone reflects the relative size of that file in a folder. However, this declarative knowledge would be crucial to perform the task: "identify the file that is the greatest percentage of its folder size".

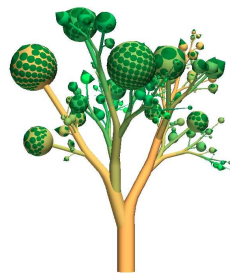


Figure 1. A sample Botanical tree visualisation showing the files and folder on a computer's hard drive.

Our research hypothesis for the eye tracking study was trained participants would exhibit significantly different scan patterns to untrained participants when viewing the tree visualisation. We hypothesized that differences in scan patterns results from the trained participants' greater knowledge about how to 'read' the botanical tree visualisation. We predicted greater similarity between parts of the scan paths of trained participants than untrained participants.

To investigate our research hypothesis, we recorded the eye movements of participants as they performed a series of tasks using the botanical tree visualisation. A total of 24 participants of varying ages and educational backgrounds volunteered for our study. We randomly allocated each participant to either the 'trained' or 'untrained' groups. Each participant undertook a pre-test on their level of knowledge of files and folders on a computer. Participants who had little or no knowledge of files and folders were excluded from the study.

We provided trained participants with detailed instruction on how to read the botanical tree visualisation. We tested each trained participant on his or her knowledge of the botanical tree visualisation prior to running the eye tracking experiment. We trained participants using different trees to the ones we used in the test. Any trained participant who responded incorrectly to one or more questions about the botanical tree visualisation had to repeat the training process before starting the eye tracking experiment.

Untrained participants viewed examples of the botanical tree visualisation prior to the eye tracking experiment, but we did not provide them with any explicit training on how to read the visualisation. Untrained participants were instructed to formulate their own mental model for what the cones, branches and spheres in the botanical visualisation represent and to use this model to guide their solution to the task.

The eye tracking experiment consisted of a series of tasks involving the botanical tree visualisation. These tasks included:

1. Find the directory that contains the greatest number of files in it.
2. Find the file that is the greatest percentage of its folder's size.
3. Determine the level of the highlighted folder in the hierarchy.

We recorded the participants' eye movements using a Tobii™ T120 eye tracker and Tobii™ Studio software package. Text explaining the task was displayed on the T120 screen prior to the presentation of the botanical tree visualisation. Participants initiated the test when they understood the task, and ended each task either by: clicking the mouse pointer on the region of the tree that contained the answer, or by providing a verbal response. None of the tasks were timed. We removed from the analysis set any recordings that contained less than 50% of the total number of fixation samples.

We analyzed each participant's eye movements in Matlab® by converting their eye fixation locations to a sequence of integer values, where each integer value, $k > 0$, uniquely identifies a predefined region of interest in the tree. We defined regions of interest (ROIs) around the entities in the tree that were relevant to the task (see Figure 2). For example, if the task was to find the directory containing the greatest number of files then we constructed a ROI around each of the spheres in the visualisation that contained more than one file. Any fixation that landed on or near a pre-defined ROI was mapped to the unique integer value for that ROI. Any fixation that did not land on or near a ROI was assigned the value 0.

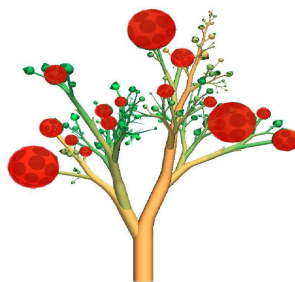


Figure 2. The ROI (shown in red) for a tree visualisation.

We used the motif discovery algorithm in Xia (2007) to search for distinct sequences, known as motifs, that are present in trained participants' scan paths but not in untrained participants' scan paths. Although the motif discovery algorithm was developed to identify regulatory motifs in DNA and functional motifs in proteins (Xia 2007), the algorithm is easily modified to find ROI sequences in two or more scan paths. The

output of the motif discovery algorithm is a Position Weighted Matrix (PWM) that records the frequency and position of each ROI within the sequences. It is possible to use the PWM to calculate the likelihood that a ROI sequence from a trained participant's scan path is present in an untrained participants' scan path. Space limitations do not permit a full discussion of the motif discovery algorithm so we refer the reader to Xia (2007) for further details.

We developed a set of 7 binary classifiers that measure different features of a participant's scan paths. For example, the 'target fixation' classifier counts the number of sequences of a given length that contain the id of the target ROI. If the number of sequences containing the target id exceeds a threshold value then the classifier designates the viewer as trained. The 'max odds' classifier uses the PWM that is output from the motif discovery algorithm to calculate the ROI sequence of a given length that is most likely to occur in the scan paths of trained participants and not untrained participants. If the likelihood value exceeds some threshold then the classifier designates the participant as trained. If no likelihood value exceeds the threshold then the classifier classifies the participant as untrained. Table 1 below summarizes the 7 different binary classifiers.

Table 1. The binary classifiers used to analyze the participant scan paths

<i>Classifier</i>	<i>Description</i>
Max odds	Uses the motif with the maximum likelihood that it is unique to trained participants to classify viewers
Max odds above threshold	Uses the motif with the maximum likelihood above a given threshold to classify viewers
Proportion odds above threshold	Uses the proportion of motifs likelihoods above a given likelihood threshold to classify viewers
Max odds including target	Uses the maximum likelihood of the motifs containing the target ROI to classify viewers
Proportion motifs including target	Uses the proportion of motifs containing the target ROI to classify viewers
Proportion motifs containing task relevant ROI	Uses the proportion of motifs that contain ROI that are highly relevant to the task to classify viewers
Proportion conspicuous motifs	Uses the proportion of motifs that contain highly conspicuous ROI (for example, the largest spheres) to classify viewers

The threshold values for classifying a participant as trained or untrained were determined during an initial training period. We employed the cross-validation method (Kuncheva 2004) to partition our experimental data into distinct training, validations and test sets. We constructed Receiver Operator Characteristics (ROC) curves to visually represent the accuracy of each classifier for different threshold values. By assuming equal costs for incorrectly classifying a trained participant as untrained (a false negative classification) and incorrectly classifying an untrained participant as trained (a false positive classification), the optimal threshold for a classifier was the most 'north-west' value in its ROC curve (see Figure 3 below).

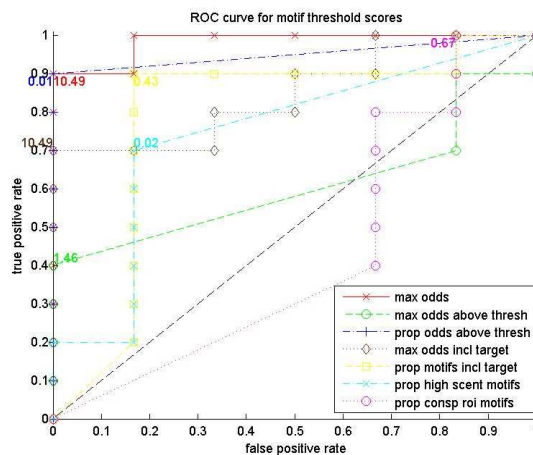


Figure 3. The Receiver Operator Characteristic (ROC) curve (see Fawcett 2006) for 7 binary classifiers during the training period. The most 'north-west' point for each curve represents the optimal threshold value for that classifier.

We combined up to 3 binary classifiers at a time to form a Naïve Bayes classifier network (see Friedman, Geiger et al. (1997) for more information on Naïve Bayes networks). We found that increasing the number of binary classifiers in the Naïve Bayes network beyond 3 did not significantly improve the network's accuracy but it did significantly increase the training time. As a result we decided to limit the maximum number of classifiers in the network to 3.

We used the hybrid simulating annealing algorithm from Brooks and Morgan (1995) to find committees of Naïve Bayes networks that accurately classify the validation sets. The hybrid annealing algorithm records potential optimal solutions as the system is 'cooled' to a lower energy state. One may then optimize these potential solutions further using an alternative optimization algorithm, such as steepest descent, once the system has cooled. Using this approach, we regularly found several committees that were equally good at classifying participant scan paths.

A classifier's true positive and false positive rates are defined as the proportion of trained participants that it classifies as trained, and the proportion of untrained participants that it classifies as trained, respectively. So to verify the accuracy of a committee we calculated its true positive rates and false positive rates for the unseen test sets. We calculated 95% Confidence Intervals for the true positive and false positive rates of the optimal Naïve Bayes networks. The intervals for the first 2 tasks are shown in **Erro! A origem da referência não foi encontrada..** As these confidence intervals attest, our committees of Naïve Bayes networks are able to correctly classify trained and untrained viewers with an accuracy that is significantly above chance levels.

Table 2. The true positive and false positive rates for the optimal classifier for the visualisation tasks: (1) find the directory that contains the greatest number of files in it; (2) find the file that is the greatest percentage of its folder's size.

	95% CI for true pos rate	95% CI for false pos rate
Task 1	(0.70,0.81)	(0.13,0.21)
Task 2	(0.75,0.86)	(0.02,0.10)

In the third task, we could not find a classifier that could accurately distinguish between trained and untrained participants. Both groups fixated heavily on the branches leading to the highlighted folder (as illustrated by the heat maps for trained and untrained participants in Figure 4). It is likely that both groups of participants recognized that the branches leading to the highlighted folder were related to its level in the hierarchy. Most trained participants correctly stated that the highlighted folder was at level 3 in the hierarchy. However, none of the untrained participants could correctly state its level. This task highlights a limitation of eye movements as a diagnostic tool: one cannot always deduce what an individual is thinking from their eye movements.

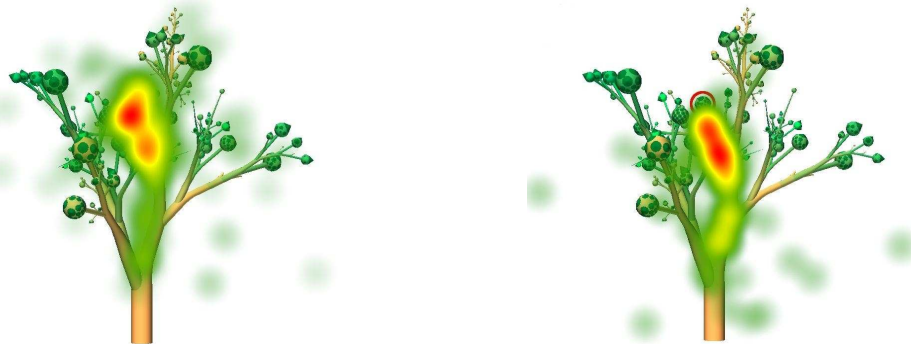


Figure 4. The heat maps of trained (left) and untrained (right) participants when performing the task: determine the level in the hierarchy of the highlighted folder. The red regions represent the areas receiving most fixations while the green regions represent areas receiving fewer fixations. Both trained and untrained viewers fixate heavily on the branches leading to the highlighted folder.

There are limits to what one can reliably conclude about a person's cognitive processing from their eye movements. Viewers can perform some cognitive tasks, such as word or object recognition, as their gaze is shifting to another location (Irwin 2004). So the amount of time that a person fixates on an object or region in a visual scene may not reflect how long they spend processing the fixated object or region. Fixation locations indicate the regions in a scene a person has attended to, but they do not indicate what that person was

thinking about each region (Hegarty 2006). Although previous eye movements studies have shown that it is possible for viewers to dissociate their gaze from their locus of cognitive processing (Yantis 1998; Irwin 2004), some researchers contend that a viewer's attention and gaze are tightly coupled for "all but the simplest displays" (Deubel and Schneider 1996; Rehder and Hoffman 2005). So while a person's scan paths, or sequence of fixation locations, are individualistic (Henderson 2003) and highly influenced by the viewer's current task (Lipps and Pelz 2006), our findings and those reported in Underwood, Humphrey et al. (2008) suggest that it is possible to assess an individual's knowledge about a visualisation on the basis of their eye movements. In summary, even though eye movements cannot divulge what a person is thinking, they can provide useful clues whether a viewer is trained for a visualisation.

Our decision to use self-efficacy beliefs and eye movements is based on pragmatic and theoretical considerations. Several previous studies have reported a correlation between an individual's self-efficacy beliefs and their task performance in different contexts. Likewise, there is growing evidence that one's eye movements are inextricably linked to his/her cognitive processing. Eye tracking is relatively unobtrusive and provides abundant real-time data about how the viewer is processing a display. As our eye tracking user study demonstrated, it is possible to distinguish between viewers that are trained to read a visualisation and those that are not.

5. CONCLUSIONS

In this paper we have argued that a visualisation is effective when it suits the data type, it matches the task the viewer is performing and the viewer has the perceptual abilities and knowledge required to comprehend it. This is the deficiency that we have identified with the Imago visualisation environment. To overcome this deficiency, we have proposed a high-level User Model that gives Imago the capability to provide the viewer with visualisations that they can comprehend. We developed and tested a Diagnostic Model that uses a multiple Naïve Bayes classifiers to determine if a viewer has certain knowledge about a visualisation. We presented two evidentiary factors for the Diagnostic Model, which we investigated in two User Studies. We found that a viewer's low self-efficacy beliefs are related to their performance in a visualisation task but we did not find sufficient evidence to draw the same conclusion for high self-efficacy beliefs. Our eye tracking study demonstrated that it is plausible to distinguish between trained and untrained viewers using their scan patterns. We described how we could distinguish between trained and untrained viewers using binary classifiers combined in multiple autonomous Naïve Bayes classifiers. In summary, our User and Diagnostic Models are a step closer towards a method of providing people with semi autonomous selection of effective visualisations for comprehending the ever-increasing amounts of information at their disposal.

6. FUTURE WORK

We are currently investigating how to most efficiently and effectively combine multiple classifiers to enhance the accuracy of our Diagnostic Model. We are also investigating what tasks and visualisations are suited to using eye tracking to distinguish between trained and untrained participants. This will enable us to predict which visualisations and tasks we can successfully apply our Diagnostic Model to. In the future we will investigate whether other physiological indicators such as skin conductivity or blood flow rate, are useful in measuring one's performance in a visualisation task.

ACKNOWLEDGEMENTS

We gratefully acknowledge the support of Dr Huub Van de Wetering from Technische Universiteit, Eindhoven for providing us with a copy of the Botanical Tree visualisation software, and Bas Tjdhof from Tobii Technology for providing us with access to the Tobii™ eye tracker.

REFERENCES

- Allen, G. L., C. R. Miller Cowan, et al. (2006). "Acquiring information from simple weather maps: Influences of domain-specific knowledge and general visual-spatial abilities." *Learning and Individual Differences* 16(4): 337-349.
- Ashcraft, M. H. (1994). *Human Memory and Cognition*. New York, NY USA, HarperCollins College Publishers.
- Bandura, A. (1977). "Self-efficacy: Toward a unifying theory of behavioral change." *Psychological Review* 84(2): 191-215.
- Brooks, S. P. and B. J. T. Morgan (1995). "Optimization Using Simulated Annealing." *Journal of the Royal Statistical Society. Series D (The Statistician)* 44(2): 241-257.
- Casner, S. M. (1991). "Task-analytic approach to the automated design of graphic presentations." *ACM Transactions on Graphics* 10(2): 111-151.
- Cleveland, W. S. (1985). *The elements of graphing data*. Monterey, CA, Wadsworth Advanced Books and Software.
- Deubel, H. and W. X. Schneider (1996). "Saccade target selection and object recognition: Evidence for a common attentional mechanism." *Vision Research* 36(12): 1827-1837.
- Domik, G. O. and B. Gutkauf (1994). *User Modeling for Adaptive Visualization Systems*. IEEE Conference on Visualization 1994. Washington, DC, USA.
- Fawcett, T. (2006). "An introduction to ROC analysis." *Pattern Recognition Letters* 27(8): 861-874.
- Freedman, E. and P. Shah (2002). *Toward a Model of Knowledge-Based Graph Comprehension*. *Diagrammatic Representation and Inference*: 59-141.
- Friedman, N., D. Geiger, et al. (1997). "Bayesian Network Classifiers." *Machine Learning* 29(2): 131-163.
- Golemati, M., C. Halatsis, et al. (2006). *A Context-Based Adaptive Visualization Environment*. *Information Visualization, 2006. IV 2006. Tenth International Conference on, IEEE*: 62-67.
- Gouin, D., P. Evdokiou, et al. (2002). *A Showcase of Visualization Approaches for Military Decision Makers*. NATO's Research & Technology Organisation (RTO) Information Systems Technology panel Workshop: Massive Military Data Fusion and Visualisation: Users Talk with Developers. Halden, Norway.
- Grant, E. R. and M. J. Spivey (2003). "Eye movements and problem solving." *Psychological Science* 14(5): 462-466.
- Hegarty, M. (2004). *Diagrams in the Mind and in the World: Relations between Internal and External Visualizations*. *Diagrammatic Representation and Inference*: 1-13.
- Hegarty, M. (2006). *Eye Fixations and Diagrammatic Reasoning*. *Diagrammatic Representation and Inference*: 13-15.
- Henderson, J. M. (2003). "Human gaze control during real-world scene perception." *Trends in Cognitive Sciences* 7(11): 498-504.
- Ikehara, C. S. and M. E. Crosby (2005). *Assessing Cognitive Load with Physiological Sensors*. *System Sciences, 2005. HICSS '05. Proceedings of the 38th Annual Hawaii International Conference on*.
- Irwin, D. E. (2004). *Fixation location and fixation duration as indices of cognitive processing*. *The Interface of Language, Vision, and Action: Eye Movements and the Visual World*. J. M. Henderson and F. Ferreira. New York, Psychology Press: 105-134.
- Kleiberg, E., H. van de Wetering, et al. (2001). *Botanical visualization of huge hierarchies*. *Information Visualization, 2001. INFOVIS 2001. IEEE Symposium on*.
- Kosslyn, S. M. (1989). "Understanding Charts and Graphs." *Applied Cognitive Psychology* 3(3): 185-226.
- Kuncheva, L. I. (2004). *Combining pattern classifiers : methods and algorithms*. Hoboken, NJ, J. Wiley.
- Lakoff, G. (1987). *Women, fire, and dangerous things : what categories reveal about the mind*. Chicago, University of Chicago Press.
- Lie, A. E., J. Kehler, et al. (2009). *Critical Design and Realization Aspects of Glyph-based 3D Data Visualization*. *proceedings of the Spring Conference on Computer Graphics (SCCG 2009)*.
- Lipps, M. A. and J. B. Pelz (2006). *Influence of high-level cognitive goals on gaze patterns: Yarbus revisited*. *ACM SIGGRAPH 2006 Research posters*. Boston, Massachusetts, ACM.
- Lohse, G. L. (1997). "The role of working memory on graphical information processing." *Behaviour & Information Technology* 16(6): 297-308.
- Lowe, R. K. (1993). "Constructing a mental representation from an abstract technical diagram." *Learning and Instruction* 3(3): 157-179.
- Lowe, R. K. (1994). "Selectivity in Diagrams: reading beyond the lines." *Educational Psychology* 14(4): 467 - 491.
- Mackinlay, J. (1986). "Automating the design of graphical presentations of relational information." *ACM Transactions on Graphics* 5: 110-141.
- Multon, K. D., S. D. Brown, et al. (1991). "Relation of self-efficacy beliefs to academic outcomes: A meta-analytic investigation." *Journal of Counseling Psychology* 38(1): 30-38.

- Neisser, U. (1976). *Cognition and Reality: Principles and Implications of Cognitive Psychology*. San Francisco, CA, USA, W. H. Freeman and Company.
- Nowell, L., R. Schulman, et al. (2002). Graphical encoding for information visualization: an empirical study. *Information Visualization, 2002. INFOVIS 2002. IEEE Symposium on*.
- Pajares, F. and M. D. Miller (1994). "Role of self-efficacy and self-concept beliefs in mathematical problem solving: A path analysis." *Journal of Educational Psychology* 86(2): 193-203.
- Petre, M. and T. R. G. Green (1993). "Learning to Read Graphics: Some Evidence that 'Seeing' an Information Display is an Acquired Skill." *Journal of Visual Languages & Computing* 4(1): 55-70.
- Rehder, B. and A. B. Hoffman (2005). "Eyetracking and selective attention in category learning." *Cognitive Psychology* 51(1): 1-41.
- Sawant, A. P. and G. H. Christopher (2007). "Need for perceptual display hierarchies in visualization." *Crossroads* 13(3): 6-6.
- Scaife, M. and Y. Rogers (1996). "External Cognition: how to graphical representations work?" *International Journal of Human Computer Studies* 45(2): 185-213.
- Schunk, D. H. (1991). "Self-Efficacy and Academic Motivation." *Educational Psychologist* 26(3/4): 207.
- Stajkovic, A. D. and F. Luthans (1998). "Self-efficacy and work-related performance: A meta-analysis." *Psychological Bulletin* 124(2): 240-261.
- Thomas, L. E. and A. Lleras (2006). "Moving eyes and moving thought: The spatial compatibility between eye movements and cognition." *Journal of Vision* 6(6): 871-871. Trafton, J., S. Trickett, et al. (2005). "Connecting Internal and External Representations: Spatial Transformations of Scientific Visualizations." *Foundations of Science* 10(1): 89-106.
- Tufte, E. R. (1983). *The Visual Display of Quantitative Information*, Graphics Press.
- Tversky, B., M. Agrawala, et al. (2007). *Cognitive Design Principles for Automated Generation of Visualizations. Applied spatial cognition: from research to cognitive technology*. G. L. Allen. Mahwah, N.J, Lawrence Erlbaum Associates: 53-74.
- Underwood, G., K. Humphrey, et al. (2008). *Knowledge-Based Patterns of Remembering: Eye Movement Scanpaths Reflect Domain Experience. HCI and Usability for Education and Work*: 125-144.
- Vernik, R., G. S. Von Itzstein, et al. (2007). *Imago: An integrated prototyping, evaluation and transitioning environment for information visualisation. 11th International Conference Information Visualization (IV '07)*. Zurich, Switzerland: 17-22.
- Ware, C. (2000). *Information visualization: perception for design*. San Diego CA, Academic Press.
- Wattenberg, M. and D. Fisher (2004). "Analyzing perceptual organization in information graphics." *Information Visualization* 3: 123-133.
- Xia, X. (2007). *Bioinformatics and the cell : modern computational approaches in genomics, proteomics, and transcriptomics*. New York, Springer: xv, 349 p.
- Yantis, S. (1998). *Control of Visual Attention*. Attention. H. Pashler. Easst Sussess, UK, Psychology Press: 223-256.
- Zimmerman, B. J. (2000). "Self-Efficacy: An Essential Motive to Learn." *Contemporary Educational Psychology* 25(1): 82-91.

MOVEMENT EPENTHESIS WITH PROSODY CONTROL FOR CHINESE SIGN LANGUAGE VIDEO SYNTHESIS

Ru Wang¹, Lichun Wang¹, Dehui Kong¹, Baocai Yin¹ and Qingming Huang²

¹Beijing Key Laboratory of Multimedia and Intelligent Software Technology

College of Computer Science and Technology, Beijing University of Technology, Beijing 100124, China

²Graduate University of Chinese Academy of Sciences

ABSTRACT

To ensure smoothness of synthesized sign language sentence in video format, which is concatenated with sign language video clips, this paper proposes a kind of movement epenthesis approach based on prosody controlling. The proposed method decides number of transition frames according to sign's velocity of pre-video clip and post -video clip, computes appropriate locations on the motion trajectories, synchronizes behavior of different models while the transition units are integrated. Experiment results show that the synthesized sign language sentence is smooth and natural as expected.

KEYWORDS

Video synthesis; Chinese Sign Language; transition; movement epenthesis; prosody

1. INTRODUCTION

For hearing-disabled people, sign language is their main tools for communicating. However, most information is expressed with natural language in our society, which is inconvenient for hearing-disabled people to understand and engage in social affairs. Sign language synthesis provides a way to generate sign language expression based on natural language. Synthesized sign language has two formats, 3D animation and video, the latter is more realistic since it uses videos acted by real person.

The process of sign-language video synthesis is concatenating video clips of real signer's demonstration into new continuous sign-language sentences. Connecting two sign-language video clips directly would result vision hop, even audience's misunderstanding in some case. A possible method for the problem is adding some transition frames to make synthesis effect smooth and natural, and the inter-sign transition periods are called movement epenthesis^[1]. Selecting what and how many clips is main problem for sign video's movement epenthesis to make synthesized video smooth, real and consistent with contexts, for which this paper contributes a method based on sign language prosody controlling.

American linguist Eileen Forestal mentioned that prosody is the feature collections that generate rhythm, stress and feel. For sign language, prosody means non-manual action and speed, rhythm, pause of gestures. According to the general concept of prosody, this paper studies prosody feature of Chinese sign language in two ways: velocity of signing which is mapped on number and location of interpolation frames; feel of signing which is mapped on multi-model behavior synchronizing for combined interpolated frames.

2. RELATED WORK

In order to synthesize natural sign language with prosody, the first thing is describing and marking prosody of sign language. The institute of linguistics in Boston University summed up a series of prosodic features in American Sign Language (ASL), and developed software "SignStream" to assist sign language video analysis and annotation^[2]. Brenda Nicodemus^[3] in Boston University researched the roles of non-gesture postures in prosody boundary dividing, summarized a series of prosodic features, and has done qualitative

analysis through video analysis and manual calibration. CSLML proposed by Kejia YE etc. [4] provides a method for prosody description of CHINESE sign language.

Based on marked prosody features of sign language, corresponding sign language animation could be synthesized. Such as eSign^[5] driven by their markup language SiGML, Chinese Sign Language Expressive Animated Broadcaster(CSLEAB) system^[4] driven by CSLML. Besides, paper^[6] designs a multi-model behavior synchronizing prosody model for Chinese sign language synthesis to control virtual human's behavior synchronously. Until currently, sign language prosody information has not been used for sign language video synthesis.

Slavko and Krapez^[7] synthesized Slovene Sign based on video clip database, their synthesized results are not discontinuous because lacking of transition frames. By inserting transition frames, Wei-Sheng Chen etc. improved vision effect of synthesized Chinese Taipei Sign video^[8], but their interpolation just consider average speed of re-joined video clips to access transitions, which ignore rhythm of sign language.

3. TRANSITION INTERPOLATION WITH PROSODY CONTROL

The process of sign-language movement epenthesis includes three steps: selecting the most appropriate joining video clips and “joining points”, which are deduced from location of key joints; computing motion trajectories of key joints; judging interpolated locations on motion trajectories. This paper considers sign language prosody for the third step.

3.1 Best “Joining Points”

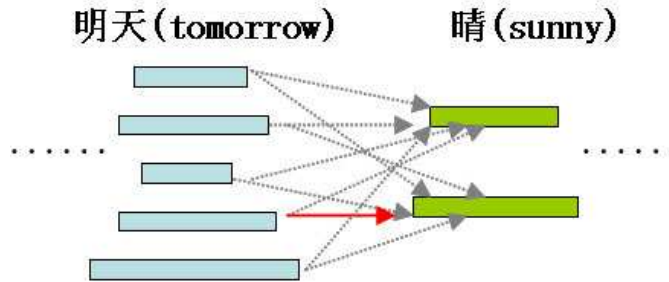


Figure 1. Selecting the most appropriate joining video clips of Sign-language words

Considering that there may be more candidate sign-language video clips for one text word, as shown in Fig.1, word “tomorrow” has 5 corresponding video clips cut from different sign language sentences. Wei-Sheng C^[8] predicted optimal video clips and best joining points through minimizing the cumulative joining cost among words, which is expressed by formula (1):

$$AccCost_j^i = \min_{k \in \mathcal{O}(i-1)} (AccCost_k^{i-1} + MinCost_{i,j}^k) \quad (1)$$

$$MinCost_{i,j}^k = \min_{(x \in A(i-1,k), y \in A(i,j))} (Cost(x, y)) \quad (2)$$

Among them,

$AccCost_j^i$: The cumulative cost of the j^{th} video clip for the i^{th} word;

$MinCost_{i,j}^k$: The minimal cost of connecting the k^{th} candidate video clip of the $(i-1)^{\text{th}}$ word to the j^{th} candidate video clip of the i^{th} word;

$\mathcal{O}(i)$: The number of candidate video clips of the i^{th} word;

$Cost(x,y)$: The cost of connecting key joint locations x and y in given video clips;

$A(i,j)$: The set of key joint locations that can be selected as joining points in the j^{th} candidate video clip of the i^{th} word;

$Cost(x,y)$ is computed based on total consideration of angle transfers and distance between key joints in the motion trajectories.

$$cost(x, y) = S(x, y) + D(x, y) \tag{3}$$

$$S(x, y) = S(u_0, u_{m(x,y)+1}) = \frac{1}{m(x, y)} \sum_{r=1}^{m(x,y)} \cos^{-1} \left(\frac{\langle \overline{u_{r-1}u_r}, \overline{u_r u_{r+1}} \rangle}{\| \overline{u_{r-1}u_r} \| \times \| \overline{u_r u_{r+1}} \|} \right) \tag{4}$$

Among them, $S(x,y)$ and $D(x,y)$ respectively represent angle transfers and distance between key joints. $D(x,y)$ is Euclidean distance between key joint locations x and y in the video clips to be connected; u_r is the r^{th} interpolated location, and $m(x,y)$ is amount of interpolated locations between x and y , $1 \leq r \leq m(x,y)$; $\| \overline{u_a u_b} \|$ is length of vector $\overline{u_a u_b}$, and $\langle \overline{u_a u_b}, \overline{u_c u_d} \rangle$ is the inner product of vector $\overline{u_a u_b}$ and $\overline{u_c u_d}$.

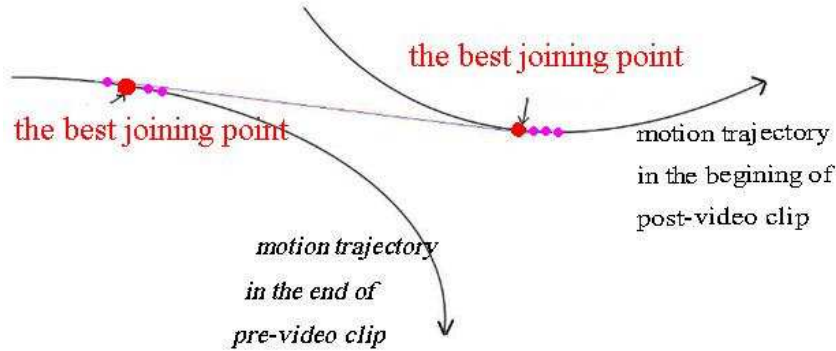


Figure 2 The best joining points for selected sign-language video clips

Through the above method, the best joining points for connecting any two sign-language words can be calculated out, as shown in Fig.2.

3.2 Motion Trajectory

Motion trajectories of key joints are fitted based on best joining points of re-joined sign language video clips. Based on the key joint locations in each frame of re-joined video clip, the corresponding motion trajectories are drawn through the following formula (5), in which P_i is control points of the motion curves.

$$C(t) = \sum_{i=0}^n P_i B_{i,n}(t), \quad B_{i,n}(t) = \binom{n}{i} t^i (1-t)^{n-i} \tag{5}$$

Having the motion trajectories of key joints, the number of needed interpolated frames could be calculated with formula (6), where $m(x,y)$ is number of interpolated frames in motion trajectories, v_x and v_y are key joint's motion velocities in pre and post-video clips, which can be gotten through video analysis.

$$m(x, y) = \frac{curve_lengths}{(v_x + v_y) / 2} \tag{6}$$

3.3 Interpolation with Prosody Control

For sign-language video synthesis, one manifestation for prosody control is adjusting interpolation locations on the motion trajectories according to changing rhythm and pauses. Due to different emotion of sign language expression, rhythm and dynamics of gestures in re-joined sign-language video clips are different. While joining them, some intermediate gestures are interpolated as transition frames to form smooth gesture transition. This paper improves the interpolation formula in paper^[8] through considering sign language prosody.

$$u_r = \sum_{i=0}^3 P_i B_{i,3} \left(\frac{r}{m(x,y)} \right) = \sum_{i=0}^3 P_i \binom{n}{i} \left(\frac{r}{m(x,y)} \right)^i \left(1 - \left(\frac{r}{m(x,y)} \right) \right)^{3-i} \quad (7)$$

$$m'(x,y) = \frac{curve_lengths}{v_x \times \frac{m(x,y)-r}{m(x,y)} + v_y \times \frac{r}{m(x,y)}} \quad (8)$$

Formula (7) computes interpolated transition locations, which considers prosody information of re-joined sign language videos. In this paper, four control points are chosen for transition trajectory fitting, two in the fore-video clip and two in the post-video clip, so value of n in formula (7) is 3. u_r is interpolated locations in transition trajectory; “*curve_lengths*” is length of transition curves; v_x and v_y are average velocities of key joints in two re-joined videos, gotten by parsing the sign language video information; “ $m(x,y) = \frac{curve_lengths}{(v_x+v_y)/2}$ ”

is the number of interpolation calculated with uniform interpolation, $m'(x,y)$ is dynamically calculated interpolated number, and $0 < r < m(x,y)$.

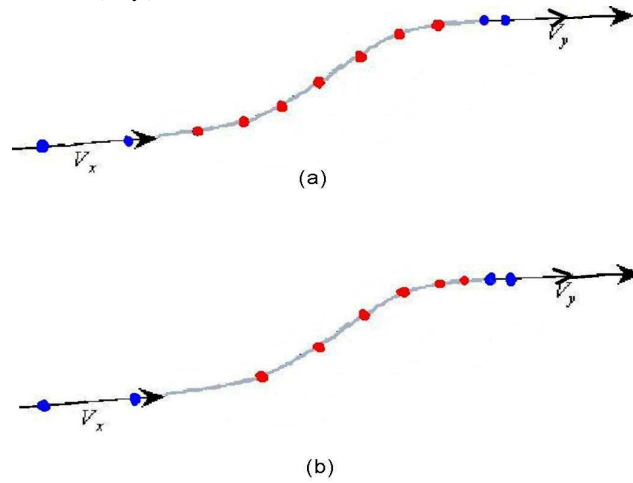


Figure 3. Location interpolation for transition frames with prosody control

In Fig.3, the four blue points are locations in re-joined frames and the red points are their interpolated locations. Trajectory (a) is uniform interpolation for transition frames, and trajectory (b) is interpolation with prosody control calculated by formula (7), where velocity conversion is a gradual process. Taking into account motion and rhythm information in the re-joined sign language videos, the synthesized transition motion is closer to the real movement.

4. INTERPOLATED FRAME SYNTHESIS

4.1 Multi-model Behavior Synchronizing

Multi-granularity transition units, including images of head, arms (left and right), hands (left and right) and torso, are re-assembled to form new interpolated frames of sign-language videos. Since numbers of interpolation calculated for different key joints are different, number of image unit for different components are not same, how to synchronize behavior in image unit for different component is important for final synthesized sign language video. This paper takes the maximum interpolation number of all key joints as final number of interpolation to make the interpolated frame have synchronized behavior.

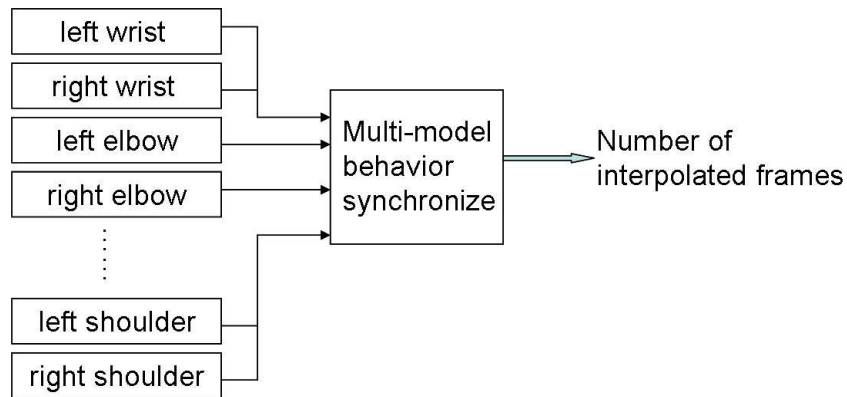


Figure 4. Multi-model behavior synchronizing of sign language motion

4.2 Synthesis of Transition Frames

Fig.5 shows synthesized interpolation result composed with image of arms, hands and the torso. Image of transition hand shapes are deduced by method proposed in paper^[9]. Image of transition arm shapes are selected from a prepared database which stores all images of arms with different shapes. Since arm shapes are limited, building a database for arm shapes is possible. Fig.6 shows some synthesized transition frames of sign language videos based on our method. Finally these continuously playing frames form transition videos, which naturally connect word units' video clips.

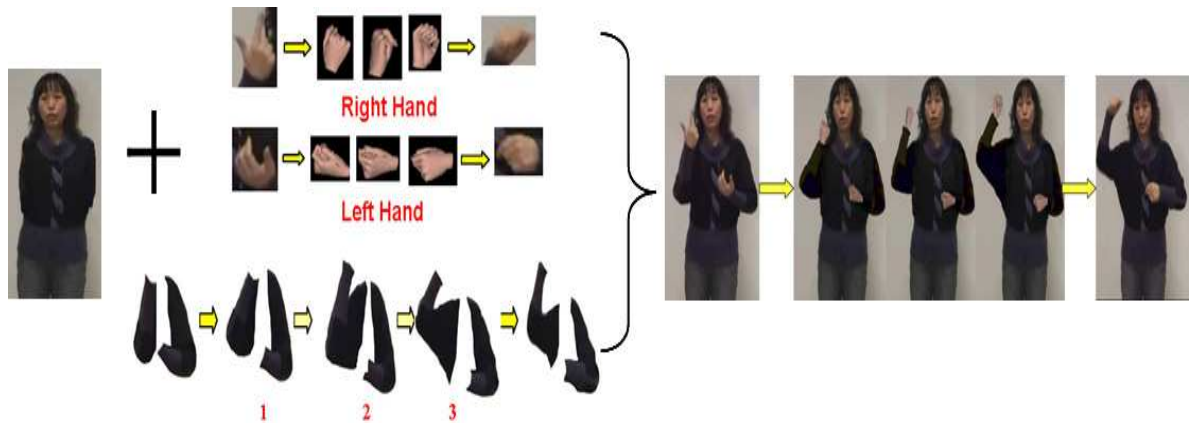


Figure 5. Integration of transition frames

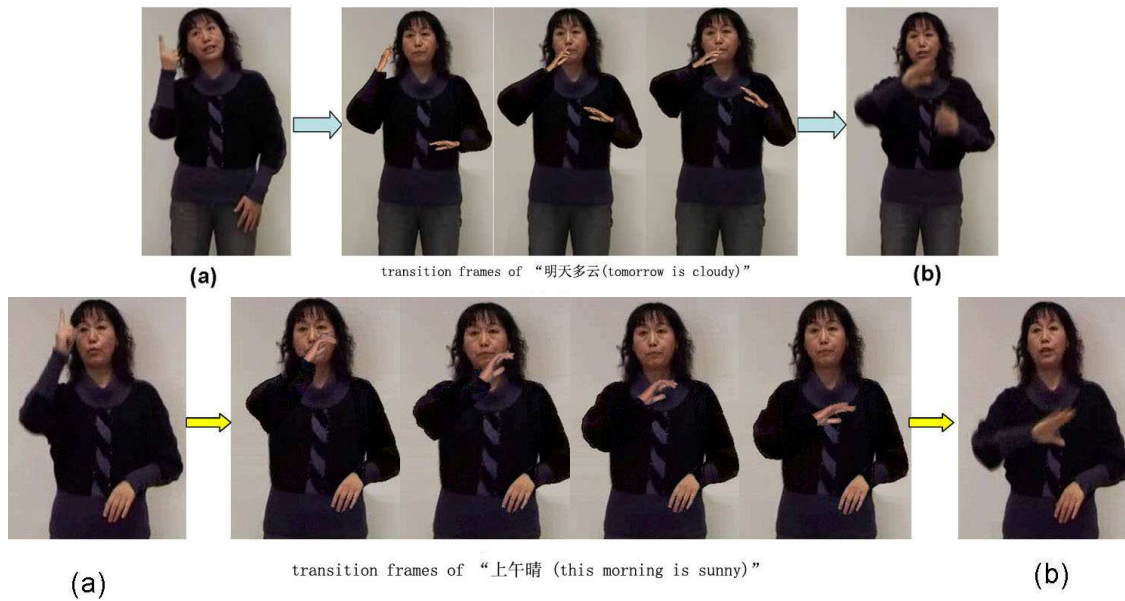


Figure 6. Synthesized transition frames for clip (a) and (b) with method proposed in this paper

5. CONCLUSIONS

This paper presents a method to synthesize transition frames under prosody control for Chinese sign language videos synthesis, which includes two aspects: computing interpolation location under consideration of sign language prosody; deciding number of interpolated frames to ensure actions in each frame is synchronized. Experiments show that transition frames are consistent with sign language video context and synthesized results are smooth and natural.

ACKNOWLEDGEMENT

This research is supported by NSFC (No. 60825203, 61033004, 60973057, U0935004), 973 Program (2011CB302703, Beijing Municipal Natural Science Foundation (4112008), PHR (IHLB), Guangdong Provincial Science and Technology project (2010A090100019). The authors thank Beijing 3rd School for the Deaf, who gives us a great help for Chinese sign's videotape.

REFERENCES

- [1] Ong S.C.W. and Ranganath S., June 2005, Automatic Sign Language Analysis: A Survey and the Future beyond Lexical Meaning. In IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 27, No. 6, pp. 873-891.
- [2] Carol N., 2007, SignStream™ Annotation: Conventions used for the American Sign Language Linguistic Research Project, Boston University, Boston, USA.
- [3] Brenda N., 2006. Prosodic Markers and Utterance Boundaries in American Sign Language Interpretation, Gallaudet Univ Pr, Washington, USA.
- [4] Kejia Y., Baocai Y., Lichun W., 2009, CSLML: a markup language for expressive Chinese sign language synthesis, Computer Animation and Virtual Worlds, Vol. 20, No. 2-3, pp. 2347-245.
- [5] John G., Rechar K., Ralph E., Barry-Jonh T., 2004, Virtual Human Signing as Expressive Animation. Symposium on Language Speech and Gesture for Expressive Characters. University of Leeds, U.K., pp. 98-106.

- [6] Yiqiang C., Wen G., Junfa L., Changshui Y., 2006, Multi-Model Behavior Synchronizing Prosody Model in Sign Language Synthesis, Chinese Journal of Computers, Vol. 29, No. 5, pp. 822-827. (in Chinese)
- [7] Slavko K. and Franc S., 1999, Synthesis of the Sign Language of the Deaf from the Sign Video Clips. Elektrotehniski vestnik, Vol. 66, No.4-5, pp. 260–265.
- [8] Wei-Sheng C., 2004, Taiwanese Sign Language synthesis based on PCA and Image overlapping. National Cheng Kung University, Chinese Taipei. (in Chinese)
- [9] Ru W., Dehui K., Lichun W., Baocai Y., 2009, Gestures'smooth motion for chinese sign language synthesis based on video joining. 2009 International Conference on Computational Intelligence and Software Engineering. Wuhan, China, pp. 1-6.

ZAPP – A MANAGEMENT FRAMEWORK FOR DISTRIBUTED VISUALIZATION SYSTEMS

Georg Tamm^{1,2}, Alexander Schiewe² and Jens Krüger^{1,2,3}

¹*Interactive Visualization and Data Analysis Group, DFKI, Saarbrücken, Germany*

²*Intel Visual Computing Institute, Saarbrücken, Germany,* ³*SCI Institute, Salt Lake City, USA*

ABSTRACT

In this work we present the ZAPP, a management framework for distributed visualization systems. The main purpose of ZAPP is to improve the user, administrator, and developer experience in dealing with distributed displays. Even a single user with no administrative knowledge can operate a ZAPP-managed display system. The goal is that practically anyone is capable of operating the display with less than two minutes of training. To achieve this we present a lightweight management framework that links and controls the render workstations, which drive the displays, but also connects to the user's very own mobile devices, such as smartphones or tablets, to enable convenient control over the display. Consequently, the users never need to leave their familiar hardware and operating system environments and can use distributed displays without external help.



Figure 1. A distributed weather analysis and emergency warning system, a visualization scenario running on top of the ZAPP framework.

1. INTRODUCTION

Driven by the rapid evolution of computer simulations, acquisition technologies and computing hardware, datasets are growing at a rapid pace and even with advanced analysis and visualization techniques it becomes ever harder for a single user or specialist to interpret data alone. A method to allow for the real-time collaboration of multiple users is a distributed display system. A tiled display wall for example (Figure 1 right) provides significantly more area and resolution than a simple computer screen or projection and thus promotes the collaboration of multiple individuals. The ability to connect further display resources from remote locations, such as other large displays, simple workstations, or even mobile devices, provides a flexible platform for visual analysis. While the software systems required for such a scenario are very complicated, the user interfaces should not be. In this work, we do not focus on the user interface of a specific visualization application running on such a display, but on user interfaces to manage the display and its applications in general. To our best knowledge this part of the user experience has been largely neglected. While a number of very mature frameworks exist for communication, rendering, and synchronization in distributed display environments, administrators and users in addition require a control layer for installation, maintenance, and configuration and to select, start, and terminate applications. Our proposed framework was developed with the concept in mind that any user can launch and interact with an application on the display with minimal training and with no help by experienced staff controlling the display in the background, which we call “director of the institute proof”. In addition to this stable platform for everyday use of the display, we

also want the system to support development of new applications without compromising the stability of existing software. With these goals in mind we developed the ZAPP framework.

The remainder of this paper is structured as follows: In the next section we will take a look at previous work for distributed displays. In Section 3 we give an overview on ZAPP. In the next section we outline ZAPP from a usage perspective and thereby focus on three main areas – administration, development and application usage. Section 5 will then detail the implementation and architecture of ZAPP, focusing on the different components involved and the communication flow between them. We close the paper with a summary of the results and give directions for future work.

2. RELATED WORK

A substantial amount of research has been done in the area of distributed rendering, where partial results from each render workstation are either reassembled on a single display or routed to multiple displays. Humphreys et al. (2000, 2001) proposed WireGL, a framework for distributed OpenGL rendering on a cluster of workstations. On top of WireGL, Chromium (Humphreys et al. 2002) extends the framework with a more general approach to arrange the cluster workstations by utilizing a modular streaming model. Employing this model also removes the bottleneck of constantly transferring geometry required by render servers over the network, which made efficient fine-grained load balancing difficult with WireGL. Independently of the WireGL branch of systems, the Cross Platform Cluster Graphics Library (CGLX) has been developed (Doerr et al, 2011) specifically targeting distributed, high-performance visualization via a transparent OpenGL interface.

Equalizer (Eilemann et al., 2008) is another parallel rendering framework based on OpenGL with a focus on scalability, flexible configuration, i.e. supporting an arbitrary amount of workstations and displays, and a developer friendly programming model (“minimally invasive”). Parts of the framework are built on the OpenGL Multipipe SDK (Bhaniramka et al., 2005).

Allowing for both rasterization based approaches as well as ray tracing, DRONE (Replinger et al., 2009) is a flexible framework for interactive visual applications rendering and displaying on multiple workstations.

SAGE implements a more general approach than the above frameworks, which focuses mostly on the rendering side (Renambot et al., 2006, Jeong et al., 2006, Nam et al. 2010). It is a distributed and scalable graphics framework with a focus on collaborative visualization allowing the exploration of large-scale scientific data sets, which may be rendered on a variable amount of workstations and displayed on a variable amount of displays, preferably on a tiled display wall. The graphics streaming employed in SAGE is built on previous work, TeraVision (Singh et al., 2004), which is a solution for high-resolution video streaming between arbitrary numbers of workstations. SAGE requires a high-bandwidth network to operate, limiting it to local collaboration. However it has been extended to support collaboration between multiple distant endpoints with Visualcasting (Jeong, 2009).

VTK is a visualization framework packaging a suite of visualization tools under a common interface. The framework has been extended to support cluster-based parallel rendering and delivering results to a single display or a display wall (Moreland and Thompson, 2003). The solution to render to a display wall is built on Chromium and IceT – implementations for both APIs are included. IceT is a parallel rendering framework targeting display walls as output. It is based on algorithms outlined by Moreland et al. (2001). Recently, Fogal et al. (2010) also used IceT to connect distributed memory multi-GPU clusters to visualize large datasets. More closely related to this work is a visualization framework described by Kim (2006) dubbed TileViewer, which supports multiple application areas including image viewing and video display. It includes a GUI to manage the displays and deploy files to the workstations.

In contrast to the above generic frameworks, which are designed to run various types of visual applications, domain-specific solutions exist which have been tailored to a particular area. Application areas include rendering and exploration of large geometric or volumetric data sets, high-resolution image display, video display and information visualization.

Vol-a-Tile (Schwarz et al., 2004) is a distributed volume rendering application able to display high-resolution data sets on a tiled display wall. Correa et al. (2002) present an extension to the iWalk out-of-core rendering system to enable distributed rendering of large static geometric data sets. Nam et al. (2009) describe the integration of ParaView into the SAGE framework to allow high-resolution rendering results to

be visualized on a tiled display wall. JuxtaView (Krishnaprasad et al., 2004) is a distributed, parallel image viewer for ultra-high resolution image data. By distributing data across a cluster of render workstations and employing a caching and pre-fetching strategy to reduce the impact of network latency, arbitrary sized images can potentially be viewed interactively on tiled display walls. With Giga-stack, Ponto et al. (2010) propose a technique to explore multi-dimensional, giga-pixel images in a high-resolution display environment, such as a tiled display wall.

While the above generic frameworks and domain-specific solutions focus on their particular distributed rendering techniques, and thereby in some cases have proven to be an especially valuable contribution to this area, none focus on how to setup, manage and operate a large distributed display system from a usability perspective in a real-world, productive and possibly public environment. In this situation it is highly desirable for the system to be installed and accessed by non-expert users, i.e. for presentation purposes. Thus, we present ZAPP, a management framework to deploy, maintain and run distributed visualization applications in a flexible, stable and user-friendly way.

3. OVERVIEW

ZAPP is a lightweight management framework to run and collaboratively interact with distributed visual applications, which output to multiple displays, i.e. via a tiled display wall. The framework has been specifically designed to make development, deployment and maintenance of applications as well as setting up and configuring the framework and finally running applications easy, flexible and stable.

3.1 Basic Architecture

Before starting with ZAPP, an initial hardware setup has to be available to deploy the framework on (see an example in Figure 2). This setup consists of a number of render workstations or nodes, each powering a single or multiple displays and running the distributed application. In addition, one machine must run control software; we call this machine the “control node”. This control node is responsible for launching applications and for configuring the overall display alignment. It directly communicates with and keeps track of the available render nodes and their displays. Therefore, the render nodes register with the control node and periodically notify it about their current state. Render and control nodes are normally connected via a reliable high-bandwidth network. Furthermore, one or multiple machines can be used as UI nodes, which issue commands to the control node. UI nodes run a user interface to launch and terminate applications and, optionally, an application-specific user interface to interact with a running application. UI nodes may also be mobile devices connected wirelessly to the control node. Note that the control node may be a render and UI node at the same time, thus a minimal ZAPP configuration can run on a single machine. The different nodes and the software components running on them will be detailed in the following sections.

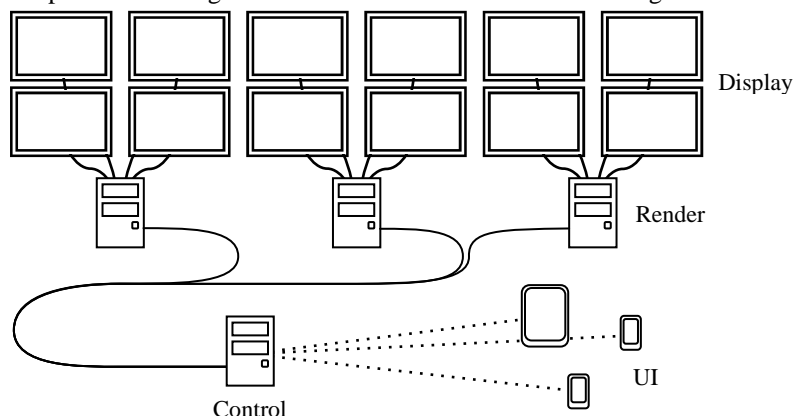


Figure 2. Conceptual setup of a distributed display environment. Note that nodes do not necessarily correspond to separate systems, i.e. the control node can also be a render node.

4. USAGE

We will now detail the usage of ZAPP from the perspective of an administrator, a developer and a user who runs and interacts with applications.

4.1 Administration

Given the above hardware setup, ZAPP can be installed in a few steps. ZAPP provides an installer, which should be copied to a USB-device or a network share. First, this installer is used to prepare the control node. The initial setup is done using a GUI manager, which guides the installer through the individual steps. This involves setting up the ports the control node uses, defining an optional password, which is required to start and interact with applications, and restricting permission to a range of IP addresses. Note that all these steps may be skipped. In this case, default settings are chosen.

After installation, the control node process is started and waits for render nodes to connect. The next step is to setup the render nodes, which can be done using the same installer that is used for the control node. This procedure can be fully automatic if the installer was run from a writable medium. In this case the control node's configuration was integrated into the render node installer. If this is not possible, another GUI lets the administrator configure these settings manually. Also, a name can be chosen for each render node, which will be used to identify it at the control node. After the ZAPP process is running on a render node, it will automatically detect the number of displays attached to it and notify the control node.

The final step is then to configure the overall display alignment, also referred to as the display grid. This step is performed on the control node. There, each render node display can be mapped to a two-dimensional coordinate to define its position within the grid. A tile is defined as part of the grid identified by its coordinate and may be empty if there is no display linked to it. This can be used to account for holes in the physical display wall. In addition, the physical size of each display and its bezel can be set up individually or collectively for all tiles. Depending on an application's requirement, the bezel will either be considered a hidden part of the available display space or be ignored. Consequently, different kinds of tiles can be combined flexibly to setup a display grid using our framework. Of course, the recommended setup would be a grid representing a homogeneous tiled display wall.

After the alignment procedure is done, ZAPP installation is complete. Maintenance can be done at runtime. The render nodes will periodically confirm their presence and display setup with the control node. Should a render node or displays attached to it become unavailable, ZAPP will automatically detect this and disable the corresponding tiles in the display grid so application behavior can remain consistent. Adding render nodes can be done at runtime by repeating the render node installation process. At any time, the display grid can be edited to adapt to missing or additional displays using the manager GUI.

Installing ZAPP enabled visualization applications is straightforward. Application binaries and settings are packaged and can then be selected from the manager for deployment on the render nodes. An application can be flagged as experimental indicating it should not be used in a production environment yet. Afterwards, the application can be launched directly from the control node or indirectly through a UI node. The software to launch applications from a UI node has to be installed separately. Since a UI node never directly communicates with the render nodes, the only setup required is the address and port of the control node and possibly a password. The control node ultimately processes the user input and forwards it to the render nodes. Currently, ZAPP provides mobile launchers for Apple's iOS device family, and an Android version is currently being developed.

4.2 Development

Every ZAPP application consists of two parts. The first is the server, which runs distributed on the render nodes and displays the visual content. The second is a client, which runs on the control node and is responsible for managing and synchronizing the application state, as well as issuing interactive commands from a user to the servers. A ZAPP application may have a third part, the UI, which runs on UI nodes and provides the user with an interface for interaction.

ZAPP provides several APIs to support developing each of these parts of an application. Templates integrating the APIs are available for the server, client and UI and can be used as a ready-to-go entry point

for development. The network and synchronization API is used for communication between servers and clients as well as between clients and the UI. It is a platform-independent library for networking, which can be used in any context. The API includes functionality to synchronize the frame rate across the render nodes.

The display grid API is used on the client to query information about the render nodes and their display alignment. It maps a 2D grid coordinate to the corresponding render node and display so commands can be addressed to a specific tile transparently. Also, this API supports generating transformation matrices for each tile to represent a global transformation on the grid (i.e. to rotate a large image which spans several tiles). Considering a tile's bezel can be optionally enabled when generating a matrix. Information about the current display grid will be automatically passed to an application on start-up. This API is also platform-independent.

The multi-monitor rendering API is available for the servers. It provides information for each available graphics adapter and the displays attached to it. An environment to render to multiple displays can be set up using this API, which has already been done for the server application template. Furthermore, the library provides an abstraction layer encapsulating common functionality of DirectX 10 and 11.

Finally, the mobile rendering API is used on the UI nodes. It encapsulates common functionality of OpenGL ES1.1 and 2 to support developing render features on the UI side of an application: an application may want to provide the user with a preview while rendering the final result on the display wall.

Note that using the aforementioned APIs and the application templates greatly simplifies creating or integrating existing applications, but none of these APIs are mandatory. Our framework is ultimately a management solution. Thus in the end, the developer is responsible for efficiently distributing an application across the available render nodes and displays, which may include distributing data, assigning render tasks, performing load balancing and choosing the render technique itself (i.e. rasterization or ray-tracing). In contrast, other generic frameworks (i.e. Humphreys et al., 2002, Eilemann et al., 2008) focus on the rendering side and features like data distribution or the render technique may be built-in.

Since ZAPP is able to fully operate on a single machine with any number of displays, development and debugging can be done locally before deploying the application to the production environment. An example application we developed is the TeraPixel Viewer, which allows exploring stacks of arbitrary sized images.

4.3 Using Applications

While launching applications can be done directly from the control node, this is only intended for developers. A casual user's entry point is a UI node, which provides a launcher to start and terminate applications. A user runs the launcher, which has been set up to connect to the control node, and then selects and runs any of the available applications (see Figure 3). If an application provides a specific UI for interaction, this part is automatically started on the UI node. The UI of a ZAPP application has to be installed separately on every UI node to have access to the intended interactive features. Both launcher and UI are ideally deployed on a device familiar to the user. Note that the UI can also be started manually without the launcher to connect to an already running application. ZAPP does not limit the amount of UI nodes connecting. Consequently, an application may allow multiple users collaboratively interacting at once.

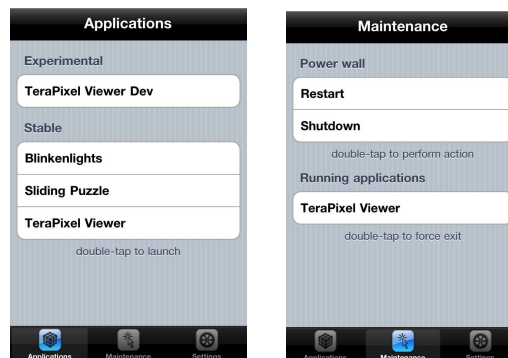


Figure 3. Launcher UI running on an Apple iPhone. In these images the display of experimental apps was enabled and the TeraPixel application is currently running.

An additional feature of the launcher is to send a shut down command to the entire display hardware. This can be used to quickly power down the display wall after a presentation. It is recommended combining this feature with an automatic startup of ZAPP on the control and render nodes. This enables a user to boot the entire system by simply turning on the power.

To evaluate that the setup is indeed “director proof”, we tested our mobile launcher installed on an iPod Touch with 15 non-expert users, who were asked to run the TeraPixel Viewer on a wall with 5x4 displays. No user had prior experience with the wall or ZAPP. Each user was able to browse through the list of applications and then select, run and interact with the TeraPixel Viewer. Especially users with prior iPod knowledge were able to finish the task instantly, while others were able to operate ZAPP with minimal instructions (i.e. explaining the touch display of an iPod).

5. IMPLEMENTATION

We will now detail the software components running on the different nodes and their communication flow with each other.

5.1 Control Node Processes

The control node is the center point of the ZAPP framework. It is responsible for managing the available render nodes and their displays as well as all configuration settings. It also runs the client part of an application, which maintains an application’s state. The control node runs a persistent controller process, which is a TCP server listening for incoming connections from render nodes and UI nodes. Render nodes will establish a persistent connection to the controller and periodically confirm their presence and the displays they have available. Accordingly, the controller will update the display grid configuration. When a render node first connects, the controller will uniquely register it using the nodes name and physical address.

On initial start-up, the display grid is empty. An administrator is thus required to start another process, the manager GUI, to adapt the display grid to newly registered render nodes. Once a display grid is set up, the controller will automatically account for missing displays by deactivating the corresponding tiles in the grid. This happens if a render node or some of its displays become unavailable. Since there is no reliable way to identify a single display on a render node, should a display become unavailable, the controller will not know the exact tile linked to it and therefore deactivate the tile with the highest coordinate assigned to a display of the render node. Ultimately, the manager should be used to revise the display grid should a permanent change occur, i.e. a full column of displays of a display wall is removed or added, which would require resizing the grid. When rebooting the system, the controller will automatically restore the tiles for a registered render node once it connects, so there is no need to setup the grid again.

The controller manages all settings in plain, human-readable text files. An expert user could thus quickly change settings without having to consult the manager. As the control node is the only point where the settings are held, a ZAPP configuration can easily be backed up and re-established by copying a few text files and replacing the files of another installation.

5.2 Render Node Processes

A render node is responsible for outputting visual content to its displays and thus runs the server part of an application. A render node runs a persistent daemon process in the background, which periodically iterates through the available displays and sends that information to the controller. The daemon is also a TCP server, which listens for connections from the controller. The controller will establish a persistent connection to this server once a render node has connected and issue commands to the daemon through it. These commands include launching an application, forcing an application to exit and shutting down the render node all together.

The daemon ZAPP currently provides is a Windows service building on DirectX to obtain information about the available displays. Future extensions to ZAPP will include a platform-independent solution.

5.3 UI Node Processes

A UI node is responsible to provide the user with an interface for interaction and thus runs the UI part of an application. There is no persistent process running here. The user can start a launcher process, which connects to the controller to request a list of available applications and the dimensions of the display grid. The request may or may not include applications flagged as experimental. An application can then be selected for launching. Note that by default an application is launched on the whole grid. However, it is perfectly valid to restrict an application to a sub-grid. Selecting a sub-grid for launching is not yet exposed in the mobile launcher GUI ZAPP provides for iOS devices.

Note that browsing through and launching applications can also be done directly from the manager GUI on the control node, but this is intended for developers only who are authorized to access the control node directly. In the following, we will focus on the scenario involving a UI node.

5.4 Communication flow when launching and running an Application

Launching an application involves several steps (Figure 4). First, the launcher on a UI node sends a request to start a particular application on a sub-grid or the whole grid to the controller. In response, the controller starts the client process on the control node. It thereby passes information about the sub-grid, which for each tile includes the render node, and displays it is linked to, as well as the tile's bezel. A tile may be flagged as deactivated if it is not currently connected to a display. Some displays might be unavailable or there might be a physical hole in the display wall the client needs to know about to guarantee a consistent application behavior. Note that ZAPP can easily be extended to support launching clients remotely on a different machine than the control node. At the same time, the controller determines the render nodes whose displays are part of the sub-grid and then remotely starts the server process on each of these nodes. This is done by sending a launch command to the daemon running on each render node. This command includes information on the displays to use, which may be a subset of the available displays of a render node in case the application runs on a sub-grid, or some displays have not yet been linked to a tile. A daemon then starts the server process of the application.

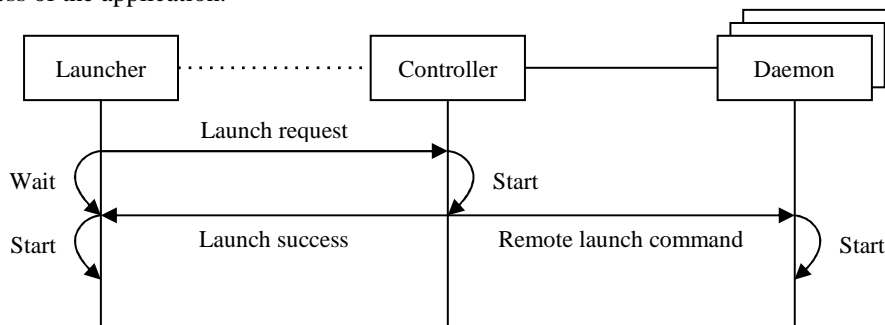


Figure 4. Communication flow between controller, daemons and launcher when starting an application

Furthermore, the controller responds to the launcher to confirm the launch of client and servers. The launcher then starts the UI process so the user can start interacting with the application.

At this point, the involvement of controller, daemons and launcher is over and the application takes over. The servers and the UI establish a persistent connection to the client, which is the central point of communication and runs a TCP server of its own. From here, communication is application-specific and may be implemented in different ways. We will focus on the usual scenario (Figure 5). User input is sent from the UI to the client, which processes it, updates the application state accordingly and instructs the servers to update their displays. The client is especially responsible to keep the state of the distributed application synchronized. It therefore locks the application state until all servers have finished their work before accepting new input for processing.

In contrast to the reliable, high-bandwidth connection between client and servers, the connection between client and UI may be wireless and narrow-banded. Should a user get disconnected or close the UI while an

application is running, a reconnection can be done at any time to regain control over interactive features. The UI will automatically attempt this if it loses connection.

To connect to the client, the client's address and listening ports were passed to servers and UI when starting them. Thus the controller is aware of the ports used by each application's client and attaches this information when remotely launching servers and UI. Should an application allow this, other users may now start the UI on their UI node to join and interact with the running application. Either the UI would directly allow the user to enter address and port of the client or the launcher can be consulted. The controller will confirm the running application with the launcher, which then starts the UI, passing the client information. For the UI node, it makes no difference whether the application was freshly started or already present. However, if a consecutive user has requested the application to launch on a sub-grid different to but overlapping the sub-grid the already running instance is using, the controller will reject the request. This brings us to the possible usage of multiple applications running at once. As long as their sub-grids do not overlap, it is perfectly valid to launch several applications or instances of a single one on the display grid. The space on the grid still available can be requested from the controller by the launcher. The controller will keep track of the applications running on which sub-grid and reject or accept new launch request accordingly.

An application is responsible for terminating itself gracefully and the UI should thus provide the user with a way to do this. An exit request issued through the UI is sent to the client, which forwards it to the servers. Since the controller needs to know about this to maintain the list of active applications, it holds a persistent, idle TCP connection to each client (a loopback connection, given that the client runs on the control node). A disconnection indicates an application has exited. An application can also be forcefully terminated through the launcher or the manager on the control node. This could be used to recover from a hanging application or should the UI get permanently disconnected from the client.

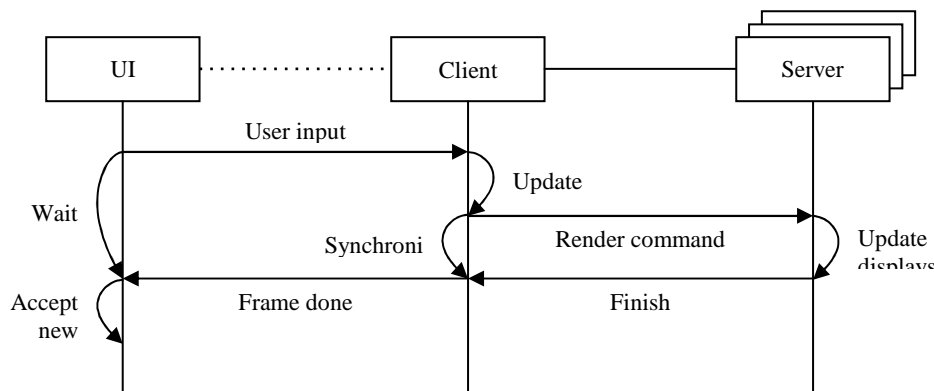


Figure 5. Communication flow between client, servers and UI when an application has been started

6. CONCLUSION AND FUTURE WORK

In this paper we presented ZAPP: a management framework for distributed, high-resolution visualization systems. To our best knowledge this is the first solution that specifically targets the administration and launch process of programs on such a system. In a small informal survey we validated that arbitrary casual users are able to operate our framework with minimal instructions.

In the future we will continue to actively improve ZAPP. We will add an automatic system to layout display grids. This system will incorporate the camera found in mobile devices to automatically detect the alignment of displays. We are also preparing to distribute ZAPP as an open source application. Therefore, we will make sure that the entire framework can be run platform independently, i.e. we will provide an OpenGL based multi-monitor rendering API and extend the background process on the render node, which currently is a Windows service, to be able to run as a POSIX daemon. In addition, we will provide our demonstrative applications such as the TeraPixel Viewer as open source.

ACKNOWLEDGEMENTS

This research was made possible in part by the Intel Visual Computing Institute, by the Cluster of Excellence “Multimodal Computing and Interaction” at the Saarland University, by the NIH/NCRR Center for Integrative Biomedical Computing, P41-RR12553-10 and by Award Number R01EB007688 from the National Institute Of Biomedical Imaging And Bioengineering. The content is under sole responsibility of the authors.

REFERENCES

- Bhaniramka, P. et al., 2005, OpenGL multipipe SDK: a toolkit for scalable parallel rendering. *In Proceedings IEEE Visualization*. Minneapolis, MN, USA.
- Correa, W. T. et al., 2002, Out-of-core sort-first parallel rendering for cluster-based tiled displays. *Proceedings of the Fourth Eurographics Workshop on Parallel Graphics and Visualization*. Aire-la-Ville, Switzerland.
- Doerr, K. and Kuester, F., 2011, CGLX: A Scalable, High-Performance Visualization Framework for Networked Display Environments. *IEEE Transactions on Visualization and Computer Graphics*. Volume 17 Issue 3.
- Eilemann, S. et al., 2008, Equalizer: a scalable parallel rendering framework. *ACM SIGGRAPH ASIA 2008 courses*. Singapore, Singapore.
- Fogal, T. et al. 2010, Large Data Visualization on Distributed Memory Multi-GPU Clusters, *High Performance Graphics*, Saarbrücken, Germany
- Humphreys, G. et al., 2000, Distributed rendering for scalable displays. *Proceedings of the 2000 ACM/IEEE conference on Supercomputing*. Washington, USA.
- Humphreys, G. et al., 2001, WireGL: a scalable graphics system for clusters. *Proceedings of the 28th annual conference on Computer graphics and interactive techniques*. New York, USA.
- Humphreys, G. et al., 2002, Chromium: a stream-processing framework for interactive rendering on clusters. *Proceedings of the 29th annual conference on Computer graphics and interactive techniques*. New York, USA.
- Jeong, B. et al., 2006, High-performance dynamic graphics streaming for scalable adaptive graphics environment. *Proceedings of the 2006 ACM/IEEE conference on Supercomputing*. New York, USA.
- Jeong, B., 2009, Visualcasting – scalable real-time image distribution in ultra-high resolution display environments. Dissertation, University of Illinois, Chicago, IL, USA.
- Kim, S-J., 2006, *The diva architecture and a global timestamp-based approach for high-performance visualization on large display walls and realization of high quality-of-service collaboration environments*. California State University, Long Beach, CA, USA.
- Krishnaprasad, N. K. et al., 2004 JuxtaView - a tool for interactive visualization of large imagery on scalable tiled displays. *Proceedings of the 2004 IEEE International Conference on Cluster Computing*. Washington, USA.
- Moreland, K. et al., 2001, Sort-last parallel rendering for viewing extremely large data sets on tile displays. *Proceedings of the IEEE 2001 symposium on parallel and large-data visualization and graphics*. Piscataway, USA.
- Moreland, K. and Thompson, D., 2003, From Cluster To Wall with VTK. *Proceedings of the 2003 IEEE Symposium on Parallel and Large-Data Visualization and Graphics*. Washington, USA.
- Nam, S. et al., 2009, Remote visualization of large-scale data for ultra-high resolution display environments. *Proceedings of the 2009 Workshop on Ultrascale Visualization*. New York, USA.
- Nam, S. et al., 2010, Multi-application inter-tile synchronization on ultra-high-resolution display walls. *Proceedings of the first annual ACM SIGMM conference on Multimedia systems*. New York, USA.
- Ponto, K. et al., 2010, Giga-stack: A method for visualizing giga-pixel layered imagery on massively tiled displays. *Future Generation Computer Systems*. Volume 26 Issue 5.
- Renambot, L. et al., 2006, Collaborative visualization using high-resolution tiled displays. *ACM CHI Workshop on Information Visualization Interaction Techniques for Collaboration Across Multiple Displays*. Montreal, Canada.
- Replinger, M. et al., 2009, DRONE: A Flexible Framework for Distributed Rendering and Display. *Proceedings of the 5th International Symposium on Advances in Visual Computing: Part I*. Berlin, Germany.
- Schwarz, N. et al., 2004, Vol-a-Tile - A Tool for Interactive Exploration of Large Volumetric Data on Scalable Tiled Displays. *Proceedings of the conference on Visualization '04*. Washington, USA.
- Singh, R. et al., 2004, TeraVision: a distributed, scalable, high-resolution graphics streaming system. *Proceedings of the 2004 IEEE International Conference on Cluster Computing*. Washington, USA.

DATA VISUALISATION AND VALIDATION FOR HYDROLOGICAL MODELS

Karsten Rink, Thomas Fischer and Olaf Kolditz¹

Helmholtz Centre for Environmental Research, Dept. of Environmental Informatics, Leipzig, Germany

¹Dept. of Applied Environmental Systems Analysis, Technical University of Dresden, Germany

ABSTRACT

We propose a data exploration framework for the visualisation and evaluation of hydrological data for the enhancement of numerical simulations. A user-controlled 3D space allows an assessment of large input data sets. Typical examples for inconsistencies between heterogeneous data sets are given and it is demonstrated how these can be detected via a number of modifications done to the scene. Also, the framework allows for the discretisation of the model domain by the generation of finite element meshes based on geo-metric input data. A number of measurements are provided to verify the quality of these meshes and evaluate their adequacy for a subsequent numerical simulation of hydrological processes such as groundwater flow. We demonstrate the application of the framework in two case studies of different scale in central Germany.

KEYWORDS

Visualisation, Data Validation, Mesh Quality, Hydrology, Climate

1. MOTIVATION

The simulation of hydrological processes is a widely researched subject with a large variety of applications. Aspects such as water resource management have become more and more important in recent years. Especially in arid regions it is crucial to develop a sustainable management scheme for agriculture. Other applications include the distribution of chemicals originating from fertilisers, the extraction of geothermal energy or the detection of potential erosion processes in mining areas.

Typical simulations calculate processes such as groundwater discharge and recharge or overland flow. The input data for these simulations are often based on data originating from different sources. Examples are borehole data from drillings for wells or other natural resources, satellite images such as digital elevation models (DEM) or information acquired via precipitation- or groundwater gauging stations. Often these data sets have large differences in scale; the format varies between image raster, point- or vector data as well as time series data. Due to these inhomogeneities along with the wide range of acquisition methods and the application of various post-processing techniques it is not uncommon that there are inconsistencies between any two given data sets. Obviously, if the simulation of processes such as groundwater recharge are based on faulty or conflicting information they might produce erroneous or deceptive results. Therefore, it is necessary for an expert to recognise these potential problems and correct them appropriately. To support this process, we propose a framework defining a simple workflow for the algorithmic and visual validation of hydrogeological data.

In section 2 we give a brief overview over existing approaches to the visualisation of geoscientific data in 3D. Details on the proposed framework are given in Section 3. Section 4 shows the application of the presented techniques on two model regions of different scale. Advantages and drawbacks of our framework are summarised in section 5.

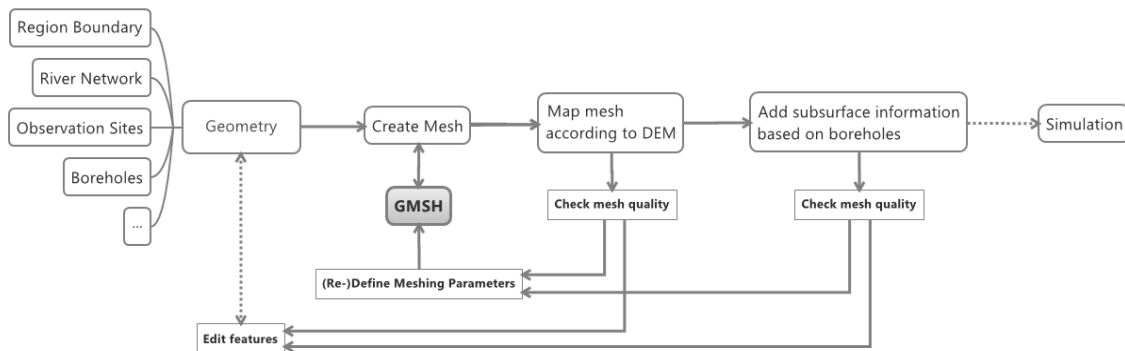


Figure 1. A typical workflow for setting up a hydrological model based on a variety of input data. Note that this workflow can vary considerably depending on the existing input data and the processes that should be simulated. At each point in the workflow correction of either mesh or input data might be necessary. Examples for modifications of input data include the simplification of region boundaries or rivers to avoid unnecessary numbers of small or deformed elements or the subdivision of stratigraphical layers to allow mesh elements with a good edge ratio (see section 3.2 for details).

2. STATE OF THE ART

While the use of 3D data analysis is extensively used for the analysis of medical imagery or technical models, there exist only few tools for the 3D visualisation of geoscientific data. Most geographic information systems (GIS) offer only a layered 2D view of data sets (Andrienko, 2006). A number of commercial tools offer 3D visualisation of specific types of data for special applications. Examples are the Aquaveo Groundwater Modeling System (GMS¹), the ESRI ArcGIS extension 3D Analyst as well as a number of geoscientific CAD tools such as Gocad² which are often used in the mining or petroleum industry. Most case studies found in literature employ a combination of the above commercial software, combining GIS projects with 3D modelling software for the visualisation of a specific model region (Johnes, 2009; Royse, 2008).

A number of specialised solutions are concerned with the stereoscopy visualisation of geoscientific data in a virtual reality environment (Zehner, 2010) or for modelling for a certain subset of applications such as geothermal fields (Alcaraz, 2011).

3. VISUALISATION AND VALIDATION OF DATA

3.1 Assessment of Hydrological Data

We developed a data exploration framework for the visualisation and validation of data and the subsequent simulation of hydrological processes using C++ and Qt. A typical workflow (see figure 1 for an example) for a given model region using our framework begins with the visualisation and subsequent assessment of all relevant input data in the graphical user interface. These data sets usually include the boundary of the model region (often based on river catchments or known boundaries of aquifer systems), digital elevation models acquired via satellite to model terrain of the area and other objects of interest such as river networks or boreholes as well as observation facilities such as precipitation gauging stations. These data sets are arranged in a 3D environment that allows the user to move freely through the visualised objects and manipulate their appearance with a number of tools provided by the framework. If the data sets are considered correct and consistent with each other, it is then possible to generate a mesh discretisation of the domain (i.e. the model region) based on the available geometric information.

¹ <http://www.aquaveo.com/gms>

² <http://www.gocad.org>

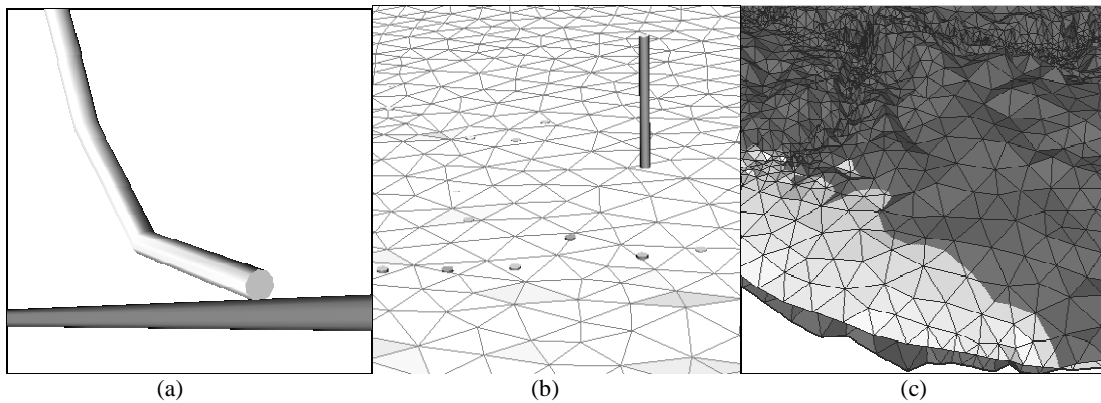


Figure 2. Examples for inconsistencies between data: (a) polylines not quite connected to each other, (b) borehole data with incorrect z-coordinate, (c) stratigraphic layers intersecting each other.

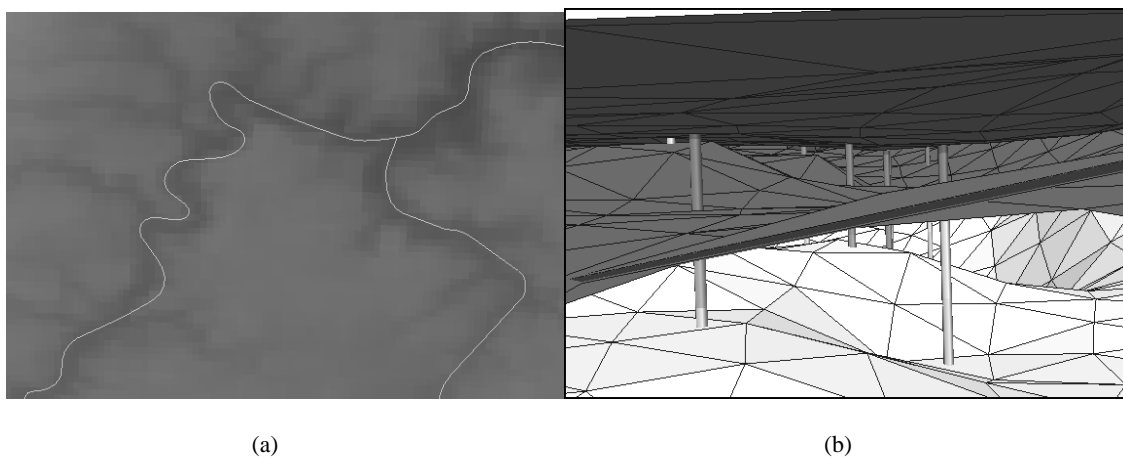


Figure 3. Assessment of geoscientific data sets in 3D. (a) Course of the river not quite matching the DEM, (b) Differences between borehole stratigraphies and interpolated layers.

Given the combined visualisation of all input data sets a large number of inconsistencies between data sets can be detected by a simple visual inspection. Typically such inconsistencies include given river networks not matching the DEM (Fig. 3a), boreholes starting a large distance above or below the surface (Fig. 2b) or geological layers intersecting each other (Fig. 2c). The reasons for all of these inconsistencies can largely be attributed to the heterogeneity of the data sets as well as different acquisition- and post-processing techniques, differences in the time of data acquisition or simple human errors when cataloguing the data. For most of the mentioned issues our framework merely supports the user in detecting the potential problem. Correction of the data is often problem-specific and can be performed using a GIS or other external software specifically designed for modification of geoscientific data sets. A limited number of problems can also be automatically detected such as river outflows not reaching the boundary of the domain (this is often a requirement as this outflow is a boundary condition for the subsequent simulation) (Fig. 2a) or mesh elements of different subsurface layers intersecting each other (Fig. 2c) which is not possible in reality.

The user is supported in detection of possible errors or inconsistencies by a number of well-proven visualisation techniques. Since the visualisation is performed using the open source graphics library VTK (Schroeder, 2006) a large number of filters for modification of the scene are easy to use and the definition of new filters is straightforward. Note that these are not filters in the same sense as filters in digital signal processing but rather VTK components that receive input data from other components, modify it in some way, and allow the output to be used by other VTK components as part of the VTK visualisation pipeline.

The visualisation of any data set can be made transparent or switched off completely. Each data set can be assigned a specific colour or a user-defined colour table. Since all kinds of soil- and rock-types have predefined colours for visualisation in geological maps, this functionality helps experts in handling borehole

data or subsurface models. The super-elevation of data sets is useful as the z-dimension of geoscientific data is usually much smaller than the corresponding lateral dimensions. The optional enlargement of data (such as the visualisation of points or lines as spheres or tubes of variable size) is a necessary requirement for visibility of such objects in large domains. For all data sets a subset of the contained objects can be selected for visualisation, i.e. only certain stratigraphic layers of geological data or single rivers in a river network data set and additional scalar data or textures can be mapped onto data sets to add more information to the evaluation process.

3.2 Mesh Quality

Model regions can be defined either in 2D or 3D space, depending on the process that should be simulated. Domain discretisation for 2D models consists of either triangle- or quadrilateral elements; in 3D space the domain is split into tetrahedra, hexahedra or prism (wedge) elements.

Typically, we employ the open source FEM mesh generator GMSH (Geuzaine and Remacle, 2009) for domain discretisation. Our framework has an interface for GMSH that allows to generate either homogeneous or adaptive meshes and to modify parameters such as mesh density or refinement towards selected locations such as observation sites or river networks. Alternatively meshes can be imported from other specialized tools such as Tetgen³, GMS or Gocad.

Similar to the input data, the quality of the domain discretisation needs to be analysed as well. The numerical simulation of hydrological processes is performed using the Finite Element Method (FEM) in 2D or 3D. Therefore, mesh quality affects the convergence as well as the accuracy of the numerical solution to a large extent. Typical sources for problems with finite element meshes are a well-researched problem (Versteeg and Malalasekera, 2007; Gresho and Sani, 2000) and include the mesh itself being too coarse, large jumps in the volume of neighbouring mesh elements or deformed or unsuitable mesh elements. The first problem can be solved by simply adjusting the mesh density using our framework. However, while more mesh elements may give a higher accuracy, they will also slow down computation and require more memory. The process of finding an optimum mesh density is therefore dependent on the size and topology of the model region as well as the type of process that needs to be simulated. The second problem can be safely ignored as it will not occur due to the tools employed for generation of meshes. For determining mesh quality a large number of measurements are found in literature (Knupp, 2000; Shewchuk, 2002). Our framework offers the following options:

- Edge Aspect Ratio: Shortest edge of the element divided by longest edge;
- Normalised Volume (only applicable for 3D elements);
- Normalised Area: Area of 2D element or minimum surface area of 3D element, respectively;
- EquiAngle Skewness, given by

$$s = \max \left[\frac{\theta_{\max} - \theta_{opt}}{180 - \theta_{opt}}, \frac{\theta_{opt} - \theta_{\min}}{\theta_{opt}} \right] \quad (1)$$

Where θ_{\max} is the maximum angle found between two edges of the element, θ_{\min} is the minimum angle and θ_{opt} is the optimal angle, i.e. 60° for triangle or tetrahedra elements and 90° for quadrilateral or hexagonal elements.

Mesh element quality can then be visualised either directly by colourisation of the mesh (Fig. 4) or via histogram. As with the aforementioned visualisation options mesh quality values can also be thresholded, allowing the user to specifically select elements with poor quality. Note, that the choice of an adequate quality measure depends on the type of process (i.e. partial differential equation) to be solved numerically. Processes such as groundwater recharge consist mainly of layered flows, meaning that large differences between horizontal and vertical element surfaces might have no effect on a correct result. The simulation of mass transport processes explicitly requires a fine mesh resolution in vertical direction to ensure a stable solution (Courant, 1928).

³ <http://tetgen.berlios.de/>

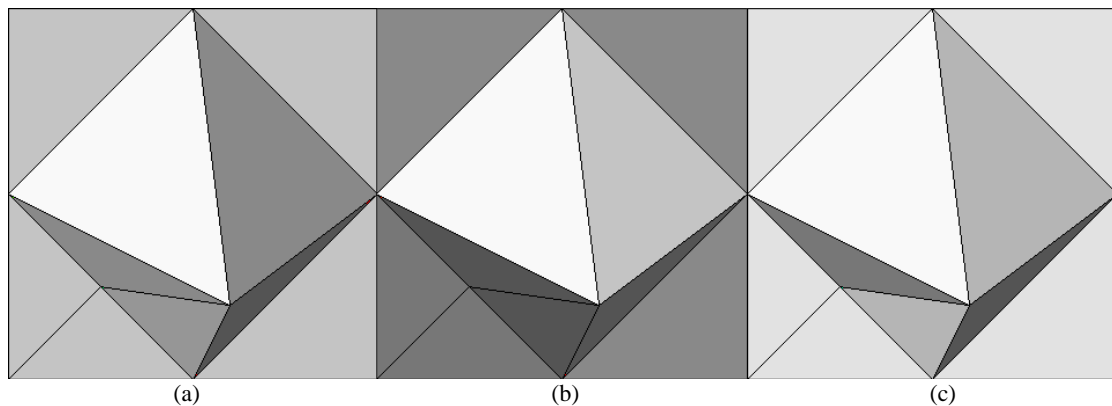


Figure 4. Visualisation of mesh element quality for a test mesh: (a) Edge Aspect Ratio, (b) Normalised Area and (c) EquiAngle Skewness. High intensities indicate elements with high quality, lower intensities imply low quality.

After verification of both input data and mesh quality, additional information based on measured data such as precipitation or the discharge of rivers can be assigned as boundary conditions, followed by an FEM-based simulation of hydrological processes. We employ the open source FEM simulation software OpenGeoSys for numerical analysis (Kolditz et al., 2008, Wang et al., 2009).

4. CASE STUDIES

Our framework has been used by experts for preparation of simulations in a number of case studies. We would like demonstrate the application for data assessment and validation based on two investigation sites in central Germany: the Selke catchment and the Thuringian Basin.

The Selke river is a tributary of the Bode river and located in the Harz mountains in central Germany. The stream has a length of 64km; its catchment with an area of 468 km² has been determined by a hydrological analysis (Fig. 5a). Because of its gradient in land use between the agriculture in the lowlands in the northeast of the catchment and the forests in the low mountain ranges in the southwest, this catchment is of special interest for hydrogeological analysis in the scope of the TERENO project (Zacharias et al., 2011) for the prediction of possible impacts of global climate change. A DEM of the area with a pixel size of 90 m is available at the NASA Earth Resources Observation and Science Center. The area is being profoundly instrumented, providing temporal information via precipitation- and runoff-gauging stations. Due to exhaustive former mining there are also over 5000 boreholes in the area. Based on the stratigraphic information from these boreholes a subsurface model of the Selke catchment has been interpolated based on the method presented in Sun et al., 2011. After verification of the quality of the model (see Fig. 3b), it can be used as the basis for a subsequent simulation of groundwater discharge and recharge. Boundary conditions and source terms are derived from the information gained from the instrumentation in the area.

Visual assessment of these data sets in three dimensions is straightforward. The domain discretisation of the surface including all of the aforementioned input data (Fig. 5b) consists of about 42.000 prism- and tetrahedra-elements (Fig. 5c). While this might not seem overly complex one has to keep in mind that each pixel of the DEM covers 8100 m² while the mean area covered by elements is about 11.000 m².

Obviously it not possible to validate the correctness of the data from the 2D layered view that GIS tools offer. Conversely, many modelling software products for subsurface structures do not support the import of geographic data. The combination of these aspects in the present framework allows a more straightforward workflow for experts and additionally allows the direct numerical simulation of hydrological processes based on the imported data (Fig. 5d).

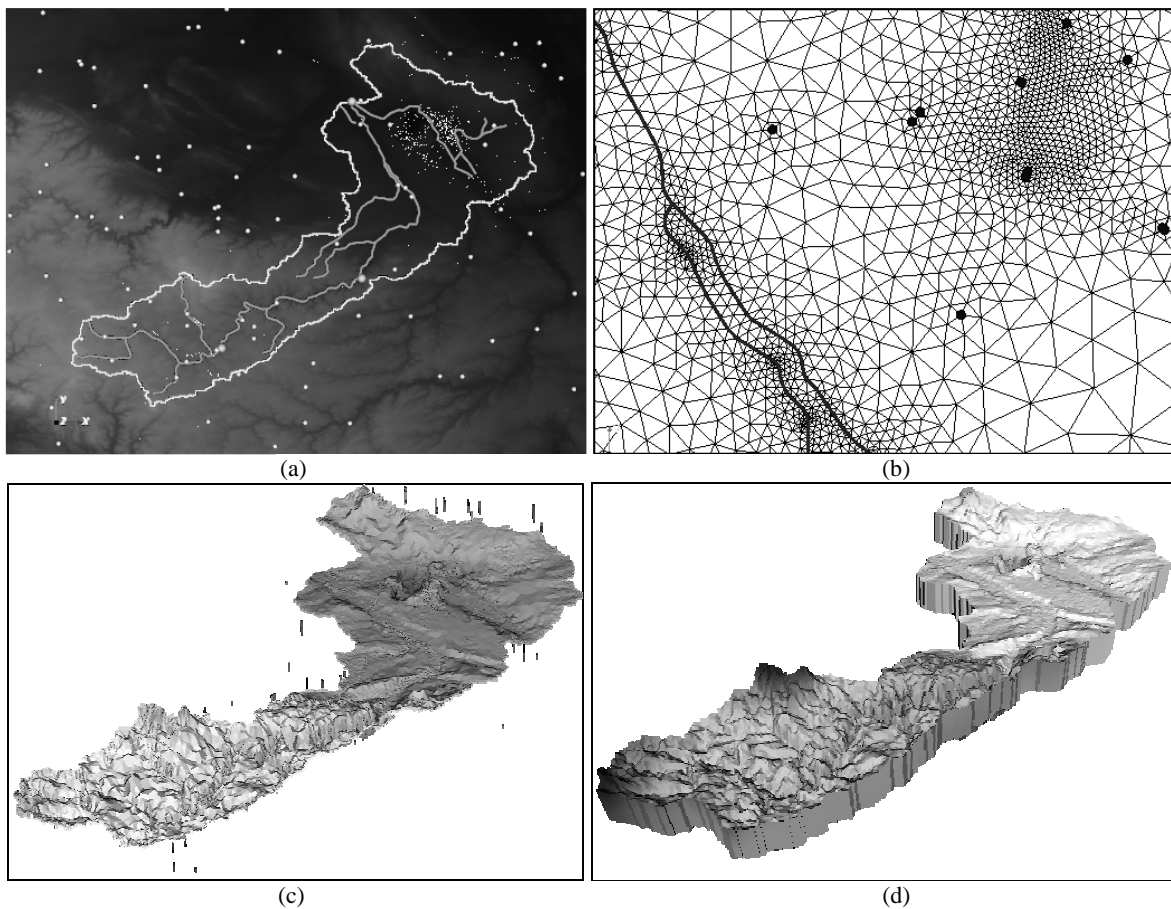


Figure 5. Data visualisation for the Selke river case study: (a) Input data including the boundary of the model region, the river network and observation sites. The large markers symbolise groundwater- and precipitation gauges, the small markers are boreholes. (b) Section of the adaptive mesh generated from the input data. The marked objects represent rivers and boreholes. (c) Super-elevated visualisation of the surface mesh of the model region including boreholes. (d) Result of groundwater recharge process simulation.

The second case study is concerned with the Thuringian Basin in the scope of the INFLUINS research project. The scope of this project is the investigation of coupled dynamics of near surface and deep flow patterns of fluids as well as mass transport of chemical species in a sedimentary basin. The Thuringian basin is located in Thuringia in central Germany and covers an area of approximately 14.000 km². The geometric model covers the three triassic units Buntsandstein, Muschelkalk and Keuper (Kober, 2009). Since the model region is considerably larger than the Selke catchment study presented in the first example, the resulting mesh consists of 159.038 nodes and 608.504 tetrahedrons. Boundaries of outcrops are included in the mesh as well as fault lines and important rivers (Fig. 6). Visual assessment of a model of this scale is tedious and a general overview over element quality provided by a histogram is helpful for evaluation of the mesh before the numerical simulation of processes.

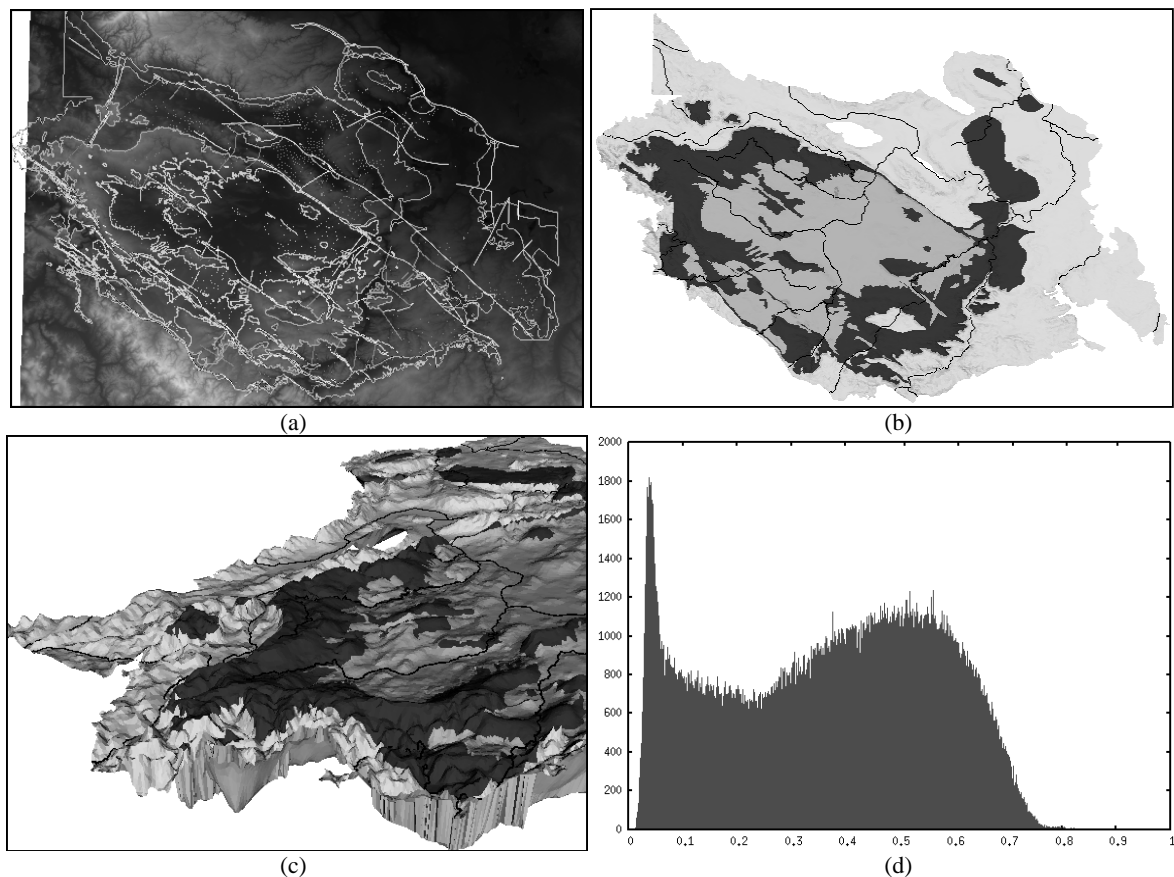


Figure 6. Visualisation from the Thuringian Basin case study: (a) Input data consisting of outcrops, fault lines and river network. (b) Surface view of model region showing the three most dominant stratigraphic layers in the basin as well as the river network. (c) Visualisation of the 3D model used for subsequent mass transport simulation. (d) Histogram of EquiAngle Skewness mesh element quality showing all elements within acceptable bounds (i.e. with $s < 0.8$).

5. CONCLUSIONS AND OUTLOOK

We presented a framework for the visualisation and assessment of hydrogeological data. It combines and augments the capabilities of data visualisation found in GIS and geological CAD modelling software, thus allowing experts to evaluate the whole range of data required for the simulation of hydrological processes. A number of inconsistencies between data sets can be detected either algorithmically or via visual assessment. An adaptive domain discretisation of the model region can be conducted with the optional inclusion of geometrical objects derived from the input data. The quality of generated meshes can be analysed based on a number of established criteria and the result can be written in the form of a histogram or visualised directly on the data.

While the automatic correction of data sets goes beyond the scope of the present framework we will include additional functionality to at least automatically detect and indicate a wider range of potential problems in the future, allowing the user to recognize existing inconsistencies easier. This will prove especially helpful in large scale modelling such as the simulation of fluid dynamics of the Thuringian Basin. At this point the only course for the correction of low quality mesh elements is performing a re-meshing of the whole domain. An automatic correction of such elements is another goal planned for implementation in the framework.

ACKNOWLEDGEMENTS

The research was supported by the Helmholtz Association with the program “Earth and Environment” and the TERENO initiative (Terrestrial Environmental Observatories) as well as the INFLUINS project (Integrierte Fluiddynamik in Sedimentbecken). The authors would also like to thank Feng Sun, Andreas Musolff, Christian Schmidt, Wenqing Wang, Alraune Zech and Björn Zehner for providing data and additional information for the presented case studies.

REFERENCES

- Alcaraz S et al., 2011. 3D Geological Modelling using New Leapfrog Geothermal Software. Proceedings of the 36th Workshop on Geothermal Reservoir Engineering, Stanford, California.
- Andrienko, N and Andrienko, G, 2006. *Exploratory Analysis of Spatial and Temporal Data*. Springer Verlag, Berlin Heidelberg, Germany.
- Courant R, Friedrichs K and Lewy H, 1928. Über die partiellen Differenzgleichungen der mathematischen Physik. *Mathematische Annalen*, Vol. 100, No. 1, pp 32–74
- Geuzaine C and Remacle J-F, 2009. Gmsh: a three-dimensional finite element mesh generator with built-in pre- and post-processing facilities. *International Journal for Numerical Methods in Engineering*, Vol. 79, No. 11, pp 1309–1331.
- Gresho P M and Sani RL, 2000, *Incompressible Flow & Finite Element, Volume 1*. John Wiley & Sons,
- Johnes, R et al., 2009. Integration of regional to outcrop digital data: 3d visualization of multi-scale geological models. *Computers & Geosciences*, Vol. 35, pp 4–18.
- Kober M, 2009. *Dreidimensionales Untergrundmodell des Thüringer Beckens*. Diploma thesis. Faculty of Chemical and Earth Sciences, Friedrich-Schiller-Universität Jena.
- Kolditz O, et al., 2008: Numerical analysis of coupled hydrosystems based on an object-oriented compartment approach. *Journal of Hydroinformatics*, Vol. 10, No. 3, pp 227–244.
- Knupp P M, 2000. Achieving finite element mesh quality via optimization of the Jacobian matrix norm and associated quantities. Part I—a framework for surface mesh optimization. *International Journal for Numerical Methods in Engineering*, Vol. 48, pp 401–420
- Royse K R, Reeves H J and Gibson A R, 2008. The modelling and visualization of digital geoscientific data as a communication aid to land-use planning in the urban environment: an example from the Thames Gateway. *Geological Society, London, Special Publications*, Vol. 305, pp. 89-106.
- Schroeder W, Martin K and Lorensen B, 2006. *Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*. 4th Edition. Kitware, Inc.
- Shewchuk J R, 2002. What is a Good Linear Finite Element? Interpolation, Conditioning, Anisotropy, and Quality Measures. *Technical Report*. Department of Electrical Engineering and Computer Science, University of Berkeley
- Sun F, et al., 2011. Groundwater drawdown at Nankou site of Beijing Plain: model development and calibration. *Environmental Earth Sciences (accepted)*, DOI 10.1007/s12665-011-0957-4.
- Versteeg, H K and Malalasekera W, 2007, *An introduction to Computational Fluid Dynamics: The Finite Volume Method*, Prentice Hall.
- Wang W, Kosakowski G and Kolditz O, 2009. A parallel finite element scheme for thermo-hydro-mechanical (THM) coupled problems in porous media. *Computers & Geoscience*, Vol. 35, No. 8, pp 1631–1641
- Zacharias S, et al., (2011). A network of terrestrial environmental observatories in Germany. *Vadose Zone Journal (submitted)*
- Zehner B, 2010. Mixing Virtual Reality and 2D Visualization - Using Virtual Environments as Visual 3D Information Systems for Discussion of Data from Geo- and Environmental Sciences. *Proceedings of the International Conference on Computer Graphics and Applications (GRAPP2010)*, Angers, France.

A STUDY ABOUT INFLUENCEABLE PARAMETERS IN THE CHOICE OF VISUALIZATION TECHNIQUES BASED ON GROUNDED THEORY

Juliana Keiko Yamaguchi and Maria Madalena Dias
State University of Maringá
Av. Colombo, 5.790 - Maringá - Paraná - Brazil

ABSTRACT

When visualization techniques are used to analyze results from a knowledge extraction process, it is necessary to choose the techniques according to certain criteria to ensure that the display is able to represent data as best as possible, especially when they are applied directly on data as an inference tool to extract information or as a way of data mining to get knowledge. In this context, this paper describes the parameters to be considered in the choice of these techniques in order to assist the analyst in the task of data exploration. These parameters are: data type, task type, data volume, dimensionality (number of attributes) and positioning of the attributes in the graph. The method used to identify the parameters is based on the Grounded Theory methodology. Further, this paper presents an analysis about the relation between that parameters and the visualization techniques of the different categories.

KEYWORDS

Data visualization; visualization techniques; knowledge discovery.

1. INTRODUCTION

We need to get useful information from data. In the other side, the amount of data is becoming large, and this process is accelerating. It is pushing the development of techniques to extract knowledge, such as data mining and knowledge discovery in databases (KDD).

Visualization techniques have been used to improve understanding about the outcome of these procedures. According to Ankerst (2001), they can be integrated into data mining techniques in three different ways: 1) to preview data; 2) to view final results; and 3) partial results of data mining algorithms. In this scenario, the visualization facilitates insights about final results of knowledge extraction processes.

Visualization can also be used as a mean to exploring data directly. Therefore, it is necessary to apply appropriate visualization techniques to data under analysis to correctly understand the final results. So, it is very necessary to know the functionality of methods and data mining algorithms and how their parameters must be fed, and it is important to understand the factors we must consider when using visualization techniques, as they may contribute to the knowledge extraction process. Thus, in this work we use Grounded Theory methodology to identify the influenceable parameters in the choice of the visualization techniques.

In the next section is presented a brief introduction to Grounded Theory, then the identified parameters are described according to this methodology. Following, we show some issues about the association of the parameters to the visualization techniques to illustrate how important they are in the process of choosing the visualization techniques. Finally, the last section presents the conclusion.

2. GROUNDED THEORY AS A RESEARCH METHODOLOGY

Grounded Theory (GT) was proposed in 1967 by Glaser and Strauss, as an alternative to the traditional scientific method (Rodrigues et al, 2004). GT does not start from any hypothesis. The first step in GT is data collection, using interviews or forms answered by groups of people or comments about the object-

environment of research (Coleman & O'Connor 2007, Matavire & Brown 2008). A systematic analysis of collected data extracts series of codes from the texts based on key-points of the subject. The codes are grouped according to similar concepts in order to make them more understandable. The concepts may also be reordered to form new categories, which are the foundation for a new theory.

In researches which use Grounded Theory methodology, data are gathered usually through field observation and / or interviews. In this work, data are collected from analysis of publications in visualization field. Each of the analysed work approaches a visualization technique category. In Exploratory Data Analysis field, for example, Morgenthaler (2009) and Myatt (2007) discuss about standard 1D-3D visualization techniques. Icon-based category is the subject of Wyatt (2008), Ward (2002) and Lee et al. (2003). Visualization techniques from Geometrically transformed displays category are studied by Inselberg (2008) and Hofmann (2008). Keim (2005) and Ankerst (2001) focused on pixel-oriented techniques. Shneiderman (2006) and Herman et al. (2000) studied Graph-based visualization techniques. The informations presented in these and more related works, such as the data and the visualization technique used, placed the role of usual interviews. Dick (2005) argues that literature can be a data source in Grounded Theory studies.

As mentioned above, data must undergo a codification process to classify and organize them into categories. When collecting data, they are analyzed through constant comparison, and the previously gathered information should be re-examined to ensure the consistency of the resulting categories of the coding process. When the verification of new information does not add or modify the existing information, we find a theoretical saturation. At that point, the theoretical sampling is formed.

The method used in codification step on the collected data was the key point coding method (Allan 2003). The concepts are abstracted from text fragment. So, they are rearranged into categories and, finally, they are related to each other and organized to build the theory. The next section describes how the procedure of detecting parameters (the concepts) was performed.

3. IDENTIFICATION AND DESCRIPTION OF THE PARAMETERS

This section describes the parameters considered in the choice of visualization techniques that were identified using the encoding method and the GT principles. They are: data type; task type; data volume, and data dimensionality; and positioning of the attributes in the visualization chart.

Table 1 shows the open coding procedure applying the key point encoding strategy (Allan 2003). The table relates the concepts abstracted from the literature to the respective references, and the codes corresponding to the identified parameters we discuss in the sequence.

3.1 Data Type

When we choose a visualization technique, we must consider the handled data type. In this work, we consider only the data nature in the analysis process. It refers to the primitive properties of attributes, distinct between quantitative or qualitative values. Quantitative data correspond to numeric values, which may be related to a range or interval (continuous quantitative data). For example, measures of speed. They may represent counts of things or objects (discrete quantitative data). Qualitative data represent the categorization according to data characteristics, which may have an order (ordinal qualitative data). For example, schooling grade. They also may nominate a set of similar data (nominal qualitative data). For example, the marital status attribute in some form.

Table 1. Key point coding method applied on collected data

Key points	References	Code
Visualization techniques can be classified, among other criteria, by data type	Shneiderman (1996) Freitas et al. (2001) Keim (2002)	Data type
Task type is one of the aspects considered in classification of visualization techniques, which provides means of interaction between the analyst and the display	Shneiderman (1996) Keim (2002) Pillat et al. (2005)	Task type
Visualization techniques are subject to some limitations, such as the amount of data that a particular technique can exhibit	Keim e Kriegel (1996) Oliveira e Levkowitz (2003) Rabelo et al. (2008)	Volume
Visualization techniques can also be classified according to the number of attributes	Shneiderman (1996) Grinstein et al. (2001) Keim (2002) Oliveira e Levkowitz (2003)	Dimensionality
In some category of visualization techniques, distribution form of attributes on the chart can influence the interpretation about the representation, such as correlation analysis, in which the relative distance among the plotted attributes is relevant for observation	Ankerst (2001) Oliveira e Levkowitz (2003) Inselberg (2008) Klippel et al. (2009)	Positioning of attributes

3.2 Task Type

The works of Shneiderman (1996), Keim (2002), Pillat et al. (2005), use task type as the key to classify the visualization techniques. So, we consider the most common task types in this work, such as:

- Overview: display data completely;
- Correlation among attributes: the degree of relationship among attributes can reveal patterns of behavior and trends;
- Recognition of rules, patterns and important characteristics of data;
- Identification of clusters: attributes that have similar behavior;
- Detecting outliers: data set with non typical characteristics when compared with remainder data.

3.3 Volume and Dimensionality

It is important to detect how much a visualization technique is capable to represent data as they increase in volume and dimension (number of attributes). According to Rabelo et al. (2008), there is no consensus to define when data have high or low dimensionality. Oliveira and Levkowitz (2003) assert that the interpretation of data with more than five dimensions is difficult for humans. Then, they suggest, for general purposes, guidelines for characterizing dimensionality and volume as described in the Tables 2.a and 2.b.

Table 2. Magnitude units considered to distinguish (a) the dimensionality and (b) the volume of data(a)

Dimensionality	N. of attributes
Low	≤ 4
Medium	5 to 9
High	≥ 1

(a)

Volume	Magnitude order
Small	10^1 to $10^2 = 10$ to 100
Medium	10^3 to $10^5 = 1000$ to 10000
High	10^6 to $10^7 = 1000000$ to 10000000

(b)

3.4 Position of Attributes

The position of attributes parameter is related to the visualization techniques rather than to the characteristics of data. In the visual analysis, changing the seats of the attributes in the charts can improve in a significantly way the knowledge discovery process depending on the visualization technique used. For example, in the correlation analysis task, the order in which the attributes are displayed is important, since correlation can become more remarkable when attributes of interest are in positions close to each other in the chart. It happens in the Parallel Coordinates technique (Inselberg 2008), for example.

In the clustering task, we may use different orders to distribute the attributes for chart properties to easily identify groups of data with similar characteristics or behavior. For example, it's the case of the Star Glyph (Klippel et al. 2009).

4. ASSOCIATION OF PARAMETERS AND VISUALIZATION TECHNIQUES

After the parameters identification, the analysis process according the GT method continues by ordering them through the association with the type of visualization techniques. In this work, we use the taxonomy defined by Keim (2002). It organizes the visualization techniques in some categories: standard charts (for 1D to 3D) displays; icon-based displays; geometrically transformed displays; pixel-oriented displays; and hierarchical or graph-based displays. We chosen some visualization techniques of each category to illustrate the analysis of the parameters. Among standard charts, it was selected: the histogram; the Box Plot; the Scatter Plot and the Contour Plot. Among Icon-based: the Chernoff Faces; the Star Glyphs and the Stick Figure. Among Geometrically transformed displays: the Scatter Plot Matrix and the Parallel Coordinates. Among Pixel-oriented displays: the Query-dependent and the Query-independent techniques. Among Graph-based or Hierarchical: the Graph; the Cone Tree; the Treemap; the Dimensional Stacking and the Mosaic Plot.

4.1 Standard Charts 1D – 3D

Standard charts represent data as one to three attributes. They are widely used in statistics to estimate certainty about an event, about a particular attribute, to illustrate the frequency distribution of an attribute, and to provide charts to make comparison and classification of data, to verify the correlation of attributes and to identify outliers which can be observed, for example, using box plot. Generally, the data to be displayed are quantitative in nature and undergo treatment processes, such as methods of data projection and data reduction (Myatt 2007), consequently resulting in a small volume of data.

For this category of techniques, positioning of attributes does not influence interpretation of the chart, due to low dimensionality and spot of attributes in the chart be well defined. We should also consider that such types of charts are generally used to analyze the behavior of an attribute according to a second or third one, such as the scatter plots and contour plot. Table 3 summarizes the main features of the techniques to 1D-3D data.

Table 3. Analysis of parameters for standard charts

Category		Concepts			
1D – 3D	Task	Data type	Dimension	Volume	Position
Histogram	Frequency distribution analysis, patterns verification and outliers detection	Any qualitative and quantitative data	1D	Small	It does not influence
Box Plot	Idem the Histogram	Quantitative continuous	1D		
Scatter Plot	Check the correlation between two attributes and outlier detection	Quantitative continuous	2D		
Contour Plot	Check the correlation of an attribute as a function of other two	Quantitative discrete and continuous	3D		

4.2 Icon-based Techniques

Iconographic techniques provide a mean to verify rules and behavior patterns of data. Each record can be viewed through graphical representations, whose attributes are mapped to the characteristics of an icon. Representations with similar properties can be recognized and thus form groups, that may be analyzed in particular. Icons with discrepant format compared to others in the chart may characterize outlier.

As the values of data attributes affect the icon format (e. g., size, position), it is more common to find visualization techniques to represent quantitative attributes (Keim 2000) . However, it is possible to represent qualitative data with such techniques. It's just necessary to carry out a process of transforming categorical data into numerical values (Ward 2008).

Regarding the representation of data volume, an iconographic technique generally can hold small volume of data (Oliveira & Levkowitz 2003), because each icon corresponds to a record, which limits the display of large amounts of data (Rabelo et al. 2008). In this case, ways of handling charts, such as zooming and brushing, can be used to obtain more accurate view when data volume is large.

The icon of a particular visualization technique has a fixed structure whose properties correspond to data attributes. Thus, generally the variation of the attributes positions is not a relevant aspect for this technique. Although the Stick Figures chart is an example in which the position of the attributes may influence data analysis because different families of icons can be derived from the varying distribution of data attributes into icon properties (Pickett & Grinstein 1988, p 516).

Therefore, the use of iconographic techniques is recommended for analysis of quantitative data with small to medium volume, with more than three dimensions, when conventional techniques of data representation 1D to 3D do not fully allow the exploration and patterns discovery on that data. Table 4 shows the main features of the iconographic techniques discussed above.

Table 4. Analysis of parameters for iconographic techniques

Category		Concepts			
Iconographic	Task	Data type	Dimension	Volume	Position
Chernoff Faces	Detection of clusters and outliers, identify behavior patterns (Bruckner 1978)	Quantitative discrete and continuous	Up to 18 attributes (Chernoff 1973)	Small	It does not influence
Star Glyphs	Detection of clusters and outliers		Up to 80 attributes (Rabelo et al. 2008)	Small	It does not influence
Stick Figure	Cluster detection in the data depends on the textures type formed by different icons		5 to 15 attributes (Keim 2005)	Medium	It influences

4.3 Geometrically Transformed Displays

That category of techniques may be used to identify patterns, rules, behavior, and also be used to analyze outliers in quantitative or qualitative data. The analyst can choose to view a dataset that may be detached from the tool in use, but clusters are not immediately identified by the techniques of this category. Furthermore, that techniques allow an overview of the data to assess the correlation between attributes.

An extremely large number of dimensions may affect the visualization, such as for techniques Scatter Plot Matrix, and Parallel Coordinates. In the case of Scatter Plot Matrix, the limitation of the dimensionality is due to the space constraint of the screen to display all combinations of attributes. As well as the Parallel Coordinates, the representation of many attributes makes the vertical axes getting too close, making it difficult to detect patterns.

That category of techniques may be applied on a medium data volume (Inselberg 2008). As the data volume is larger, the possibility of overlapping elements representing each record increases. It makes difficult to detect clusters and patterns. In the case of Parallel Coordinates, a solution is the use of saturation (Theus 2008), or colors (Rabelo et al. 2008) to highlight different groups, which are also applicable to the Scatter Plot Matrix.

According to (Inselberg 2008), the axes positioning of parallel coordinates may influence data analysis. On the other hand, in the Scatter Plot Matrix, the positioning of attributes does not cause too much impact on their interpretation, as it can be composed of scatter plots. As seen in the section about 1D to 3D charts, it is not greatly influenced by that parameter. Table 5 summarizes the main characteristics of this category of technique.

Table 5. Analysis of the parameters for geometrical techniques

Category		Concepts			
Geometric	Task	Data type	Dimension	Volume	Position
Scatter Plot Matrix	Check correlation among attributes and clusters detection	Quantitative continuous	Around 15 (Rabelo et al. 2008)	Medium	It does not influence
Parallel Coordinates	Overview on data, identification of clusters, outliers and correlation among attributes	Quantitative and qualitative	Hundreds of attributes (Inselberg 2008)		It influences

4.4 Pixel-oriented Techniques

Detection and identification of patterns in data are the main tasks that may be performed by this category of techniques. According to Keim (2001), the pixel-oriented techniques can be used to visualize quantitative data with high dimensional and stand out due to represent large amounts of data. In addition, the overlap of elements on the display does not influence the visual analysis, being this a distinguishing aspect from geometric techniques, but they are not recommended for viewing qualitative data (Keim & Kriegel 1996).

In pixel-oriented techniques, the deploy of attributes in the chart affects the correlation analysis of dimensions of data. Thus, to verify the relationship between attributes, it should arrange them near each other on the display.

A particular characteristic of pixel-oriented techniques is the algorithm to positioning the pixels on the display. It may interfere in the interpretation of results. This is the main factor that differ pixel-oriented techniques from the other techniques previously discussed in this paper. According to Keim (2000), pixels may be arranged according to recursive pattern algorithm when it uses the query-independent technique. The organization of pixels when using query-dependent techniques can be performed by the circle segments algorithm. Table 6 summarizes the visualization techniques of the pixel-oriented category discussed.

Table 6. Analysis of the parameters for pixel-oriented techniques

Category		Concepts			
Pixel-oriented	Task	Data type	Dimension	Volume	Position
Query-independent technique	Detect patterns and correlations among attributes	Quantitative (e. g. temporal data)	10 to 100 attributes (Keim 2005)	Large	It influences
Query-dependent technique	Check relationship among attributes conforming a query	Quantitative (the attributes must be in the context of the query)			

4.5 Hierarchical or Graph-Based Techniques

In general, both hierarchical or graph-based techniques are employed to visualize relationship structure of data items disposed in a hierarchy or network. In addition, this category has techniques to represent qualitative data as, e.g., the Mosaic Plot, which has some features of the Treemap technique, but with less resources, such as proportional areas conforming the value of attribute, construction of the display based on a classification inherent to the data, and format dependent on the order of attributes (Hofmann 2008). According to Oliveira and Levkowitz (2003), Dimension Stacking technique can be applied to data stored in tables, but not necessarily organized in a hierarchical format like it is assumed by the Cone Trees, and Treemaps techniques. Techniques of this category may represent high dimensional data, since the attributes can be represented by the nodes of a graph.

Regarding the amount of data, these techniques are appropriate for a volume ranging from small to medium (Oliveira & Levkowitz 2003), as can be seen, for example, in (Shneiderman 2006).

Hierarchical, or graph-based techniques are often influenced by the position of the attributes, because the elements follow naturally a structure of relationship. Therefore, it is needed to be careful to locate the attributes in the graph, especially in hierarchic levels. Table 7 shows the hierarchical techniques discussed.

Table 7. Analysis of the parameters for hierarchical or based-graph techniques

Category		Concepts			
Graph-based	Task	Data type	Dimension	Volume	Position
Graph	Overview on the relationship structure among elements	Data with relational structure (hierarchical or network)	High	Small to medium	It influences
Cone Tree	Facilitate navigation in data with a tree structure	Suitable for viewing structures of files and directories	High (may exhibit around 1000 itens)	Small to medim	It does not influence
Treemap	View clustering data	Data with some type of hierarchy or taxonomy	High (may exhibit 2 to 20 clusters)	Large	It influences
Mosaic Plot	Overview on qualitative attributes	Qualitative nominal and ordinal	Medium to high	Small	It influences
Dimensional Stacking	Detect patterns, clusters and outliers	Quantitative discrete and continuous	Medium	Medium	It influences

5. CONCLUSION

This paper describes the parameters to be considered when planning to use visualization techniques in the process of extracting knowledge from data. That parameters are identified through Grounded Theory methodology, which is basically composed by theoretical sampling and coding process.

The theoretical sampling represents a set of data and information built by the data collection task. As soon data have been obtained, they have undergone the coding process in which was used a strategy of encoding by key points. That method allowed emergence of the concepts that led to the parameters: data type, task type, volume, dimensionality and positioning of the attributes in the graph.

Then each parameter was analyzed in conjunction with visualization techniques, and those most frequently found in the literature were selected, and separated by categories defined in the Keim's taxonomy.

Through analysis of this relationship, it was observed that each technique type (1D to 3D standard graphics, icon-based displays, geometrically transformed displays, pixel-oriented displays and graph-based or hierarchical displays) have a certain configuration of parameters that reflect the data characteristics and the objectives of its use.

Besides the parameters, another point to be considered is the meaning of the data analyzed by the analyst. As the analyst's familiarity about data increases, greater is his/her incentive to make data exploration to get new hypotheses to be verified by visualizations, or just comparing the results generated by various graphical representations.

ACKNOWLEDGEMENT

This work was supported by Fundação Araucária.

REFERENCES

- Allan, G., 2003. A critique of using grounded theory as a research method. *Electronic Journal of Business Research Methods*, Vol. 2, No.03. Available from: <<http://www.ejbrm.com/>>
- Ankerst, M., 2001. Visual Data Mining with Pixel-oriented Visualization Techniques. In *ACM SIGKDD Workshop on Visual Data Mining*. San Francisco, CA.
- Chernoff, H., 1973. The Use of Faces to Represent Points in K-Dimensional Space Graphically. *Journal of the American Statistical Association*, v.68, n. 342, p.361-368.
- Coleman, G. and O'Connor, R., 2007. Using grounded theory to understand software process improvement: A study of Irish software product companies. *Elsevier – Information and Software Technology*, v. 49, p. 654–667.

- Dick, B., 2005. *Grounded theory: A thumbnail sketch*. [online] Available at: <<http://www.scu.edu.au/schools/gcm/ar/arp/grounded.html>>. [Accessed January 2010].
- Freitas, C. M. D. S., et al, 2001. Introdução à Visualização de Informações. *Revista de Informática Teórica e Aplicada (RITA)*, Vol. 08, No. 02.
- Grinstein, G., et al, 2001. High-Dimensional Visualization. In: *Proceedings of the Visual Data Mining Workshop (KDD'01)*.
- Herman, I.; et al., 2000. Graph Visualization and Navigation in Information Visualization: A Survey. In *IEEE Transactions on Visualization and Computer Graphics*, v. 06, n.1, p. 24-43.
- Hofmann, H, 2008. 'Mosaic Plots and Their Variants' in eds Chen, C., et al., *Handbook of Data Visualization*, Springer, pp.618 - 642.
- Inselberg, A, 2008. Parallel Coordinates: Visualization, Exploration and Classification of High-Dimensional Data' in eds Chen, C.; et al., *Handbook of Data Visualization*, Springer, pp. 643-680.
- Keim, D. and Kriegel, H., 1996. Visualization Techniques for Mining Large Databases: A Comparison. *IEEE Transactions on knowledge and data engineering*, Vol. 08, No. 06, pp. 923-938.
- Keim, D., 2000. Designing Pixel-Oriented Visualization Techniques: Theory and Applications. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 06, No. 01, pp. 1-20.
- Keim, D., 2001. Visual Exploration of Large Datasets. *Communications of the ACM*, Vol. 44, No. 08, pp. 39-44.
- Keim, D., 2002 Information Visualization and Visual Data Mining. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 07, No. 01, pp.100-107.
- Keim, D., 2005. Scaling Visual Analytics to Very Large Data Sets. In *Workshop on Visual Analytics*, Darmstadt.
- Klippel, A., et al, 2009. Star Plots: How Shapes Characteristics Influence Classification Tasks. *Cartography and Geographic Information Science*.
- Lee, M. D.; et al., 2003. An empirical evaluation of Chernoff faces, star glyphs, and spatial visualizations for binary data. In: *Proceedings of the Asia-Pacific Symposium on information Visualisation*, Australia, v. 24.
- Matavire, R. and Brown, I., 2008. Investigating the use of "Grounded Theory" in information systems research. In: *SAICSIT '08: Proceedings of the 2008 annual research conference of the South African Institute of Computer Scientists and Information Technologists on IT research in developing countries*, Africa do Sul, p. 139-147.
- Morgenthaler, S., 2009. Exploratory Data Analysis. *Wiley Interdisciplinary Reviews: Computational Statistics*, v.1, n.1, p.33-44. [online] Available at: <<http://www3.interscience.wiley.com/journal/122511655/abstract>>. [Accessed 24 July 2009].
- Myatt, G. J. 2007. *Making Sense of Data - A Practical Guide to Exploratory Data Analysis and Data Mining*, John Wiley & Sons, Inc..
- Oliveira, M. C. F. and Levkowitz, H., 2003. From Visual Data Exploration to Visual Data Mining: A Survey. *IEEE Transactions on Visualization and Computer Graphics*, Vol. 09, No. 03, pp. 378-394.
- Pickett, R. M. and Grinstein, G. G., 1988. Iconographic Displays for Visualizing Multidimensional Data. In *Proc. IEEE Conference on Systems, Man and Cybernetics, IEEE Press*, Piscataway, NJ, p. 514-519.
- Pillat, R. M., et al, 2005. Experimental Study on Evaluation of Multidimensional Information Visualization Techniques. In *Proceedings of the 2005 Latin American conference on Human-computer interaction*, pp. 20-30.
- Rabelo, E., et al, 2008. Information Visualization Which the most Appropriate Technique to Represent Data Mining Results? In *Proceedings of the International Conference on Computational Intelligence for Modelling, Control and Automation*, Viena – Austria, pp. 1218-1223.
- Rodrigues, A., et al, 2004. *Grounded Theory: Problemas de "Alicerçagem"*. Technical Report. Universidade de Coimbra, Portugal.
- Shneiderman, B., 1996. The Eyes Have It: A Task by Data Type Taxonomy for Information Visualizations. In *Proceedings of IEEE Symposium on Visual Languages*. Boulder, CO, p. 336- 343.
- Shneiderman, B., 2006. *Discovering Business Intelligence Using Treemap Visualizations*. Technical Report. Human Computer Interaction Lab – HCIL, University of Maryland.
- Theus, M., 2008. 'High-dimensional Data Visualization' in eds Chen, C., et al., *Handbook of Data Visualization*, Springer, pp. 151- 178.
- Ward, M. O., 2008. 'Multivariate Data Glyphs: Principles and Practice' in eds Chen, C., et al., *Handbook of Data Visualization*. Springer, pp. 179-198.
- Wyatt, R., 2008. Face charts: a better method for visualizing complicated data. In *Proceedings of the IADIS International Conference on Computer Graphics and Visualization*, Amsterdam, p. 51-59.

Short Papers

INDICATION OF METHANE GAS IN IR-IMAGERY

Julia Åhlén and Stefan Seipel

*University of Gävle, Department of Industrial Development, IT and Land Management
80176 Gävle, Sweden*

ABSTRACT

There are human produced sources of methane gas, such as waste storages, that contribute to the global warmth and other negative effects. There is not much research on the correlation of such leakage and greenhouse effect. Methane gas is not visible for humans and thus impossible to detect using commercial cameras. Specially designed IR-camera can detect this gas and thus is used in this study. Using digital video taken over a waste disposal place we create a detection algorithm that is sensitive to the spectral and morphological characteristics of methane gas. Different kind of leakage can take place in waste disposal places. In case of small spot leakage there is a reason to assume failure in piping system and in case of widely spread leakage area we can state that it is caused by unsupervised storage of waste and this should be attended immediately. In digital video, background and target gas are distinguished using spectral and morphological classifiers, which are extracted from the analyzed IR-imagery. It is shown that indications of methane gas can be carried out efficiently using image processing techniques and the definition of turbulence of the image.

KEYWORDS

Methane, IR, detection, video.

1. INTRODUCTION

Methane (CH₄) is a greenhouse gas that remains in the atmosphere for approximately 12 years. It is established that over a 100-year period, methane is over 20 times more effective in trapping heat in the atmosphere than carbon dioxide (CO₂) (Wuebbles, 2002). It is emitted from a variety of natural and human-influenced sources, which include landfills, natural gas systems, agricultural activities, coal mining, stationary and mobile combustion, wastewater treatment, and certain industrial process (IPCC, 1996). Methane emissions from open dumping and improper land filling of municipal solid waste (MSW) contribute to 3–19% of the anthropogenic sources in the world (Talyan, V, et al., 2007). The reduction of CH₄ emissions from landfills would make a significant contribution to the curtailment of greenhouse gas stock. The generation of landfill gases is found to occur in five sequential phases (S.Kumar et al., 2004) thus detection of methane is one of the most important chemical measurements with particular relevance to environmental applications. Methods for safe, efficient and economical methane gas detection are required by waste disposal industry, which is a major source of methane leaks. There is a huge amount of different approaches to detect and measure different properties of methane gas.

Pellistor based detection systems were used for decades, where palladium catalyst is incorporated to accelerate methane gas ignition and measure the consequent temperature rise (Jones, E., 1987). The limitations are in susceptibility to surface poisoning. The alternative technique is to measure the spectroscopic absorption of a gas sample and a few systems have been realized to date based upon this principle. Typically, these systems operate in the mid infra-red range (3–5 μm wavelength) and use relatively broadband optical filters to select the appropriate range of wavelengths. The detectors, since they operate at relatively long optical wavelengths must be cooled in order to achieve reasonable signal to noise ratio (Aggarwal, 1991). The methane gas absorption line at between 3 and 3.3 μm is relatively strong and so despite the poor noise performance of the detection system, reasonable sensitivities can be obtained (Culshaw, B, 1998). Finding methane leaks using this method is labor intensive, time consuming and surveys must be conducted point by point in close proximity with leaks. It is clear that more efficient methods for methane gas parameter acquisition are required.

1.1 Use of IR-Imaging Technique For Methane Gas Control

Optical imaging techniques must meet several requirements to qualify as a smart approach. According to (Epperson, et al., 2007) they should be able to significantly reduce monitoring cost, locate leak without having to monitor every individual piping component, provide better control effectiveness and give better environmental protection.

Despite of obvious benefits of IR-camera imaging as a methane detection technique, there are drawbacks, such as inability to measure the concentration, or the leak rate of the gas. There are no methodologies that rely only on IR-imagery. In this work, carried out using FLIR IR-camera, we propose a method suitable to process and analyze IR-imagery in order to characterize the methane gas with spectral descriptors and morphology. This will allow for computing features useful to describe the gas flow dynamics. The approach is built on visualization of gas against the background. Detection can occur only if the target signature can be distinguished from the background. The radiation intensity of the target object and background depends on their temperatures, however, using FLIR camera allows for thermal imaging of the gas leaks with high precision. It is possible to visualize the gas against the background only if the background and the target gas cloud are not in thermal equilibrium. If the gas temperature is uniform with the background temperature, the gas cannot be discriminated against the background. In this application, the infrared camera can visualize gas, which has higher temperature from the surrounding. By (Ljungberg, S.Å. et al. 2009) it is established that in order to detect methane gas leaks the temperature difference between the surroundings and methane gas should be more than 2°C.

Benson et al. showed in ((Benson et al., 2006) that methane gas concentration as low as 275 parts per million (ppm) could be distinguished by the infrared detector, however, the test was conducted in a laboratory where wind force is negligible. In this study the concentration of 1000 ppm and more is necessary for the methane to be visible in IR-imagery. In this work we will investigate outdoor waste disposal scene and therefore gas may spread in an unpredictable manner. A dispersed gas can be recognized by the infrared camera to a certain threshold concentration. Generalized Gaussian dispersion model is proposed in (Safitri, A. et al., 2011) to estimate the flow rate and concentration of methane gas leaks. However, weather conditions could not be taken into account by this mode. In addition the drawback of this model is that the calculation of cloud size is limited to a passive state of the gas, the model will not consider the transient condition at the time the gas has been discharged

In (Åhlén, Seipel, 2010) the method for smoke detection using primitive features combined with turbulence estimation is proposed. Smoke exhibits the same flow characteristics in the air and morphological properties as methane gas in IR-images, however, transparency levels are much higher for the IR-imagery and thus we can suggest only partial usage of their model in our work. Varying background is another deferens to the smoke detection scenario.

In this article we suggest a method for IR-video analysis where optical flow (Horn and Schunck, 1981) combined with image turbulence indicates presence of methane gas and thus can be modeled for each dispersion stage. The dispersion caused by winds is making the process highly complicated. We argue that in case of IR-imagery on methane gas it is less successful to use image primitives in order to detect the spreading of gas. Quantitative analysis is not done in this work since it requires ground truth with the equipment that allows to measure concentration of gas in site. We do not take into account the situation where the sun introduces turbulence due to heating of cold air over a dark ground, however, this will occur every morning during summer time.

2. METHANE GAS DETECTION IN IR-IMAGERY

Each molecule has a specific absorption range within the infrared spectrum. For methane gas strong absorption rate lies in interval of 3.2-3.4 μm . FLIR infrared camera used in this study visualizes gas by inverting the physics of fugitive gas leaks. We suggest a method of methane gas presence estimation in FLIR-IR imagery where the outdoor environment varies in a big extent. In some imagery, where background is a snow, methane gas temperature is warmer than the surroundings. In the other video, where methane gas is leaking from pipes there is a vice versa relation of methane to background: the gas temperature is colder than backgrounds. In some imagery there are other features, such as people and vegetation, than only

methane and background. In those cases we admit the hopelessness of orthodox image processing approach, where edge detection and intensity value analysis usually is applicable. We investigate if the background can be efficiently separated from gas features and subject those features for turbulence calculation.

Our method is based on optical flow combined with image turbulence analysis. As a first step we read in video into the system and cut out the part with FLIR-camera settings, which is disturbing in optical flow calculation.

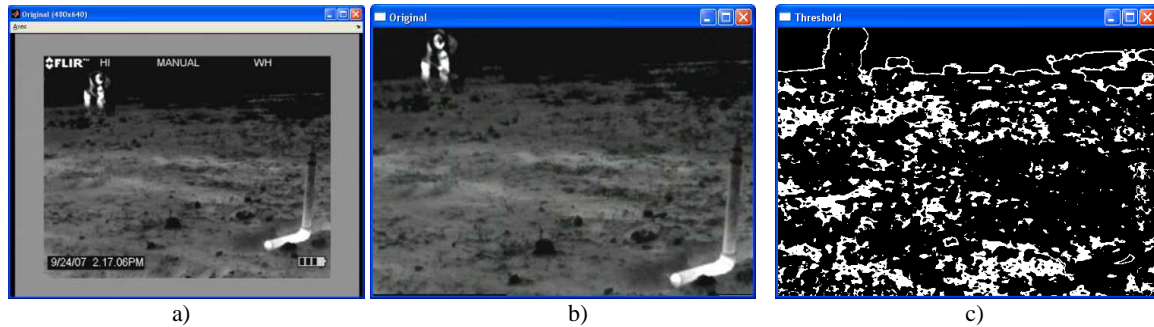


Figure 1. a) Original video frame b) cropped area c) After Optical Flow algorithm

The next step is to de-noise with median filter and then continue with calculation of optical flow vectors using Horn-Schunk algorithm. We choose median because of the smooth nature of methane gas presence in the image. The fact is that there is not much edge information and thus using linear smoothing filters would cause extra disappearance of edges. As the complexity of the imagery is high and methane gas dispersion is severe we get a cluttered result from this step, thus we need to perform thresholding of the resulting imagery. We search after specific values of threshold that aims to give gas movements in a binary form, see Figure 1.

Only by viewing video we can see that the gas is spread out over the whole imagery. Methane gas is visible only against background and only if temperatures are different. The bigger distance there is from the source the higher the transparency. However, the algorithm determines movements by comparison of two adjacent frames and thus we are able to detect even hardly visible movements, such as of semi-transparent gas. After this step we estimate parameters for the input of turbulence calculation. We perform morphologic closing. The result is the much clearer binary image of the gas because the broken features are eliminated. Now it is time to subject the image to statistics calculation, such as finding bounding boxes and areas of the detected gas clouds. We need this operation so that we can calculate the rate of change of the turbulence. This quality can be detected by monitoring positive local extrema of the turbulence. Rate of change of turbulence in the image is calculated using Equation 1, described in (Åhlén, Seipel, 2010).

$$\frac{\partial \Omega}{\partial t} = \frac{2P'(t)A(t) - P(t)A'(t)}{4\sqrt{\pi A(t)^{3/2}}} \quad (1)$$

Where P represents the perimeter/edges of the remaining moving gray features and A represents the area of the region. As the last step we exclude values that do not correspond to the criteria of gas turbulence. We

define such criteria by analyzing the rate $\frac{\partial \Omega}{\partial t}$. In each frame we use cumulative sum of the values that are suitable for this application. Values are much lower than in the application described by (Åhlén, Seipel, 2010). In this application we look at the span of 0-500 values as opposed to 0-2700 in previous work. As a result we find positions in the image where the methane gas clouds are found by the described algorithm and display these positions in the original video. In this case we have tested high Threshold value on Optical Flow against low threshold interval of the estimated turbulence.

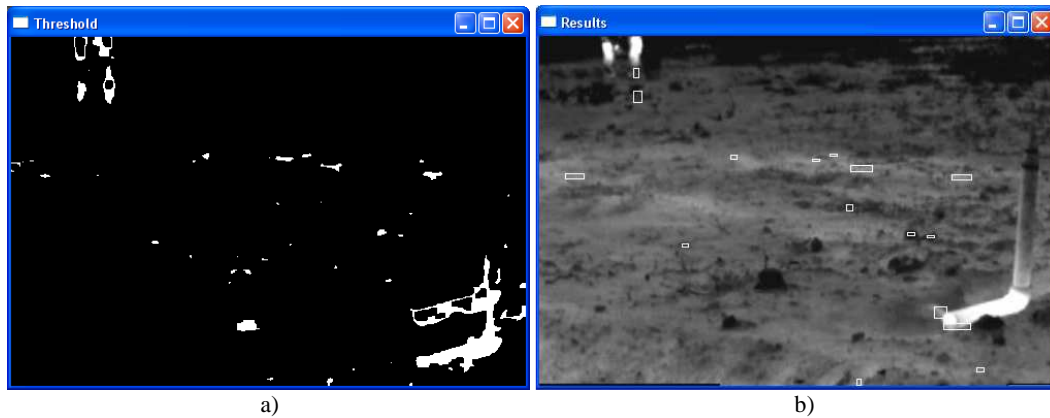


Figure 2. a) Found positions, in a frame, based only on optical flow, with thresholding, b) image frame, where gas rate of change was the highest and positions of such places.

We choose to display the result in two different ways to make it possible to visually explore the usefulness of turbulence processing. When we are comparing those two images we see that in the case of optical flow and thresholding, shown in Figure 2 a), we detect positions, which not exclusively correspond to the methane gas. This is done by taking measurements of flow vectors and keeping only those with the appropriate angle. After the step of turbulence estimation we could reduce the detected positions to those corresponding only to flow directions of methane gas, see Figure 2 b). In the turbulence estimation our algorithm performance depends on the correct choice of threshold for the cumulative sum of the found values of rates. This threshold can be automatically set by using the values from minimum and maximum rate of change in each frame. By empirical tests we have found optimal combination of thresholds that can extract two different types of gas leaks. The one is with low threshold on optical flow and low on turbulence result. This combination detects gas that exhibits low dispersion and thus indicates leaks from big source areas. The opposite combination of values, where the optical flow threshold and turbulence thresholds are high, predicts the point source caused by broken pipe.

3. RESULTS

Different combinations of thresholds that give satisfactory results are presented in Table 1:

Table 1. Threshold values and the amount of found gas clouds in video

Optical Flow Threshold <i>OF</i>	Threshold for Turbulence <i>T</i>	Amount of Gas Clouds
$OF < 0.00000072$	$18 < T < 24$	17
$OF < 0.00000082$	$18 < T < 24$	10
$OF < 0.000000003$	$18 < T < 24$	11
$OF < 0.00000072$	$11 < T < 17$	12
$OF < 0.00000082$	$11 < T < 17$	5
$OF < 0.000000003$	$11 < T < 17$	5
$OF < 0.00000072$	$5 < T < 11$	7
$OF < 0.00000082$	$5 < T < 11$	5
$OF < 0.000000003$	$5 < T < 11$	2

We choose to test turbulence threshold T against optical flow threshold OF . As we can see from the table there is clear indication on dependency of those values and the amount of found gas features in video. The lower the T the fewer amounts of methane gas clouds are segmented. However, due to big difference in outdoor conditions it is not justified to create more thresholds on Optical Flow calculation. In case of low T -value and low OF value, while not searching for the low dispersion, we could not establish meaningful results.

4. CONCLUSION

We present a method for robust methane gas detection from IR-video taken with FLIR camera. This algorithm use similar approach of many smoke detection algorithms. There are several differences though. In most of cases in waste deposits there is no need to stress the speed of the detection. The importance lies in the correct position indication, since it is not possible to see or smell methane gas. IR-imagery provides for the mid-infrared band, which is particularly sensitive for methane gas. Leaks can be caused by broken pipes or by wrongly placed landfill. In both cases repair work should be executed. It is important to get knowledge on the source of the leak and the type of it. We have shown in this work that it is possible to do that by analyzing the dispersion of gas. In our method we analyze the incoming video in the sense of presence of gas and the type of the leak. If turbulence encounters high values of the rate of change we are looking at the point source, which is caused by pipe brake. This is du the fact that the dispersion of gas outdoor is quite high and thus differs significantly from the initial value. We can detect places with low turbulence, which will point to the massive leak of methane gas. Using only flow direction calculation we can only show the presence of methane gas in the IR-imagery. With added turbulence calculation we narrow the search to some specific type of the leak. Our approach worked in the case of 7 tested video with different backgrounds and varying amount of features on them. The algorithm could detect two types of leaks and possible source positions on all of the tested videos. We can argue that installing IR-video camera at potential places of methane gas can safely and efficiently detect and analyze gas leaks.

REFERENCES

- Aggarwal, I.D., Lu, G., 1991. *Fluoride Glass Fibre Optics*, Academic Press, London.
- Benson, R. et al. (2006). Standoff passive optical leak detection of volatile organic compounds using cooled In Sb based infrared imager. *Proceedings of Air and waste management association (AWMA) 99th annual conference and exhibition*, New Orleans, USA, pp. 131-139.
- Culshaw, B. et al. 1998. Fibre optic techniques for remote spectroscopic methane detection-from concept to system realization. *In Sensors and Actuators B: Chemical*, Vol. 51, Issues 1-3, 31, pp. 25-37.
- Epperson, D. et al. 2007. Equivalent leak definitions for smart LDAR (Leak detection and repair) when using optical imaging technology, *In Air and Waste Management Association*. Vol. 57, Issue 9, pp. 1050-60.
- Horn, B.K.P., and Schunck, B.G., 1981. Determining Optical Flow, *In AI Vol.17*, No. 1-3, pp. 185-203.
- IPCC, 1996. Intergovernmental Panel on Climate Change (IPCC). *Revised IPCC guidelines for national greenhouse gas inventories*. Workbook, 6:1-22. Available at: <http://www.ipcc-nggip.iges.or.jp/public/gl/guidelin/ch6ref1.pdf>.
- Jones, E., 1987. *The pellistor catalytic gas detector*. *Solid State Gas Sensors*. P.T. Mosley and B.C. Tofield Editors, Adam Hilger, Bristol.
- Kumar, S. et al. 2004. Qualitative assessment of methane emission inventory from municipal solid waste disposal sites: a case study. *In Atmospheric Environment*, Vol. 38, Issue 29, pp. 4921-4929.
- Ljungberg, S.Å. et.al. 2009. Detection and quantification of methane gas leaks from waste management places. Swedish Gas Centre. Report SGC 203.
- Never, N. , 1995, *Air Pollution Control Engineering*. McGraw-HILL, Singapore.
- Safitri, A. et al. 2011. Dispersion modeling approach for quantification of methane emission rates from natural gas fugitive leaks detected by infrared imaging technique. *In Loss Prevention in the Process Industries*, Vol. 24, Issue 2, pp.138-145
- Talyan, V. et al. 2007. Quantification of methane emission from municipal solid waste disposal in Delhi. *In Resources, Conservation and Recycling*, Vol. 50, Issue 3, pp. 240-259
- Wuebbles, D., J. et al. 2002. Atmospheric methane and global change. *In Earth-Science Reviews*, Vol. 57, pp. 177-210.
- Åhlén, J, Seipel, S., 2010, Early Recognition of Smoke in Digital Video, *Proceedings of European Conference of Computer Science*, Puerto De la Cruz, Spain, pp. 301-306.

MICRO MACRO IMAGE SEGMENTATION

Asmar A. Khan, Costas Xydeas and Hassan Ahmed

School of Computing and Communications, Lancaster University, LA1 4WA, Lancaster, UK

ABSTRACT

The MMIS image segmentation (IS) approach that is presented in this paper is novel and can be used to enhance the performance of currently existing micro-block based image segmentation techniques. The proposed scheme has been designed to automatically derive the appropriate number of regions so that object oriented and semantically important image content is preserved. Experimental performance results based on both subjective and objective tests clearly show the superiority of MMIS as compared to a state of the art IS.

KEYWORDS

Image analysis, Segmentation, Clustering.

1. INTRODUCTION

Considerable research effort has been directed over the years in the area of Image Segmentation (IS), as this process is a key element of image analysis / vision systems. However, the goal of achieving an “ideal”, object oriented and visually meaningful segmentation performance over a wide spectrum of image content is still a challenge.

Conventional segmentation processes produce a number of image segments, with pixels being grouped together on the basis of their intensity/color value and / or edge related characteristics. Furthermore IS schemes usually operate directly on “micro-image” building blocks in general and on pixels in particular. Although the direct use of such a high level of granularity in the input data aims at making available to the segmentation process fine-detail image information, it also has the potential to add a significant amount of “localized” uncertainty. This is similar to the situation where one finds it hard to read a newspaper that is placed too close to his eyes, or to the situation of a maze where one cannot see beyond nearby obstacles and hence is unable to easily find a way out. In contrast, if a “bird’s eye”, macro-view of the maze was to be made available, the problem becomes much simpler as “global” information, that is present beyond these nearby obstacles, becomes evident.

Segmentation techniques may also mark segments (regions) with a label or class. Hence the segmentation process results in a number R of regions and each region may be associated to one of M classes. Furthermore most existing segmentation techniques [Cour] require the user to define a-priori the number of regions and or regions/classes to be produced.

The image segmentation approach that is presented in this paper is novel and should be viewed not just as another segmentation algorithm but as a methodology that can be used to enhance the performance of currently existing micro- block based image segmentation techniques. The idea here is to apply to the output of such a conventional IS scheme, a process which examines the already segmented image “macroscopically”. Thus, rather than relying only on information related to individual pixels and their localized neighborhoods, the proposed IS approach also pays attention, within a Micro-Macro combined Image Segmentation (MMIS) framework, to information obtained from larger image-regions. As a result, higher level, object oriented image content becomes evidently more apparent in the resulting MMIS segmented image.

Accurate object oriented segmentation has been the main driver behind the development of MMIS and, in such a system, the ability to automatically estimate the right number of image segments is of paramount importance. Over-segmentation has the tendency to damage the visual appearance of content-critical object information, whereas under-segmentation tends to wrongly join together regions representing semantically

different objects. Thus the paper not only offers a methodology that, for a given number of segments, improves the performance of conventional segmentation techniques, but also provides a novel solution to the problem of automatically calculating the appropriate number of segmented regions.

The MMIS system performance has been investigated, using computer simulations, by employing i) informal, subjective preference comparisons of segmented images produced by different schemes, and ii) objective metrics. Objective results are derived using IS ground truth data that is obtained from a large number of manually segmented images [Martin]. Experimental results based on both subjective and objective tests clearly show the superiority of MMIS as compared to a state of the art, IS scheme [Cour].

2. IMAGE SEGMENTATION RELATED WORK

A large number of IS approaches have been developed over the years for a variety of image processing applications. These may be conveniently categorized as Region based [Arbelaez], Edge based and Object-Oriented based approaches [Unnikrishnan]. Recently two major types of IS implementation techniques have been extensively used on natural input image datasets [Martin]. One type is employing graph theoretic techniques [Zahn, Felzenszwalb] whereas the other follows the region merging paradigm using spectral methods [Cour, Shi].

In graph based techniques an input image is considered to be a graph with pixels as nodes and pixel differences (normally intensity based) used as weights assigned to edges. Early graph based segmentation and grouping work was inspired by Zahn's Minimum Spanning Tree scheme [Zahn] and resulting systems were of high complexity [Cour, Shi, Wu]. Since then work on graph based IS techniques has progressed considerably and recent contributions [Felzenszwalb], claim linear system complexity characteristics. However, their performance is dependent upon a number of predefined system parameters. These are normally image specific and cannot be easily predicted prior to the segmentation process. Hence when applying such IS schemes in image analysis and understanding applications involving unconstrained input image data, these system parameters can only be derived experimentally via training/learning.

Other graph based techniques take pixels as nodes and using a certain threshold, allocate pixels as belonging to a region. Although such approaches exhibit a relatively low level of complexity and may perform reasonably well on certain types of images, they do not provide segment classification and cannot always achieve high levels of performance.

More recently a number of segmentation techniques have been proposed which employ elements of spectral theory. The concept of spectral driven IS revolves around the eigenvector-based partitioning of connected components [Cour, Shi]. Nevertheless such IS systems are computationally complex.

Finally, a segmentation scheme that attempts to cluster randomly selected "macro" related and equal size regions has been reported in [Ren]. However this scheme does not benefit from the important micro type of information that is extensively used in other conventional IS systems.

3. MMIS SYSTEM DESIGN

Consider that an input image is presented to a conventional segmentation system which produces at its output R regions. Furthermore each region may have been also classified to one of M classes. MMIS accepts the segmented image and commences with the assumption that $M=R$.

The R input regions are most likely to have being formed using micro-image related information and MMIS, see Figure 1, proceeds by considering this segmentation result under the light of macro-image type of information. Thus the system extracts certain **Region based image Information (RI)** and then enters into a recursive **Region Clustering (RC)** mode. RC, as applied here, is a process performed on the initial R regions in a sequence of processing steps (iterations), with the number of regions reduced by one at each step. At the end of each step a clustering termination decision is made which, when positive, allows the MMIS system to provide its final segmentation output of the original input image.

3.1 Region Clustering

The segmented input image is used to form a graph whose nodes are the R image regions and whose edges define a “similarity” type of distance $d(p,q)$ estimated between regions p and q. Moreover the graph is fully connected with every region having an edge connecting to every other region, giving a total of $R(R-1)/2$ edges. In terms of complexity, this graph formulation is considerably simpler than the pixel based graphs often used in traditional IS schemes, since $R \ll K \times L$ with K, L being the image dimensions.

Given this fully connected graph and $d(p,q)$, a number of region clustering methods can be used which are based ,for example, on a minimum spanning tree search or the Min-Cut max-flow theorem [Wu]. Lately the concept of graph Normalized Cut has been introduced by Shi and Malik [Shi]. This method is particularly attractive since clustering results representing local minima are avoided. Moreover its high complexity can be eased via spectral approximations [Cour, Arbelaez, Shi].

The MMIS formulation used in this paper, employs a hierarchical, agglomerative, clustering approach to form clusters using the $d(p,q)$ distance. This is an iterative process where the number of clusters is reduced by one every iteration. Thus in the beginning, each region is a cluster and based on $d(p,q)$ values, regions are clustered together sequentially to form fewer clusters. The cluster formed at iteration ‘i+1’ is made of two separate iteration ‘i’ clusters.

Of course key to this hierarchical, iterative clustering process is the way the distance d is formed between clusters. The issue of calculating d has been explained in [Theodoridis] whereas the way MMIS determines d can be conveniently highlighted in the following simple example.

Consider two clusters C_a and C_b that exist at a certain hierarchy level (iteration) ‘t’. If C_a is the result of merging clusters C_i and C_j at the t-1 level , then $d(C_a,C_b)$ can be defined in general as

$$d(C_a, C_b) = wd(C_i, C_b) + xd(C_j, C_b) + yd(C_i, C_j) + z|d(C_i, C_b) - d(C_j, C_b)| \quad (1)$$

Choices of w, x, y and z values determine the specific implementation of agglomerative clustering where MMIS employs:

$$d(C_a, C_b) = \max\{d(C_i, C_b), d(C_j, C_b)\} \quad (2)$$

Thus regions are clustered until the desired level of hierarchical clustering is achieved. Furthermore MMIS terminates clustering automatically and in a way that relates to image content.

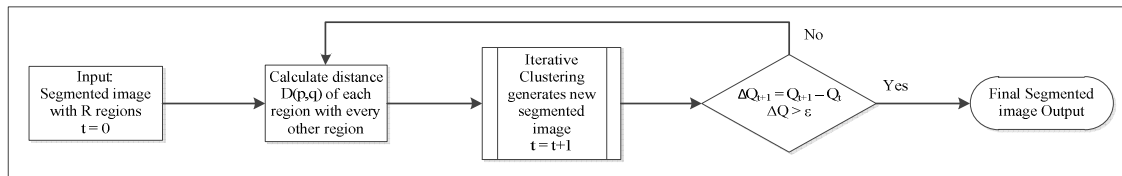


Figure 1. System level block diagram

3.2 Region Distance Metric

IS systems operating at pixel level employ predominantly measures based on i) intensity/ color or ii) edges / texture. Similarly when comparing regions, the majority of measures are either based on first order statistics e.g. mean or on second order statistics of transform related information e.g. Wavelets. Thus whilst the objective here is to merge regions which are visually similar and while region characterizations based on color, variance and spectrum are often appropriate, MMIS opted for the relatively simple but powerful region characterization provided by Probability Density Functions (PDF histograms). The information theoretic measure $d(p,q)$ used in MMIS is the Relative Entropy also known as the Kullback-Leibler Divergence (KLD):

$$D(p \parallel q) = \sum_{x \in X} [p(x) \log \{ \frac{p(x)}{q(x)} \}] \quad (3)$$

An information theoretic similarity comparison is therefore formed between two probability distributions using (3). Regions p and q should be large enough to avoid the occurrence of $p(x) > 0$ and $q(x) = 0$. Note

that the Relative Entropy (KLD) is not only simple, but it performed quite well in a recent comparison of information theoretical metrics used for clustering [Calderero]. Also note that a number of different region clustering metrics were also employed and tested within the MMIS framework and that they were all outperformed by KLD. This reinforces the argument that important macro-visual information, that is present in region related PDFs, is sometimes not adequately represented by first or second order region statistics.

Another important reason in favor of using KLD is that the formulation of other conventional metrics depends on several parameters which in turn are image dependent. The estimation of these parameters becomes cumbersome, when dealing in practice with unconstrained input images in object recognition and image understanding applications.

3.3 Termination of Clustering Process

The majority of image segmentation schemes operate according to a-priori given numbers of classes or regions in which case termination of the segmentation process is dictated by these numbers. Alternatively termination of segmentation can be based upon “learning” the appropriate number of regions via a training process. Thus learning techniques [Arbelaez, Unnikrishnan] have been proposed which involve the supervised evaluation of segmentation results by comparing outputs with manually obtained “masked segmentation” results [Martin]. Nevertheless these techniques are computationally expensive and often have limited success.

MMIS employs a distinctively different segmentation termination process that is based on a function Q evaluated at every step ‘ t ’ of the iterative clustering process. In particular

$$Q = \frac{\sum_{i=1}^k (d(x_i, x_j))^2}{M} \quad (4)$$

where for each ‘ i ’ the adjacent regions ‘ j ’ = 1,2,..., n and M is the no. of adjacent classes

The process relies on the fact that similar regions will have relatively small KLD value assigned to them. Now, as the number of regions is reduced sequentially via clustering, Q will vary. Let’s say at iteration ‘ t ’, $Q(t)$ is the normalized square sum of the KLD’s formed across adjacent regions. Let us also denote the value of Q at step ‘ $t+1$ ’ as $Q(t+1)$ and also $\Delta Q = Q(t+1) - Q(t)$. The ΔQ sequence of differential values calculated across clustering iterations, will assume relatively small values which are also close to the mean of the ΔQ sequence. This will be true as long as adjacent regions, which are assigned to the same cluster, are similar to each other. Clustering continues till the point where an important natural boundary in the image is removed; resulting to a significant change in ΔQ which in turn indicates that clustering should be terminated, see Figure 1. Thus MMIS compares the current ΔQ value to the ΔQ mean estimated over all previous iterations and clustering is terminated when the current value is significantly greater (say 25%) than the mean.

4. EXPERIMENTAL RESULTS AND CONCLUSION

In general a pre-specified large number of IS classes leads to over-segmentation and the introduction of false boundaries. Conversely, under-segmentation allows the false merger of image regions belonging to semantically different objects. MMIS overcomes this difficulty by automatically defining an appropriate number of classes and therefore regions, in a way that maintains the visual integrity of objects.

Figure 2 is an example that supports this statement with segmented output images obtained from the conventional IS (C-IS) [Cour] and the MMIS systems. The performance of both systems was tested using images with a wide spectrum of visual content and MMIS outperformed C-IS subjectively.

Objective performance tests were also performed using the Berkeley segmentation dataset and benchmark BSDS300 and results were obtained in terms of Precision / Recall measurements over a 100 images, see Table 1. Precision is the probability of generated boundaries to be true (Number of generated true boundaries / number of generated boundaries) and Recall is the probability of detecting the true boundaries (Number of detected true boundaries / total number of true boundaries). In general an over-segmented image would have high recall but low precision, and vice versa for an under-segmented image. F-measure is the geometric mean of Precision and Recall.

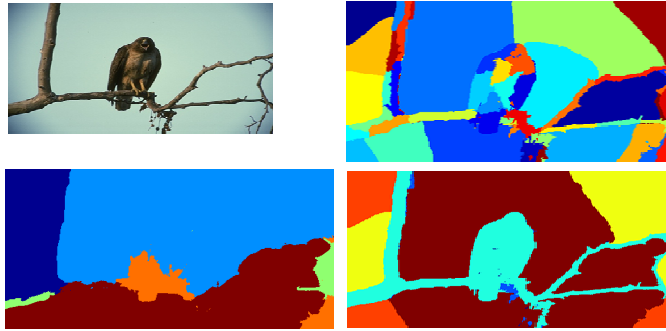


Figure 2. Top left, original input image, top right is C-IS output for 50 classes, bottom right is MMIS operated on C-IS 50 classes the no. of classes automatically calculated is 5, bottom left is the C-IS output run for 5 classes.

Note that MMIS produces an appropriate trade-off between Recall and Precision measurement values, across a wide range of class numbers.

Table 1. MMIS and C-IS system performance. F (a, b) is the F measure with a, b being the recall and precision respectively. ‘*’ denotes the average number of classes automatically generated over the entire input dataset

C-IS		MMIS	
Classes	F- measure	Classes	F-measure
8	0.48 (0.38, 0.65)	8* from 40	0.57 (0.58, 0.55)
10	0.49 (0.39, 0.65)	9* from 50	0.59 (0.60, 0.58)
30	0.59 (0.59, 0.58)	11* from 100	0.59 (0.67, 0.52)
40	0.59 (0.64, 0.55)	30 from 30	0.59 (0.59, 0.58)
50	0.59 (0.67, 0.53)	40 from 40	0.59 (0.64, 0.55)

In this way MMIS provides the observed, in subjective tests, and highly desirable object oriented segmentation performance. It is exactly this improved and robust, across different image contents, object oriented Image Segmentation performance that makes MMIS a particularly valuable part of image analysis /understanding systems.

REFERENCES

- Arbelaez P. et al, 2009. From Contours to Regions: An Empirical Evaluation. in *Proceedings Conference on Computer Vision and Pattern Recognition* IEEE, pp.2294-2301.
- Calderero F. and Marques F., 2010. Region Merging Techniques using Information Theory Statistical Measures. *IEEE Transactions on Image Processing*, Vol.19, pp. 1567-1586.
- Cour T. et al, 2005. Spectral Segmentation with Multiscale Graph Decomposition. in *Proceedings Computer Vision and Pattern Recognition* IEEE, Vol.2, pp. 1124-1131.
- Felzenszwalb P. and Huttenlocher D., 2004. Efficient Graph-based Image Segmentation. *International Journal of Computer Vision* Vol. 59.
- Martin D. et al, 2001. A Database of Human Segmented Natural Images and its Application to Evaluating Segmentation Algorithms and Measuring Ecological Statistics. in *Proceedings International Conference of Computer Vision*, vol. 2, pp. 416-423.
- Ren X. and Malik J. 2003. Learning a Classification Model for Segmentation. in *Proceedings International Conference of Computer Vision*, pp. 10-17.
- Shi J. and Malik J., 2000. Normalized Cuts and Image Segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 22, no. 8, pp. 888-905.
- Theodoridis S. et al, 2009. *Introduction to Pattern Recognition: A Matlab Approach*. Academic Press.
- Unnikrishnan R. et al, 2007. Toward Objective Evaluation of Image Segmentation Algorithms. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.29, no.6, pp.929-944.
- Wu Z. and Leahy L., 1993. An Optimal Graph Theoretic Approach to Data Clustering: Theory and its application to Image Segmentation. *IEEE Transaction Pattern Analysis and Machine Intelligence*, Vol. 15, no.11, pp. 1,101-1, 113.
- Zahn C.T., 1971. Graph-theoretic Methods for Detecting and Describing Gestalt Clusters. *IEEE Transactions on Computing*, vol. 20, pp. 68-86.

PHOTOMOSAIC USING QUAD-TREE STRUCTURE ON GPU

Jinsuk Yang, Choongjae Joo and Kyoungsu Oh
Department of Media, Soongsil University

ABSTRACT

Photomosaic is a method of representation an input image in forms of mosaics by a set of small tile images. Generally, equal rectangular tiles are used in photo mosaics. Therefore, there are limitations to pictorialization according to characteristics of images. In this paper, photo mosaics using quad-tree structure is proposed to create tiles in varies sizes. Initially, color variance per each level of an input image is computed using the mipmap of graphic hardware. Depending on the value of variance, the input image is divided into tiles in varies sizes. Each tile finds the most similar reference image. As a result, the method provides another pictorial effectiveness by dividing the input image into tiles in varies sizes depending on color held by the input image. In addition, all processing is done on the graphic hardware and thereby allowing high performance.

KEYWORDS

Photomosaic, Quad-Tree, Variance, Mipmap, Real-Time Rendering

1. INTRODUCTION

Photomosaic is a method of representation an input image in forms of mosaic by a set of small tile images. Mosaic has been frequently used for decorating floors, walls and handicrafts since ancient times due to its pictorial effectiveness and decorative values. There are many researches for photomosaic. As shown in Figure 1, previous photomosaic has variation in size, location and shape of tiles([1]~[4]). Silver et al. [1] divided an input image into equal rectangular tiles and then replaced each tile with the most similar image from image database. Hausner [2] and Dobashi et al. [3] arranged tiles along the contour line of an input image and used the Voronoi Diagram in order for the tiles to form suitable shapes. However, there are limitations of above mentioned methods, since all tiles used were in equal size without considering characteristics of images [5]. Gianpiero et al. [4] divided an input image into tiles using the quad-tree structure in order to add pictorial effectiveness according to characteristics of the image. The region having low color complexity was divided into large tiles and the region having high color complexity was divided into small tiles. Therefore, another pictorial effectiveness was developed. However, since the process of dividing tiles was computed totally on CPU, its performance is quite limited.

In this paper, similarly to Gianpiero et al. [4], the quad-tree structure is used to divide an input image into tiles in varies sizes depending on image characteristics. However, all processing are computed on graphic hardware and thus its performance speed is very quick.

Algorithm is divided into three steps as follows.

- 1) Generate quad-tree structure based on varies of each input image region.
- 2) Divide input image into differ sized tiles using quad-tree structure.
- 3) Divided tiles find and replace the most similar image of the reference images saved in the image database.

In this paper, in the chapter 2 we give more detailed descriptions of main algorithm, the chapter 3 consists of implement results, and in the chapter 4 concluded with future work.

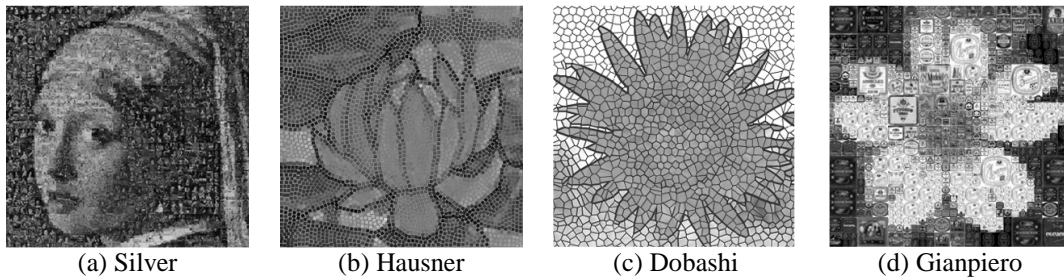


Figure 1. The previous work of photomosaic

2. PHOTOMOSAIC USING QUAD-TREE STRUCTURE

Quad-tree([6], [7]) is a tree structure having four children nodes per each node. The quad-tree divides rectangular into four regions and again it on demand. By using such process, the rectangular can be divided into tiles according to criteria, and the divided tiles can be used for photomosaic. For instance, the sublevel node region of the quad-tree has tiles with small sizes and the upper level node region has tiles with large sizes.

In this chapter, photomosaic using quad-tree structure on GPU is described. First, in the paragraph 2.1, generate quad-tree structure based on variation of each input image region. Paragraph 2.2 consist of dividing input image into differ size tiles using quad-tree structure. Lastly, in the paragraph 2.3, divided tiles find and replace the most similar image of the reference images saved in the image database. Figure 2 shows the overview of our algorithm.

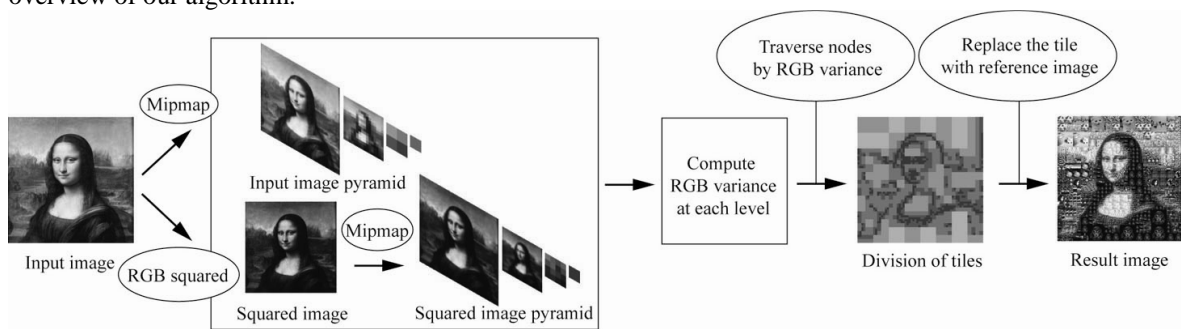


Figure 2. Overview of our algorithm

2.1 Computing Variance of Input Image

William et al. [8] saved the shortest distance from the light source to point in one channel of the shadow map and the squared value of the shortest distance in another channel in order to compute the variance of depth. As shown in equation 1, the variance($V(x)$) is computed based on the difference between the squared value of the average within the given filter size ($E(x^2)$) and the squared value of the average($E(x)^2$).

$$V(x) = E(x^2) - E(x)^2 \quad \text{(Equation 1)}$$

In this paper, region of the quad-tree are divided using variance of the input image. The method used by William et al. [8] was used to compute the variance. Initially the input image was mipmapped like a quad-tree structure. The mipmapped image is a collection of images from 0 level with $2^n \times 2^n$ texels to n level with $2^0 \times 2^0$ texels. Each texel of each level contains four children nodes. In addition, the average color of the four children nodes are saved in the image texel of upper levels. This process is conveniently and quick computed using the mipmap of graphic hardware. Similarly, the squared pixel value image is computed just as the input image done. The mipmapped input image and the mipmapped image with squared pixel value use the Equation 1 to compute variance per level.

2.2 Determining Size of Tile Using Variance

In this chapter, a method for dividing boundaries of the quad-tree using variance is described. Initially, the variance computed per level is traversed in a top-down form from the uppermost level(n level) to the lowermost level(0 Level). As shown in figure 3, the next lowermost level should be traversed when the variance saved in the current level should be greater than the threshold set by the user, and the traverse should stop when it is smaller. If the level where the traverse stopped should be called L, the tile size should be $2^{n-L} \times 2^{n-L}$.

Figure 4 shows tile sizes divided according to the variance. A region indicating high color variation(= high variance) of input image contains tiles in large sizes and a region indicating low variation(= low variance) contains tiles in small sizes.

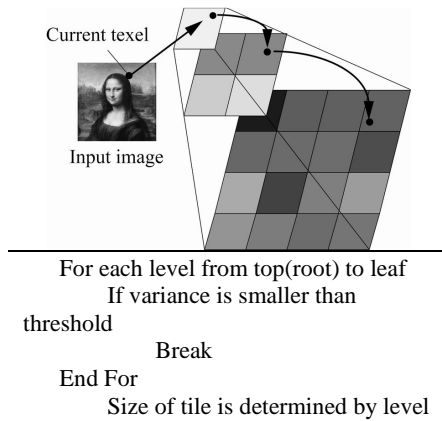


Figure 3. Determining size of tile using variance

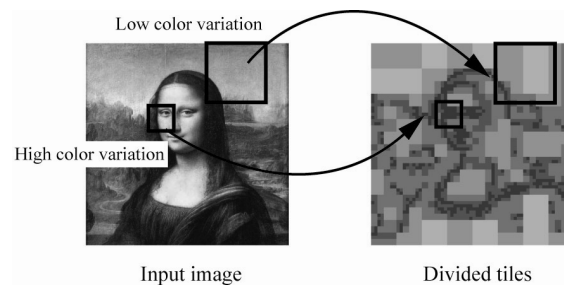


Figure 4. Example of tile size determining using variance

2.3 Selecting Reference Image for Each Tile

To achieve photomosaic close to an input image, the selection of reference images to replace tiles is important. Gianpiero et al. [4] used Antipole-tree structure [9] to find the most similar image to the target tile from image database in a quick speed. However, this method requires image database to be structured into the Antipole-tree structure on pre-processing. In addition, its similarity to the input image is quite low since an accurate search of multidimensional data such as an image is difficult.

In this paper, the exhaustive search method is used to find the most similar image. Then we select the most similar image using average of color value such as method of Adam et al. [10]. Initially, the reference images were turned into tiles as shown in figure 5, and a number of reference image textures were made. Then the mipmap was used to compute the average color of the reference image textures. A selection of the most similar reference image was done by comparing the average of input image located within the tile involved with the average of all reference images within image database. Finally, a reference image having the most similar average was selected to replace the tile involved.

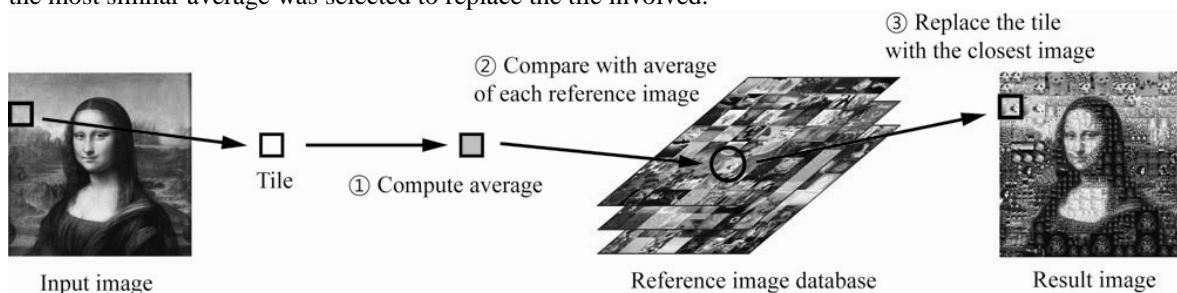


Figure 5. Processing of selecting reference image for each tile

3. RESULT

In this paper, DirectX 9.0 was used as graphic API, and nvidia Geforce GTX 480 graphic card and Intel CPU 3.0GHz were used as hard-wares for the implement. The resolution of input image is 512 by 512. We select 256 images at random and use them for reference images. Table 1. shows comparative performances between the tiles divided by CPU as described in the method used by Gianpiero et al. [4] and the tiles divided by GPU as proposed in this paper. It can be found that the compared to that from CPU, the results from GPU were approximately 14 times faster.

Figure 6 shows the results of various input images. It can be confirmed that the region indicating high variation is comprised of tiles in small sized and the region indicating small variation is comprised of tiles in large tiles. Figure 7 shows the results of tiles divided by traversing variance of one input image with criteria values of mutually different threshold. It can be seen that a number of large tiles are created since the frequency of traverse stop in upper level rises as the criterion value increases.

Table 1. Comparison performance result



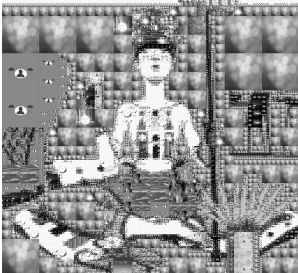
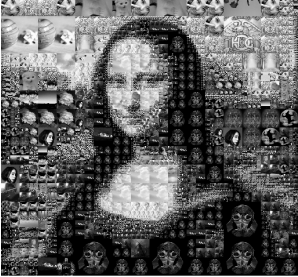
	CPU	GPU
FPS(frame per second)	18~19	257~258
(a) Input images		
(b) Result images		

Figure 6. Some examples of photomosaic by our algorithm

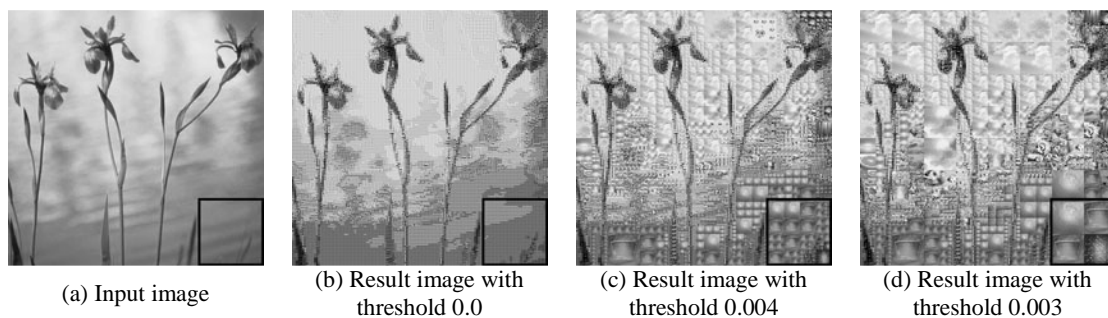


Figure 7. Result images according to different threshold

4. CONCLUSION AND FUTURE WORK

In this paper, photomosaic using quad-tree structure on GPU is described. The method proposed in this paper is set to form tiles in different sizes based on variance of the input image, and allowing high performance since all processing are computed on GPU. In addition, no complex data structuring is required since the quad-tree structure is established using the mipmap for easy implement.

However, the problem regarding overlapping reference images between adjacent tiles isn't considered in the method proposed in this paper. Therefore, some images being displayed overlap between tiles. Consistent researches are to be performed in the future to improve such problem.

ACKNOWLEDGEMENT

This paper was supported by BK 21 Program in Soongsil University.

REFERENCES

- [1] R.Silvers and M. Hwaley, 1997. *Photomosaics*. Holt Paperbacks, New York, NY, USA.
- [2] HAUSNER. A. 2001. Simulating Decorative Mosaics. *In Proceedings of SIGGRAPH*. pp. 573-580.
- [3] DOBASHI J et al., 2002. A Method for Creating Mosaic Images Using Voronoi Diagrams. *In Proceedings Eurographics*. pp. 341-348.
- [4] Gianpiero Di Blasi, et al., Smart Ideas for Photomosaic Rendering. *In Proceeding of Eurographics Italian Chapter Conference*. pp. 267-272.
- [5] Dongwann Kang, Young-Sup Park, Sang-Hyun Seo, Kyung-Hyun Yoon, 2006. Tow Layer Image Tile Mosaics. *In Proceedings of the Eurographics2006 short paper*. pp. 145-148.
- [6] FINKEL R.A., BENTLEY J.L., 1974. Quad trees: A data structure for retrieval on composite keys. *Acta Informatica* Vol. 4, No. 1, pp. 1-9.
- [7] John R. Smith, Shih-Fu Chang, 1994. Quad-tree segmentation for texture-based image query. *MULTIMEDIA In Proceedings of the second ACM international conference on Multimedia*. pp. 279-286.
- [8] William Donnelly, Andrew Lauritzen, 2006. Variance shadow maps. *In Proceedings of the symposium on Interactive 3D graphics and games*. pp. 161-165.
- [9] D. Cantone et al., 2005. Antipole tree indexing to support range search and k-nearest neighbor search in metric spaces. *IEEE Transactions on Knowledge and Data Engineering*. Vol. 17, No. 4, pp. 553-550.
- [10] Adam Finkelstein and Marisa Range, 1998, Image Mosaics, Lecture Notes in Computer Science, Vol. 1375, pp. 11-22.

AN ENLARGEMENT METHOD BASED ON UNDECIMATED WAVELET TRANSFORM AND ENHANCEMENT FUNCTION

Wen-Chien Yen, Ying-Wen Chang, Zhong-Xin Yu and Yi-In Lee
*Department of Information Management, Chung Chou Institution of Technology
6, Line 2, Sec 3, Shan-Chiao Rd., Yuanlin Changhwa. 51003, Taiwan R.O.C*

ABSTRACT

The image resolution reconstruction is a partial technology in many application including medical imaging, satellite imaging, and video applications. Due to the wavelet representation has characteristics of the efficient time-frequency localization and the multi-resolution analysis; the wavelet transforms are suitable for processing the image resolution enhancement. Therefore, this method focuses on wavelet-based image resolution enhancement and proposes a framework of image resolution enhancement. The wavelet-domain image resolution enhancement algorithm is based on the estimation of detail wavelet coefficients at high resolution scales. The method exploits shape function according to wavelet coefficient correlation in a local neighborhood and employs undecimated discrete wavelet transform to estimate the unknown detail coefficients. The simulation results objectivity show that the proposed method is considerably superior to conventional image interpolation techniques.

KEYWORDS

Elargenent method, Undicimated wavelet transform, Shape function

1. INTRODUCTION

The image resolution reconstruction is proved to be useful in many practical cases where multiple frames of the same scene can be obtained, including medical imaging, satellite imaging, and video applications. Image interpolation generates a larger image from a smaller size image. The problem of image interpolation is that the image magnifies many times and loses the sharpness of the picture. In this paper, we propose a wavelet based method which estimates the high-frequency wavelet coefficients and reconstruct the high resolution image.

Wavelet transform is useful to image resolution enhancement. A common feature is the assumption that the image pixel values are seemed to be the low-pass filtered subband coefficients of a wavelet-transformed high-resolution image and used to estimate the detail wavelet coefficients in high-pass subbands. The trivial approaches approximate the HR image by filling the unknown subbands with zeros and applying the inverse wavelet transform. More sophisticated methods have attempted to estimate these unknown detail wavelet coefficients.

The detail coefficients are estimated using the coarser subband coefficients. The sign estimation of descendant coefficients in wavelet subbands is generally assumed to be random and is relied upon [1, 2]. HMT-based methods model the unknown wavelet coefficients as belonging to mixed Gaussian distributions which are symmetrical around the zero mean [5, 6]. The proposed method uses a wavelet-domain image resolution enhancement algorithm which is based on the estimation of detail wavelet coefficients. According to wavelet coefficient correlation in a local neighborhood, the proposed method exploits shape function to enhance the intensity of discontinuity and then employs undecimated discrete wavelet transform to estimate the unknown detail coefficients. Finally, the proposed method uses quadtree weight function to modify the estimated detail wavelet coefficients.

2. RESOLUTION ENHANCEMENT

The approach to image interpolation, which we call prediction of image detail, can be explained in Fig. 1. The high resolution image is represented as the signal X at the input to the filter bank. The low resolution image, more coarsely sampled image, is the result of a low-pass filtering operation followed by decimation to give the signal A . The low-pass filter, L , represents the effects of the image acquisition system. The original high-resolution image X filters with the high pass filter H to obtain the detail signals D . If we have perfect reconstruction filter bank, we can reconstruct the original image, it would then be possible to. In resolution enhancement, it's assumption that the signal A is known and the detail signal D is unknown, and the estimation of the signal D is a critical point to decide the visual quality image. Many recent researches discuss the estimation technologies.

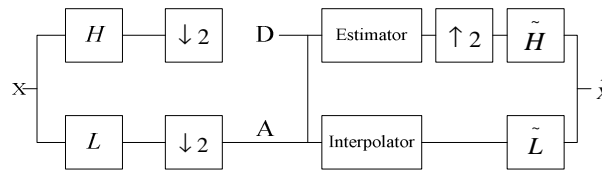


Figure 1. Problem Model

In the problem of resolution enhancement, most of the difficulties arise in areas around edges and sharp changes. Around edges, many resolution enhancement methods tend to smooth and blur image detail. Fortunately, most of the signal information is often carried around edges and areas of sharp changes and can be used to predict these missing details from a sampled image.

3. PROPOSED ALGORITHM

Because the image magnify, the blurred edge is an obvious problem. The proposed method suggests shape function to enhance edge information and undecimated discrete wavelet transform to estimate detail wavelet transform coefficients. Furthermore, quadtree weight function is adopted to adjust the detail wavelet transform coefficients. The visual quality of enlarging image is improved. The detailed descriptions of shape function, undecimated discrete wavelet transform and quadtree weight function are as follows.

3.1 Shape Function

The decimated discrete wavelet transform coefficients of images tend to be separated to four subbands. After wavelet transform, the full image energy will be congregated in low frequency. The coefficients imply noise and edge in high frequency subbands (LH, HL, and HH).

The algorithm treats the image (X) as the LL subband coefficients of an image that decomposed by decimated wavelet transform. If the method directly decomposes the image (X) by undecimated wavelet transform to estimate the coefficients in LL, HL, and HH. The discontinuations of the reconstructive image is blur and unobvious therefore the proposed algorithm uses shape function to enhance the edge of LL in order to preserve more edge information while estimating the coefficients in LL, HL, and HH. The more edge information preserve, the more clarity advance.

$$Var_v = [2x(m,n) - (x(m,n+1) + x(m,n-1))] / \lambda \quad (3)$$

$$Var_h = [2x(m,n) - (x(m+1,n) + x(m-1,n))] / \lambda \quad (4)$$

$$z(m,n) = x(m,n) + Var_v + Var_h \quad (5)$$

In the equations (3), (4), and (5), $x(m,n)$ is the current pixel to be modified using neighborhood correlation and λ is a scale factor. $z(m,n)$ is the enhancement pixel. Var_v and Var_h are the vertical and horizontal variances and both of them affect the enhancement intensity. The large Var_h variance means that there is a horizontal boundary and then shape function heavily enhances the edge intensity. On the contrary,

the small Var_h variance means that there is no horizontal boundary and then shape function lightly enhances the edge intensity.

3.2 Undecimated Discrete Wavelet Transform

The traditional DWT has exerted a remarkable influence on several signal processing applications such as denoising, estimation, and compression. However, in signal denoising, the DWT is known to create artifacts around the discontinuities of the input signal. These artifacts degrade the performance of the threshold-based denoising algorithm. It has been shown that many of the artifacts could be suppressed by a redundant representation of the signal.

The translation invariant property of the UDWT makes it preferable for use in various signal processing applications, as it relies heavily on spatial information. Let $L(z)$ and $H(z)$ respectively the low-pass (LP) and high-pass (HP) filters, let $X(z)$ be the low-resolution image.

$$LL_0(z) = L_{col}(z)L_{row}(z)X(z) \quad (6)$$

$$HL_0(z) = L_{col}(z)H_{row}(z)X(z) \quad (7)$$

$$LH_0(z) = H_{col}(z)L_{row}(z)X(z) \quad (8)$$

$$HH_0(z) = H_{col}(z)H_{row}(z)X(z) \quad (9)$$

This algorithm applies the wavelet transform but omits down-sampling. Figure 2 shows that the ‘lena’ test image is decomposed by undecimated discrete wavelet transform into four subbands. The most edge informations are the coefficients in the high frequency subbands therefore the proposed method employs the four subbands that decomposed by undecimated discrete wavelet transform to estimate the high frequency subbands coefficients and composes these subbands to high-resolution image by common discrete wavelet transform.

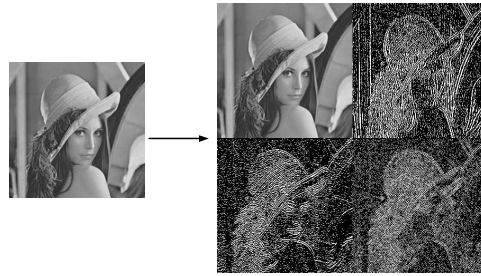


Figure 2. Undecimated Discrete Wavelet Transform

3.3 Quadtree Weight Function

After undecimated wavelet transform, the high-frequency coefficients is roughly estimated. In improvement of the perceptual quality, the adjustment of those high-frequency wavelet coefficients is important. The enhanced low-resolution image \hat{X} is decomposed by discrete wavelet transform into four subbands that includes LL_1 , LH_1 , HL_1 , and HH_1 . We exploit the inter correlation between LL_1 , LH_1 , HL_1 , and HH_1 to estimate the weight factor w_c in high-frequency wavelet subbands LH_0 , HL_0 , and HH_0 . In Fig. 3, the wavelet coefficient P is the parent of wavelet coefficients C_1, C_2, C_3 and C_4 and the wavelet coefficients C_1, C_2, C_3 and C_4 is the children of wavelet coefficient P . According to this quad-tree property, while the coefficient is classified to significant coefficient and its children is also significant. σ is the standard deviation of subband and α is a weighting factor. If the parent coefficient is classified to significant, the children weight factor w_c is $\alpha \times w_p$, the other coefficients set to zero. The quadtree weight function formula is as follows:

$$w_c = \begin{cases} w_p \times \alpha & \text{if } |w_p| \geq \sigma \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

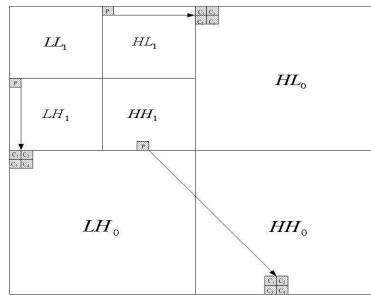


Figure 3. Quad-tree Property in Wavelet Domain

3.4 Flowchart

The complete block diagram of the proposed algorithm presents in Fig. 4. First, the input is a high-resolution image. This image (HR) gets low-resolution image (LR) through down-sample (low-pass filter and down-sample) and then the low-resolution image uses shape function to enhance the low-resolution image. The enhanced low-resolution image is symbol as \hat{X} . And then the proposed algorithm decomposes the enhanced low-resolution image into the LH, HL, and HH subband coefficients (\hat{X}_{HL} , \hat{X}_{LH} , and \hat{X}_{HH}) by undecimated discrete wavelet transform and uses the quadtree weight function to adjust high-frequency subband coefficients. Finally, the proposed method synthesis \hat{X} , \hat{X}_{HL} , \hat{X}_{LH} , and \hat{X}_{HH} to get the output image by inverse discrete wavelet transform.

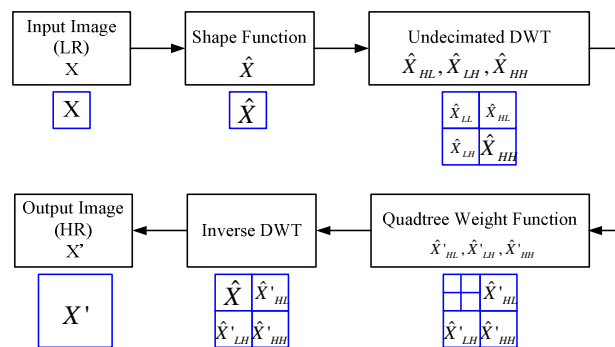


Figure 4. Proposed Algorithm Flowchart

4. SIMULATION RESULTS

In our experiments, the test images are ‘Lena’ and ‘Baboon’. These images are regarded as the unknown HR originals. There are gray-level image with a size of 512 x 512 with 8 bits per pixels. We have filtered and downsampled the original images by factors of 2 for $2\times$ resolution enhancement and used those as the available LR images. PSNR value has been accepted as a widely used quality measurement in the field of image processing.

The simulation results obtained by simulating the above algorithm are shown in Tab. 1 where PSNR performance is tabulated for $2\times$ enlargement factors. Table 1 show the comparison of the proposed algorithm and the standard bilinear interpolation approach with a number of alternative approaches in the wavelet domain. The wavelet-based resolution enhancement methods uses two well-known classes of analysis-synthesis filters namely Haar and Daubechies 9/7 and the unknown detail coefficients are estimated as zeros. The simulation results shows that PSNR of the proposed method is outperform than that of the other image resolution enhancement methods.

Table 1. PSNR results for $2\times$ enlargement of 'Lena' and 'Baboon' test images (from 256×256 to 512×512)

Method / Image	Lena(dB)	Baboon(dB)
Bilinear	30.13	22.85
NEDI [4]	34.10	23.87
WZP(Haar)	31.46	23.61
WZP(Db. 9/7)	34.45	24.22
Carey et al. [2]	34.48	24.24
HMM [5]	34.52	24.24
HMM SR [7]	34.61	24.31
WZP and CS [8]	34.93	24.28
Proposed Method	35.46	24.33

5. CONCLUSIONS

This wavelet-domain image resolution enhancement algorithm proposes shape function with wavelet coefficient correlation to enhance the discontinuances and employs undecimated wavelet transform to estimate the unknown detail coefficients. Finally, the proposed algorithm uses quadtree weight function to adjust the high frequency coefficients. Experimental results demonstrate that the performance of wavelet-based schemes outperforms that of conventional methods and shows that our method outperforms conventional image resolution enhancement methods such as bilinear interpolation, for a wide range of standard test images. More importantly the proposed method compares favorably with state-of-the-art competing methods operating in the wavelet domain both in objective and subjective terms.

ACKNOWLEDGEMENT

This research is supported partially by the National Science Council of Taiwan under the contract number of NSC 99 - 2221 - E - 235 - 007.

REFERENCES

- [1]Kinebuchi, K., Muresan, D.D., and Parks, T.W.: 'Image interpolation using wavelet-based hidden Markov trees'. Proc. ICASSP01, May 2001, 3, pp. 7–11
- [2]Zhao, S., Han, H., and Peng, S.: 'Wavelet domain HMT-based image superresolution'. Proc. ICIP03, Sept. 2003, 2, pp. 933–936
- [3]Nguyen, N.: 'Numerical techniques for image superresolution'. PhD Thesis, Stanford University, 2000
- [4]M. Unser, A. Aldroubi, and M. Eden, "Enlargement or reduction of digital images with minimum loss of information," IEEE Trans. Image Processing, vol. 4, pp. 247–257, Mar. 1995.
- [5]K. Xue, A. Williams, and E. Walowitz, "And edge-restricted spatial interpolation algorithm," J. Electron. Imag., vol. 1, pp. 152–161, Apr. 1992.

SEGMENTATION AND DISCRIMINATION OF BREAST TUMORS IN ULTRASONIC IMAGES USING AN ENSEMBLE CLASSIFIER AND APPLICATION TO A DIAGNOSTIC SUPPORT SYSTEM

Atsushi Takemura

*Tokyo University of Agriculture and Technology
2-24-16, Naka-cho, Koganei-shi, Tokyo 184-8588, Japan*

ABSTRACT

This study proposes a novel diagnostic support system for automated diagnosis of breast tumors in ultrasonic images. The proposed system performs breast tumor segmentation and discrimination using the AdaBoost machine learning algorithm. In this study, novel features for segmentation and discrimination are defined on the basis of an estimated log-compressed Nakagami distribution parameter, which corresponds to physical characteristics of ultrasonic echoes from tumors. The segmentation process is performed by using an ensemble classifier trained by the AdaBoost algorithm with a Markov random field and is followed by a geodesic active contour to increase the sophistication of the extracted tumor boundary. The process for breast tumor discrimination is determined using the multi-class AdaBoost algorithm. The performance of the proposed diagnostic system was evaluated by 10-fold cross validation tests, where 300 carcinomas, 60 fibroadenomas, and 50 cysts were used.

KEYWORDS

Ultrasonic image, breast tumor, Nakagami distribution, AdaBoost, Markov random field, geodesic active contour

1. INTRODUCTION

Early discrimination and treatment of breast tumors is important in medicine and for the general public. The following ultrasonic image diagnostic observations are critical in the discrimination of breast tumors: (a) internal echo, (b) shape, and (c) boundary echo (Hong, 2005). However, ultrasonic echoes obtained from breast tumors have low intensities, and hence, the tumor regions are not clear in ultrasonic images. The automated segmentation and discrimination of breast tumors (e.g., carcinomas and fibroadenomas) in ultrasonic images is therefore essential for computer-aided diagnosis (CAD) and surgery.

There has been a great deal of research involving the discrimination of breast tumors using internal echoes and shape features. For instance, texture analysis is used to detect the differences in the statistical characteristics of internal echo patterns for benign and malignant tumors (Chen, 2002; Huang, 2006). On the other hand, some methods used features corresponding to the tumor's shape (Chang, 2005), where the depth-width ratio and the roundness are used. To improve the discrimination accuracy of breast tumors, it is also necessary to use particular characteristics of ultrasound echoes obtained from tumors. Shankar *et al.* demonstrated that the K-distribution and Nakagami distribution can quantify the homogeneity of an internal echo pattern in a breast tumor (Shankar, 2001). However, these conventional methods employ features corresponding to only one of the two diagnostic observations ((a) or (b)), although use of the features based on all three characteristics ((a)–(c)) would provide more accurate results. Moreover, conventional methods for estimating the K-distribution and Nakagami distribution are not applicable to log-compressed ultrasound images obtained from commonly used ultrasonic medical equipment.

I have previously proposed a method for estimating the K-distribution parameter (Takemura, 2010) and the Nakagami distribution parameter (Takemura, 2009) from a log-compressed ultrasonic image. From these studies, statistics based on a log-compressed K-distribution (defined as K-p) are effective features for the segmentation and discrimination of breast tumors. However, the discrimination accuracy for tumors having

unclear boundary echoes was higher when a log-compressed Nakagami distribution parameter (defined as N-m) was used in addition to K-p. The preceding study (Takemura, 2010) also proposed a novel cost sensitive extension of AdaBoost with Markov random field (MRF) priors to avoid the irregular shape, isolated points, and holes observed in segmentation results. This method uses features consisting of statistics that are based on K-p and the conventional features in order to incorporate the shape, internal, and boundary echoes. Tumors having clear boundaries were accurately extracted using the preceding method, but this was not possible for the tumors having unclear boundaries.

The conventional methods stated above also have a disadvantage in that images that were analyzed and evaluated were not original ultrasonic images. Instead, they were manually extracted regions neighboring the tumors and are referred to as regions of interest (ROI). Moreover, evaluations using the conventional methods for tumor discrimination generally used the actual regions of tumor extracted manually. As shown in Figure 1, this study proposes a fully automated diagnostic system that is constructed by sequential processes: 1) ROI extraction, 2) tumor segmentation, and 3) tumor discrimination. To improve the accuracy of segmentation and discrimination, novel features based on N-m are added to the feature set.

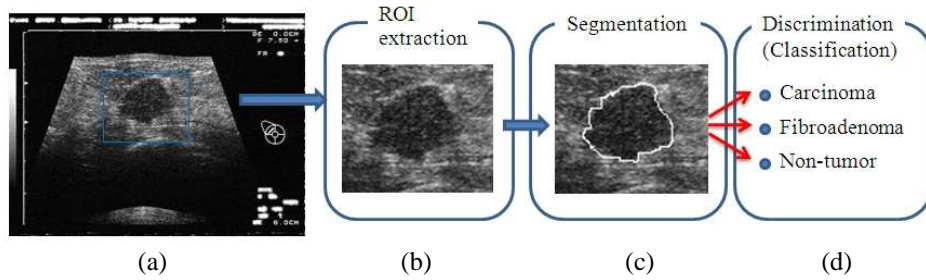


Figure 1. Schema of the proposed diagnostic support system: (a) original image, where the description in the image relating to patient information was blurred for privacy reasons, (b) image of extracted ROI, (c) image of segmented tumor, and (d) classification of tumor

2. METHODOLOGY

2.1 Automated ROI Extraction

The description and lines superimposed in an original image are eliminated by using binary morphological operations (Haralick, 1987) that are based on linear structural elements, followed by thresholding of the original ultrasonic image. The adaptive speckle suppression filter (Aleman-Flores, 2007) suppresses speckle pattern noise, which is an interference pattern caused by scattering echoes from normal tissues. Rough extraction of the tumor region is subsequently achieved by thresholding the speckle-suppressed image. The extracted ROI is a rectangular region whose vertical and horizontal boundaries extend 30 pixels beyond the approximate edges of the tumor region so that the segmentation process (described in section 2.3) computes features of individual pixels in each neighborhood region.

2.2 Estimation of a Log-Compressed Nakagami Distribution Parameter

The L -th order moment m_L , ($L=1,2,3,4$) of intensities of a log-compressed ultrasonic echo signal are obtained from the equations (1)–(4) (Takemura, 2009):

$$\mu_1 = C \varphi(m) \quad (1)$$

$$\mu_2 = C^2 [\varphi(m)^2 + \psi^{(1)}(m)] \quad (2)$$

$$\mu_3 = C^3 [\varphi(m)^3 + 3\varphi(m)\psi^{(1)}(m) + \psi^{(2)}(m)] \quad (3)$$

$$\mu_4 = C^4 [\varphi(m)^4 + 6\varphi(m)^2\psi^{(1)}(m) + 4\varphi(m)\psi^{(2)}(m) + 3\psi^{(1)}(m)^2 + \psi^{(3)}(m)] \quad (4)$$

where $\varphi(m) = \psi(m) - \beta$, $\psi(m)$ is the digamma function, and $\psi^{(l)}(m)$ is the l -th order polygamma function. The skewness S is given by the following equation:

$$S = \mu_{c3} / \mu_{c2}^{3/2}, \quad (5)$$

where μ_{cL} is the L -th order central moment. When using the L -th order moment m_L , the theoretical value of the skewness $S(m)$ is given by

$$S(m) = \frac{\mu_3 - 3\mu_1\mu_2 + 2\mu_1^3}{(\mu_2 - \mu_1^2)^{3/2}} = \frac{\psi^{(2)}(m)}{\psi^{(1)}(m)^{3/2}}. \quad (6)$$

Note that $S(m)$ can be computed without the unknown system parameters of the log-compressor in ultrasonic diagnostic equipment. The procedure for estimating N-m is as follows: 1) Compute the L -th order central moments of intensity in a given log-compressed ultrasonic image. 2) Compute the skewness S using Eq. 6. 3) Estimate the parameter \hat{m} by selecting m that minimizes the Euclidean distance between S and $S(m)$.

2.3 Segmentation of Breast Tumors

In the preceding study (Takemura, 2010), a total of 1,221 features per pixel were used for segmentation based on the following ten statistics that were computed from the intensities included in the neighboring region. The neighboring region was a square whose width and height ranged from 9–21 pixels in two pixel intervals. In this study, to improve the segmentation accuracy, novel features based on N-m are added as (11) and (12), bringing the total number of features to 1,319.

(1) average, (2) standard deviation, (3) median, (4) maximum, (5) minimum, (6) kurtosis, (7) skewness, (8) output of the speckle suppression filter (Aleman-Flores, 2007), (9) K-p, (10) gradient magnitude of K-p, (11) N-m, and (12) gradient magnitude of N-m.

The segmentation process is performed by using an ensemble classifier that is trained by a cost sensitive extension of AdaBoost with MRF priors in order to prevent the occurrence of irregular shapes, isolated points, and holes in the segmentation results. In a series of training rounds of AdaBoost, each weak classifier selects an optimal feature that can be extracted from the tumor regions by thresholding. To increase the sophistication of the boundary of the tumor, a geodesic active contour (Caselles, 1997) is applied after the segmentation.

2.4 Discrimination of Breast Tumors

The proposed system performs discrimination of breast tumors after the segmentation process. This discrimination process uses a multi-class ensemble classifier, trained by AdaBoost.M2 (Freund, 1997) with feature selection (Takemura, 2010). Cysts are not generally classified by physicians as tumors, and the ensemble classifier classifies an extracted region obtained by automated segmentation into the following three classes: 1) carcinoma, 2) fibroadenoma, and 3) non-tumor regions. The preceding study used 208 features corresponding to internal echo and shape, which included statistics that are based on K-p. In order to improve the discrimination accuracy, the features used by this study includes statistics that are based on N-m in addition to the conventional features. Therefore, a total of 226 features are used for discrimination.

3. EXPERIMENTAL RESULTS AND DISCUSSION

3.1 Materials and Validation

The resolution of the tumor images in this experiment is $10^{-1} \times 10^{-1}$ mm²/pixel, and they were quantized to 256 levels of grayscale. For each tumor, the width ranges from 5 to 28 mm. The experimental ultrasonic

images consisted of 300 carcinomas, 60 fibroadenomas, and 50 cysts. The performance of the proposed system was evaluated by 10-fold cross validation testing.

3.2 Segmentation of Breast Tumors

Figure 2 shows an example of a tumor (carcinoma) and the region extracted using the segmentation method. Figures 2(a)–(c) show the original image, the extracted ROI, and the true region extracted manually, respectively. Figure 2(d) shows the segmentation result obtained using the preceding method without the N-m based features. Figure 2(e) shows the segmentation result with the novel features that are based on N-m. From the 10-fold cross validation test, the Jaccard index between true regions and the regions extracted by the preceding segmentation without N-m ranged from 71.26% to 94.68%, with an average of 92.91%. The Jaccard index between true regions and the regions extracted by the segmentation with N-m ranged from 78.79% to 95.41%, and the average was greater than 94.06%. These results indicate that the features based on N-m were effective for improving the segmentation accuracy.

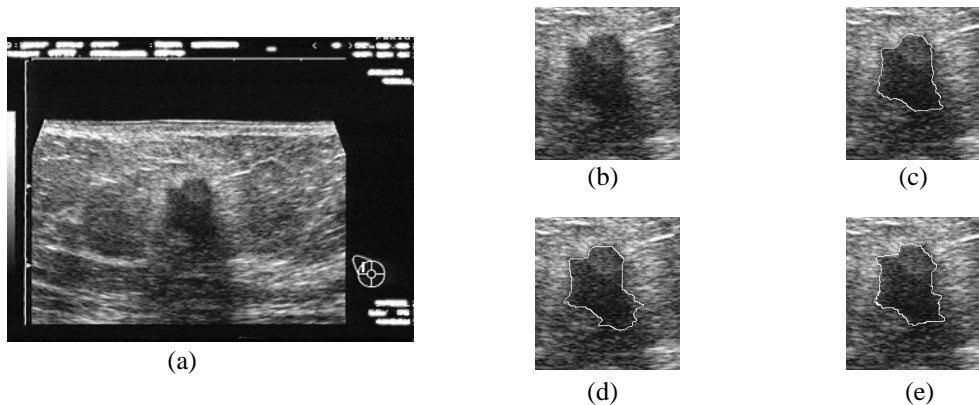


Figure 2. Examples of segmentation results. (a) the original image, with the patient's information blurred for privacy reasons, (b) image of extracted ROI, (c) image of true region, (d) segmentation (without N-m), and (e) segmentation (with N-m)

3.3 Discrimination of Breast Tumors

The discrimination accuracy was evaluated by the 10-fold validation cross test. Figure 3 shows the relationship between the average error rates and the number of weak classifiers of the AdaBoost.M2 algorithm. Figures 3(a) and (b) show the error rates obtained with the preceding method (Takemura *et al.*, 2010) and those obtained with the proposed method using statistics based on N-m, respectively. These figures show that the error rates decreased as the number of weak classifiers increased. From Figure 3(a), the error rates were observed to be at a minimum when there were eight weak classifiers, but the error rate for fibroadenoma had not yet reached its minimum value. Figure 3(b) indicates that the error rate of each class reached 0% when there were seven weak classifiers. These results demonstrate that N-m based statistics are effective features for improving the discrimination accuracy.

4. CONCLUSION

This study proposed an automated diagnostic system for the diagnosis of breast tumors. The proposed system consists of a sequential process involving 1) ROI extraction, 2) tumor segmentation, and 3) tumor discrimination. To improve the accuracy of segmentation and discrimination, novel features based on N-m were added to the feature set used in the preceding study. A validation test using 300 carcinomas, 60 fibroadenomas, and 50 cysts indicated that the novel features that were statistics based on the N-m improved the segmentation and discrimination accuracies of breast tumors in ultrasonic images. For future study,

investigations using a wider variety of tumor cases, especially some that are more difficult (e.g., infiltrating cancer), will be important for determining the applicability of the proposed system to practical scenarios.

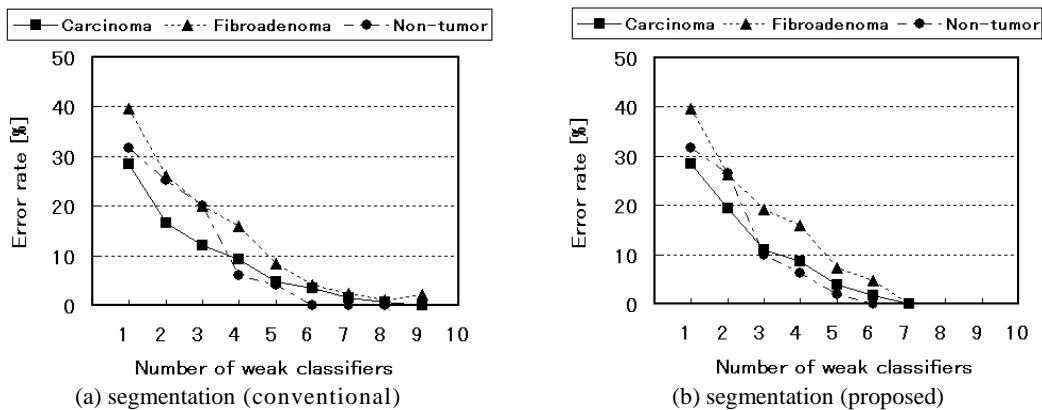


Figure 3. Graphs showing the relationship between error rate and the number of weak classifiers in the discrimination process.

ACKNOWLEDGMENTS

The author would like to thank Dr. Kazuhiko Hamamoto of Tokai University and the ultrasonographers at Tokai University Hachioji Hospital for supplying ultrasonic image data and their useful comments. This work was partly supported by a Grant-in-Aid for Scientific Research (KAKENHI) 22500878 by the Japan Society for the Promotion of Science (JSPS).

REFERENCES

- Aleman-Flores, M. *et al.*, 2007, Texture-oriented anisotropic filtering and geodesic active contours in breast tumor ultrasound segmentation, *In J. Math. Imaging and Vision*, vol.28, no.1, pp.81-97.
- Caselles, V. *et al.*, 1997, Geodesic active contour, *In Int. J. Comput. Vision*, vol.22, no.1, pp.61-79.
- Chang, R.F. *et al.*, 2005, Automatic ultrasound segmentation and morphology based diagnosis of solid breast tumors, *In Breast Cancer Research and Treatment*, vol.89, no.2, pp.179-185.
- Chen, D.R. *et al.*, 2002, Diagnosis of breast tumors with sonographic texture analysis using wavelet transform and neural networks, *In Ultrasound in Med. & Biol.*, vol.28, no.10, pp.1301-1310.
- Freund, Y. *et al.*, 1997, A decision-theoretic generalization of on-line learning and an application to boosting, *In J. Comput.and Syst.Science*, vol.55, no.1, pp.119-139.
- Haralick, R. M. *et al.*, 1987, Image analysis using mathematical morphology, *In IEEE Trans. PAMI*, vol.9, no.4, pp.532-550.
- Hong, A.S. *et al.*, 2005, BI-RADS for sonography: Positive and negative predictive values of sonographic features, *In American Journal of Roentgenology*, vol. 184, pp.1260-1265.
- Huang, Y.L. *et al.*, 2006, Diagnosis of breast tumors with ultrasonic texture analysis using support vector machines, *Neural Computing & Applications*, vol.15, no.2, pp.164-169.
- Shankar, P.M. *et al.*, 2001, Classification of ultrasonic B-mode images of breast masses using Nakagami distribution, *In IEEE Trans.UFFC*, vol.148, no.2, pp.569-580.
- Takemura, A., 2009, Estimation of a log-compressed Nakagami distribution and application to breast tumor discrimination in ultrasonic images, *In Proc. IADIS Applied Computing 2009*, pp.191-196.
- Takemura, A., *et al.*, 2010, Discrimination of breast tumors in ultrasonic images using an ensemble classifier based on the AdaBoost algorithm with feature Selection, *In IEEE Trans. Med. Imag.*, vol.29, no.3, pp.598-609.
- Takemura, A., *et al.*, 2010, A cost-sensitive extension of AdaBoost with Markov random field priors for automated segmentation of breast tumors in ultrasonic images, *In Int. J. CARS*, vol.5, no.5, pp.537-647.

INTEGRATION DATA HIDING WITH JP3D COMPRESSION ON VOLUMETRIC MEDICAL IMAGES

Marouen Toueti and Azza Ouled Zaid
*SysCom Laporatory, National Engineering School of Tunis
1002, le Belvedere Tunis, Tunisia*

ABSTRACT

In this paper we discuss the combination of data hiding with compression in order to improve patient data management and maintainability. Data hiding is adapted here for interleaving patient information with volumetric medical images, while offering privacy and security. The image compression scheme under consideration is JP3D (JPEG2000 Part 10 standard) that gives support to volumetric datasets. The embedding is applied in the Discrete Wavelet Transform (DWT) domain, before passing through the 3D-EBCOT (three dimensional Embedded Block Coding with Optimized Truncation) algorithm. By this way, the hidden data can be detected on the fly during image decoding, or, whenever we apply the DWT to the reconstructed image. Experiments performed on a database of Magnetic Resonance and Computerized Tomography sequences show that our data hiding scheme for data hiding performs a large data payload. Furthermore, it is robust against JP3D compression attack whereas keep a relative lower distortion.

KEYWORDS

Image coding, Error correction coding, Data security, Wavelet transforms, Biomedical communication

1. INTRODUCTION

Modern health care infrastructure is based on digital information management. The expanding paradigm of digital medical imaging and the recent advancement in information and communication technologies provide new means to access, handle and move medical images. An attractive motivation is to have the complete medical information of a patient available in one consistent application rather than over several information systems. The confidentiality of the medical data is very critical and thus it is essential to efficiently hide the data during transmission. Medical image watermarking, for data hiding purpose (Giakoumaki et al.-2006), is the process of hiding a set of patient data into a diagnosis image imperceptibly such that it does not perceptually distort the image and such that the hidden data can be accurately recovered at the receiver end. The major requirements of data hiding (robustness, imperceptibility and capacity) are of specific importance to medical images also. In particular, the maximal number of embedded bits an image can hold with an acceptable distortion should be large enough for physicians to write their opinion or annotation. This is a critical point, which can be resolved when applying data embedding to volumetric datasets (Kobayashi et al.-2008). As they are typically of large size, volumetric medical images must be compressed with minimal bitstream size to save storage space and minimize transmission overheads. Several methods have been explored so far, in order to integrate the dual operations of compression and data hiding on digital images. However, most of them deal only with video and still images (Cheng et al.-2000) (Campisi-2005) and lack satisfactory solutions for multiframe modalities (Kobayashi et al.-2008).

Joint data hiding/compression system requires extreme care when applied to medical images. The patient data must be embedded imperceptibly such that it does not perceptually distort the image and such that the hidden data can be accurately recovered at the receiver end. In this paper, the issue of integrated data hiding and wavelet based compression is investigated to ensure secure medical information management while reducing storage and transmission overheads. The adopted data hiding process is based on a modified version of quantization index modulation (QIM) strategy. The popularity of QIM is, in part, due to its ease of implementation, computational flexibility and large data payloads (Li and Cox-2007). In our

watermarking/compression scheme, the embedding is applied in the discrete wavelet transform (DWT) domain, in conjunction with JP3D coder (JPEG-2000 Part 10), which is a complete volumetric coding

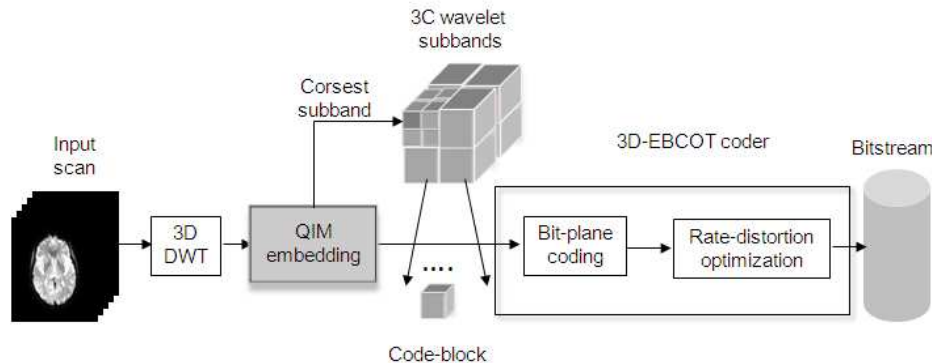


Figure 1. Main building blocks of watermarking/JP3D compression system.

scheme (Schelkens et al.-2006). We investigate the appropriateness of the proposed system for Magnetic Resonance (MR) and Computerized Tomography (CT) medical imaging modalities. Based on the experimental results, the proposed hybrid scheme distinguishes itself by its large payload, high quality reconstruction, and robustness under JP3D compression attacks.

This paper is organized as follows. In section 2, we describe our data hiding/compression scheme. Simulation results are given in section 3 to show the performance of the proposed scheme. And, finally, conclusions are drawn in section 4.

2. DATA HIDING IN JP3D

To date, most data hiding solutions, in medical applications, focus on single frame images. In our previous work (Ouled Zaid et al.-2009), the Turbo TCQ (Trellis Coded Quantization) approach was investigated to design a medical multiframe image watermarking scheme, used in conjunction with JP3D compression algorithm. Based on the obtained results, the TTCQ watermarking approach, integrated to JP3D coding scheme is promising. However, it heavily increases the computational cost of the global coding scheme, which is not efficient when real time application is required. Here, we propose to reduce the computational cost while using the same coding architecture. A modified version of QIM based embedding is used in place of the Turbo TCQ method. The watermark embedding is performed on-the-fly during image compression. The computational cost to derive the transform domain a second time for watermarking purposes can therefore be saved. This system is illustrated in a block diagram form as shown in Fig. 1. The embedding/detection algorithms will be described throughout the next subsections.

2.1 Embedding Process

As cited earlier, the global embedding process is applied on three dimensional discrete wavelet transform (3D DWT) domain. To overcome low-pass filtering and lossy JP3D compression attacks, the embedding is only performed on wavelet coefficients contained in the 3D lowest frequency subband. To increase the watermark robustness, lowest frequency subband is divided into equally length sequences. Each sequence is independently quantized to encode one bit of the message to embed. The embedding process uses a sequence of quantizer values. The quantizer sequence length depends on the message length. Therefore, a component, from the message to be embedded, acts as an index which is assigned to a fixed quantizer value. Due to the high sensitivity of the lowest frequency wavelet coefficients to quantization losses, the embedding quantizer step size is finely selected. It should be noted that the coded coefficients can be selected randomly by the user and keep their location as a secret key. The watermark message corresponds to a binary sequence assigned by $m = \{b_1, \dots, b_i, \dots, b_n\}$, with $b_i \in \{0,1\}$. A binary decision is then carried out depending on the quantizer value Δ . To lessen image degradation caused by embedding, we refine the selection of reconstructed

watermarked wavelet coefficients by using a fixed scaling parameter, α . For a given wavelet coefficient y , when $b_i = 0$ is considered, the binary decision results on the component c_0 , given by:

$$\begin{array}{lll}
 \text{if} & y \in \left[\tilde{y} + \alpha, \tilde{y} + \frac{\Delta}{2} - \alpha \right] & \text{then} & c_0 = y; \\
 \text{if} & y \in \left[\tilde{y}, \tilde{y} + \alpha \right] & \text{then} & c_0 = \tilde{y} + \alpha; \\
 \text{if} & y > \tilde{y} + \frac{\Delta}{2} - \alpha & \text{then} & c_0 = \tilde{y} + \frac{\Delta}{2} - \alpha,
 \end{array} \tag{1}$$

with $\tilde{y} = \left\lfloor \frac{y}{\Delta} \right\rfloor \times \Delta$, and $\lfloor x \rfloor$, the closest integer to x Otherwise, if $b_i = 1$ is considered, c_1 is given by:

$$\begin{array}{lll}
 \text{if} & y \in \left[\tilde{y} + \frac{\Delta}{2}, \tilde{y} + \frac{\Delta}{2} + \alpha \right] & \text{then} & c_1 = \tilde{y} + \frac{\Delta}{2} + \alpha; \\
 \text{if} & y \in \left[\tilde{y} + \frac{\Delta}{2} + \alpha, \tilde{y} + \Delta - \alpha \right] & \text{then} & c_1 = y; \\
 \text{if} & y > \tilde{y} + \Delta - \alpha & \text{then} & c_1 = \tilde{y} + \Delta - \alpha.
 \end{array} \tag{2}$$

2.2 Recovery Process

The recovery stage can directly be performed after the dequantization during the decompression phase, without going through all the decoding cycle. The reconstructed wavelet coefficients \hat{y} located in the coarsest volumetric subband, are manipulated using Equation 3 to obtain Q_0 and Q_1 indices.

$$Q_0 = \begin{cases} \left\lfloor \frac{\hat{y}}{\Delta} \right\rfloor & \text{if } \hat{y} \geq 0 \\ \left\lfloor \frac{\hat{y}}{\Delta} \right\rfloor & \text{otherwise} \end{cases} \quad \text{and} \quad Q_1 = \begin{cases} \left\lfloor \frac{\hat{y} + \frac{\Delta}{2}}{\Delta} \right\rfloor & \text{if } \hat{y} \geq 0 \\ \left\lfloor \frac{-\hat{y} - \frac{\Delta}{2}}{\Delta} \right\rfloor & \text{otherwise} \end{cases} \tag{3}$$

At this step, we measure the distance between \hat{y} and its approximation, as follows:

$$\begin{array}{lll}
 \text{If} & |\hat{y} - (Q_1 \times \Delta)| < |\hat{y} - (Q_0 \times \Delta)| & \text{then} & b_i = 1 \\
 & & \text{else} & b_i = 0
 \end{array}$$

This recovery process can be achieved, either, after the dequantization in the decoder cycle, or, whenever we apply the wavelet transform to the reconstructed image.

3. EXPERIMENTAL RESULTS

To examine the effectiveness of the proposed data hiding algorithm, we coupled it with JP3D wavelet-based coder. We conducted our experiments with OpenJpeg implementation of the JP3D coding and we used the default coding parameters for our experiments.

Initially, we start by analyzing the imperceptibility results. Secondly, we study the robustness against JP3D compression attacks. In the simulation, 50 8-bit CT and MR sequences were processed. Here, we only show the results obtained for two multiframe images. The first one, KNEE, is from a Magnetic Resonance Imaging (MRI) modality. The second one, CTBRAIN, is from a Computed Tomography (CT) modality. Note that the tested images vary in size: $400 \times 400 \times 250$ for the MRKNEE scan and $512 \times 512 \times 83$ for CTBRAIN. It should be noted that the capacity payload adequacy depends on the content information in the lowest frequency subband of the transformed image. To model the perceptual effect of image changes, the peak-signal-to-noise ratio (PSNR) quality metrics was used in the experiments.

3.1 Imperceptibility Results

Fig. 2a and 2b illustrate the PSNRs using our data hiding/compression scheme and conventional JP3D coder, over a large range of bitrates. The size of the message to embed was fixed to 50 000 bits.

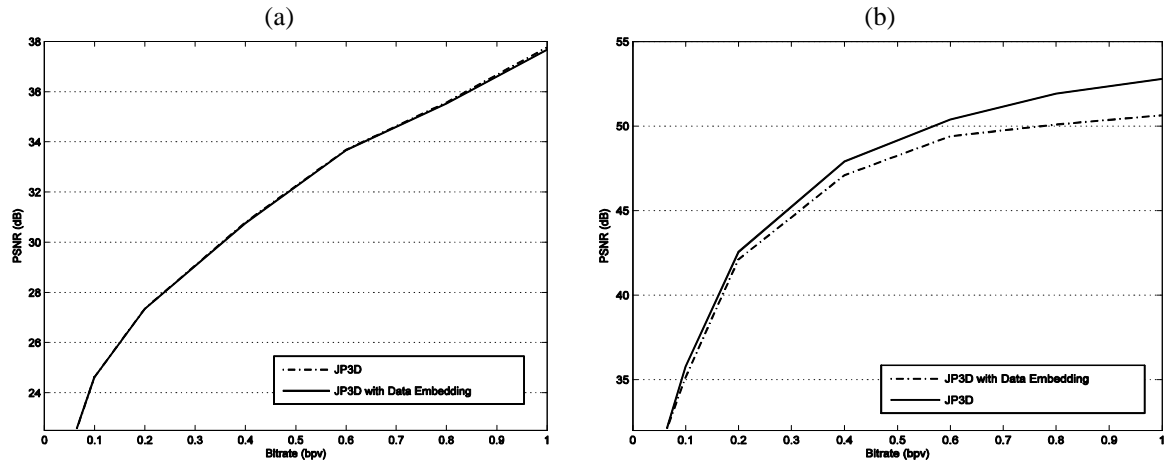


Figure 2. PSNR variation at different bitrate values, using JP3D coder with and without data hiding for, (a) MRKNEE scan, (b) CTBRAIN scan.

Table 1. PSNR results of embedding binary payloads with varying capacities into medical scans.

	MRKNEE	CTBRAIN
NEB	PSNR (dB)	PSNR (dB)
15 000	60.43	53.96
20 000	59.17	53.10
30000	57.44	51.60
40 000	56.31	50.59
50 000	55.21	49.67
100 000	52.04	46.89

Table 2. Percentage of hidden data recovery under JP3Dcompression attacks.

Bitrate (bpv)	MRKNEE	CTBRAIN
0.065	< 50%	< 50%
0.1	50%	76%
0.2	100%	100%
0.4	100%	100%
0.6	100%	100%
0.8	100%	100%

From the results reported in these figures, we can notice that the quality degradation due to the data embedding varies with the compression ratio. At low bitrates, our data hiding/compression scheme gives comparative results, in terms of PSNR, with those obtained by JP3D coder alone. For instance, when using our method at bitrate equal to 0,2 bpv (bits per voxel), the PSNR decrease is about 0 dB for MRKNEE scan, and 0,68 dB for CTBRAIN scan. However, for higher bitrates, the quality degradation, in terms of PSNR, induced by the embedding, is quite observable. For example, at bitrate equal to 1 bpv, the PSNR decrease is about 0,02 dB for MRKNEE image, and 2,15 dB for CTBRAIN image. As depicted in Fig. 2a and 2b, for MR dataset, the PSNR decrease, due to the embedding, is higher for CT dataset. This is principally due to the fact that the tested CT scan is characterized by homogenous regions which are sensitive to quantization based embedding. Consequently, high contrast cover images have more resistance than low contrast ones. The quality assessment of an image after data embedding was also carried out to measure the amount of distortion due to data hiding. Table 1 depicts the Number of Embedded Bits (NEB) versus PSNR measures. Lossless JP3D compression mode has been enabled to measure only the amount of distortion due to data hiding. From

the results reported in Table 1, it can be seen that, for all the tested Number of Embedded Bits (NEB), CT scan provides the highest degree of imperceptibility, in terms of PSNR, compared to MR dataset.

3.2 Robustness against JP3D Lossy Compression

To evaluate system performance on robustness, we demonstrate the robustness against lossy JP3D compression attacks. The experiment was repeated for different Bitrate values, and the percentage of the binary data recovery has been computed. As seen in Table 2, the percentage of the data recovery is slightly influenced by the bitrate values. We can notice that for the aforementioned tested scans, the embedding scheme is entirely robust against JP3D compression attacks with bitrates upper than 0.2 bpv. It is worth outlining that at bitrates lower than 0.2 bpv, the compression process induces bad visual effects that are intolerable in clinical practice. Consequently, the data recovery will be no more necessary.

Due to the nature of content information in MR and CT datasets, direct comparison with other data hiding/compression systems, applied on bidimensional images, is inaccurate. Compared to related work, adapted to multiframe medical sequences (Ouled Zaid et al.-2009), our hybrid data hiding/compression scheme exhibits larger data payload and higher robustness to compression attacks. It remains robust for JP3D compression bitrates upper than 0.2 bpv. Whereas, A. Ouled Zaid scheme (Ouled Zaid et al.-2009) is robust for JP3D compression bitrates upper than 0.4 bpv, in the case of CT dataset, and 0.8 bpv in the case of MR dataset.

4. CONCLUSION

In this research, the issue of integrated data hiding and JP3D image compression is investigated. With the introduction of non-linear scaling in the QIM based embedding process, image degradation is very slight even though a large amount of data is embedded. The extraction of hidden data can be performed in the decoder cycle, or, whenever we apply the 3D DWT to the reconstructed volumetric image. Experimental results have also shown the robustness of our data embedding algorithm against JP3D compression attacks for low bitrates. Future work will deal with the automatic selection of the scaling parameter, so that the best tradeoff between robustness and minimum quality degradation can automatically be achieved.

REFERENCES

- Campisi, P., 2005, Video watermarking in the 3D-DWT domain using quantization-based methods. *IEEE 7th Workshop on Multimedia Signal*. pp. 1–4.
- Cheng, Y., Arvin Chiang, F., and Shen, J. H., 2008. ROI-based Watermarking Scheme for JPEG 2000. *Journal of Circuits Syst Signal Process*, vol. 27, pp. 763–774.
- Giakoumaki, A., Pavlopoulos, S., and Koutsouris, D., 2006. Multiple Image Watermarking Applied to Health Information Management. *IEEE Trans. on Technology in Biomedicine*, vol. 10, pp. 722–732.
- Li, Q., and I. Cox, J., 2007, Improved spread transform dither modulation using a perceptual model: robustness to amplitude scaling and JPEG compression. *Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*.
- Ouled Zaid, A., Makhloufi, A., and Olivier, C., 2009. JP3D Compressed-Domain Watermarking of Still and Volumetric Medical Images. *Signal Image and Video Processing Journal, Springer Verlag (London)*, vol. 4, pp.11–21.
- Kobayashi, L. O. M., and Furuie, S. S., 2008. Proposal for DICOM Multiframe Medical Image Integrity and Authenticity. *Journal of Digital Imaging*, vol. 22, pp. 71–83.
- Schelkens, P., Tzannes, A., and Brislawn, C., 2006, JPEG 2000 Part 10 Volumetric Data Encoding. *IEEE ISCAS Conference*. pp. 3874–3877.

GAUSSIAN MIXTURE BACKGROUND MODEL WITH SHADOW INFORMATION

Jung-Ming Wang¹, Sei-Wang Chen², and Chiou-Shann Fuh¹

¹*Department of Computer Science and Information Engineering, National Taiwan University, Taiwan*

²*Department of Computer Science and Information Engineering, National Taiwan Normal University, Taiwan*

ABSTRACT

In this paper, we integrate shadow information into the background model of a scene in an attempt to detect both shadows and foreground objects at a time. Since shadows accompanying foreground objects are viewed as parts of the foreground objects, shadows will be extracted as well during foreground object detection. Shadows can distort object shapes and may connect multiple objects into one object. On the other hand, shadows tell the directions of light sources. In other words, shadows can be advantageous as well as disadvantageous. To begin, we use an adaptive Gaussian mixture model to describe the background of a scene. Based on this preliminary background model, we extract foreground objects and their accompanying shadows. Shadows are next separated from foreground objects through a series of intensity and color analyses. The characteristics of shadows are finally determined with the principal component analysis method and are embedded as an additional Gaussian in the background model. Experimental results demonstrated the feasibility of the proposed background model.

KEYWORDS

Dynamic scene, Adaptive Gaussian Mixture Model, Foreground detection, Shadow detection

1. INTRODUCTION

Detecting foreground object from a video sequence plays an important role in video understanding. Foreground objects can be considered as those objects we are interested in the scene, which could be moving objects, static large objects, text, and human face [2]. Four techniques have commonly been employed [1]: temporal differencing, motion-based detection [6], model matching, and background subtraction [8].

No matter which method is applied for extracting foreground objects, we often suffer the problem of shadow. Many techniques [3][5][6] have been proposed for removing shadows from images. In the method proposed by Scanlan, et al.[6], the input image is first divided into blocks. For each block, the mean intensity value of the block is computed. Next, from the mean values of all blocks the median of the mean values is determined. Thereafter, for each block a local ratio is calculated by dividing the median with the local mean value of the block. The pixel values of the block are then multiplied by the ratio. If there are shadows existing in the block, they will be suppressed after the multiplication.

In Fathy and Siyal's work [5], shadow models were constructed beforehand from motion figures extracted from video sequences. Constructed shadow models were later used as references for recovering objects from the figures detected in the input video sequences. The major weakness of this method is that objects with colors close to black are easily regarded as shadows. In [3], shadow candidates were first located due to the postulation that shadows usually correspond to dark image regions. Having obtained shadow candidates, their orientations were estimated. In order to discriminate between cast and self-shadows, the authors further assumed that cast shadows contain fewer edges than self-shadows. According to this assumption, the authors searched for the boundaries between cast and self shadows along the estimated orientations of shadows. Clearly, this method will be fail when cast shadows contain significant numbers of edges.

Shadows can be broadly divided into cast and self-shadows. The self-shadow is a part of object, which is not illuminated by the light source. The cast shadow lying beside the object belongs to the background. For object recognition and many other applications, cast shadows are undesired and would be eliminated, while self-shadows are parts of the object and should be preserved. However, cast- and self-shadows are close in

intensity. How to distinguish between them is always a challenge. Moreover, if objects have intensities similar to those of shadows, shadow removal could become extremely difficult. Even though objects and shadows can be separated, object shapes are often incomplete.

The cast shadow sometimes consists of umbra and penumbra. The umbra locates on the inner region of the shadow, where the light source is completely occluded by the object. The penumbra locates on the boundary of the shadow, where partial light source is occluded. For a point light source, only the umbra is cast. Since the point light source does not exist in the natural scene, the penumbra should be concerned in our application.

Sunlight often is the main light source in our scene. After it illuminates to the ground surface, some light will be diffused as the skylight. Given an object surface with constant reflectance, the observed surface intensity I can be formulated as the combination of the sunlight reflection I_1 and skylight reflection I_2 :

$$I = I_1 + I_2 = \int K_1 L_1(\lambda) f d\lambda + \int K_2 L_2(\lambda) d\lambda \quad (1)$$

where K_1 and K_2 is the reflectance coefficient to the sunlight and skylight respectively, and f is the geometric term relating to the incident angle of the illumination, angle for viewing direction, and angle for specula reflection [10]. The object surface and body do not influence the skylight, so there is no geometrical function in the second term.

In the shadow region, L_1 is occluded partially. Let we introduce a scale term β for occluded sunlight and formulate the shadow model I_s as:

$$I_s = \int K_1 \beta L_1(\lambda) f d\lambda + \int K_2 L_2(\lambda) d\lambda = \beta I_1 + I_2 \quad (2)$$

where β is zero in umbra and has the value between 0 and 1 in the penumbra. Equation (2) shows that the shadow intensity is linear combination of the skylight and sunlight reflections with slope value β . After subtracting I_2 from I , we can have I_1 that is the object's body color if $L_1(\lambda)$ is white light (sun light).

In this paper, we propose a method for generating background model embedded shadow information. After subtracting the background from the current image, the foreground object without shadow can be obtained. The paper is organized as follows. In Section 2, the system flowchart is described. The background model and shadow detection are presented in Section 3 and Section 4, respectively. Finally, we have experimental results and conclusions.

2. FLOWCHART

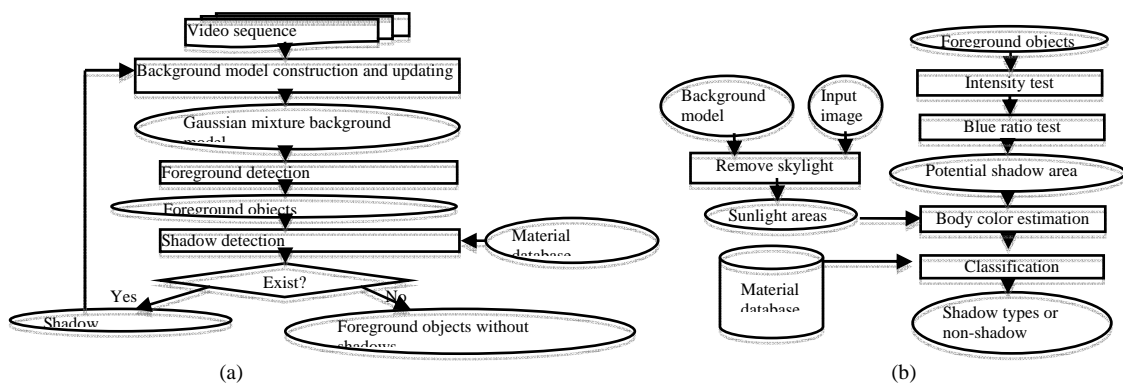


Figure 1. The flowcharts of the (a) foreground object detection and (b) shadow detection.

In this paper, we design a method for generating the background and shadow model using the static camera. Fig. 1(a) shows the flowchart. At first, general background model using mixture of Gaussian is constructed and updated. This construction helps us to obtain the Gaussian mixture background model as that doing in [8]. After we have the background model, foreground object can be detected using background subtraction method. If the shadow information has been embedded into the background model, we will have the foreground objects without shadow. However, if the foreground object is connected with shadow, that will be detected and then we can provide the information for background model generation.

In the region of the foreground objects, shadows are detected using the following tests: intensity, blue ratio, and reflectance tests. Among them, intensity and blue tests are based on the attributes of the shadow color, while reflectance test is to compare the regions with the known material so that we can separate the cast shadow from the self-shadow. The details of the shadow detection will be addressed in Section 4. After having the shadow information, the background model updating will prevent to remove the shadow feature even that is not appeared so often as the other background features.

3. GAUSSIAN MIXTURE BACKGROUND MODEL

Let $X = \{X_1, \dots, X_t\}$ be the visual feature values of a certain pixel observed over t previous frames, and $P(X_t)$ be the appearance probability of X_t . In Gaussian mixture model, $P(X_t)$ is approximated by:

$$P(X_t) = \sum_{i=1}^K \omega_{i,t} \times \eta(X_t, \mu_{i,t}, C_{i,t})$$

where K is the number of Gaussian distribution function η with mean $\mu_{i,t}$ and covariance matrix $C_{i,t}$, and $\omega_{i,t}$ is the weight of the corresponding Gaussian distribution function. In our application, X_t is set as a 3-dimensional color feature vector in RGB color space, and the covariance matrix is set as:

$$C_{k,t} = \sigma_{k,t}^2 I$$

in the assumption of the independent color channels.

Given an input feature vector X_t , we search for the matched η function between the K functions. Here, we say that the i th η function is matched if the distance $|X_t - \mu_{i,t}|$ smaller than $2.5 * \sigma_{i,t}$ as that defined in [9], and the corresponding parameters are updated using the following equations with learning rate α :

$$\begin{aligned} \omega_{i,t} &= (1 - \alpha)\omega_{i,t-1} + \alpha \\ \mu_{i,t} &= (1 - \rho)\mu_{i,t-1} + \rho X_t \\ \sigma_{i,t}^2 &= (1 - \rho)\sigma_{i,t-1}^2 + \rho(X_t - \mu_{i,t})^T (X_t - \mu_{i,t}), \end{aligned} \quad (3)$$

where $\rho = \alpha \eta(X_t, \mu_{i,t-1}, C_{i,t-1})$. For the other unmatched η functions, their corresponding weights $\omega_{k,t}$ are updated using

$$\omega_{k,t} = (1 - \alpha)\omega_{k,t-1}. \quad (4)$$

If there is no η functions matched (it is also happened in system beginning), a new η function with mean value X_t and initial parameter values (σ^2 and ω) is given, and K is increased with 1. Finally, the first b η functions with the summation of their weights greater than a threshold are chosen for combining P of the background model. We denote P of pixel (x, y) as $P^{x,y}$.

Shadow, however, appear so rarely that it's intensity value cannot be preserved in the above computation. To embed the shadow information into the background model, we must know which η function is initialized because of shadow pixel values, and keep it from weight being decreased by Equations (4). Besides, we must initial a η function for shadow value according the background model, so that we can handle the shadow in the following video sequence. For the survey of the mixture of background model, please refer to [8].

4. SHADOW DETECTION

We apply Nadimi and Bhanu's method [10] to locate the shadow region. Fig. 1(b) shows the flowchart. In the beginning, the foreground objects, F , are extracted by comparing the current frame (Fig. 2(a)) with the Gaussian mixture background model (Fig. 2(b)):

$$F(x, y) = \begin{cases} I(x, y) & \text{if } P^{x,y}(I(x, y)) > T \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

where T is a threshold predefined. Since the shadow should have lower intensity value than the background, we test the pixels in the foreground objects with the corresponding pixels in the background, and construct the darker region (Fig. 1(c)), D , after intensity test:

$$D(x, y) = \begin{cases} I(x, y) & \text{if } I(x, y) < \mu_i^{x,y} \\ 0 & \text{otherwise} \end{cases},$$

where $\mu_i^{x,y}$ is the mean value of the η function in $P^{x,y}$. We may say $I(x,y) < \mu_i^{x,y}$ if the intensity values in RGB channels are all smaller than $\mu_i^{x,y}$.

The skylight is supposed to tend to blue color, so the intensity value of blue color channel will have fewer decreasing in shadow. Blue ratio test is designed for testing the pixels, DB , with fewer blue color decreasing:

$$DB(x,y) = \begin{cases} I(x,y) & \text{if } [(I_b(x,y)/\mu_{i,b}^{x,y}) > (I_r(x,y)/\mu_{i,r}^{x,y})] \& [(I_b(x,y)/\mu_{i,b}^{x,y}) > (I_g(x,y)/\mu_{i,g}^{x,y})] \\ 0 & \text{otherwise} \end{cases}$$

where I_c and μ_c mean their color value in c channel. Fig. 2(d) shows the testing result. The corresponding background pixels are also extracted as DF (Fig. 2(e)):

$$DF(x,y) = \begin{cases} \mu_i^{x,y} & \text{if } I(x,y) > 0 \\ 0 & \text{otherwise} \end{cases}$$

In the potential shadow area, the cast-shadow and self-shadow have similar attributes except for the surface type. We can locate the surface types according to their reflectance feature. As mentioned in Section 2, we can remove the skylight effect from the input image to estimate body color: $BC = DF - DB$. The neighbor pixels P_a and P_b in BC are clustered as the same surface if they have a smaller reflectance ratio:

$$H = (I_a - I_b)/(I_a + I_b) = (\rho_a - \rho_b)/(\rho_a + \rho_b),$$

where I_a and I_b are the intensity values of P_a and P_b , and ρ_a and ρ_b are the albedo (reflectance) value of P_a and P_b . Fig. 2(f) shows the clustering result.

The pixel values of each cluster in BC are averaged to compute the body color of that region. The body colors of some ground surfaces (road, way mark, concrete, and grassland) also are estimated beforehand as our body color database. Let V_p as the color vector of one region and V_t as the color vector of the material surfaces in the database, their similarity is computed as

$$d(V_p, V_t) = (V_p \cdot V_t) / \|V_p\| \|V_t\| \quad (6)$$

When they have larger similarity value, this region belongs to cast-shadow, otherwise, it would be the self-shadow and can be eliminated. Fig. 2(g) is the cast-shadow of the detection result.

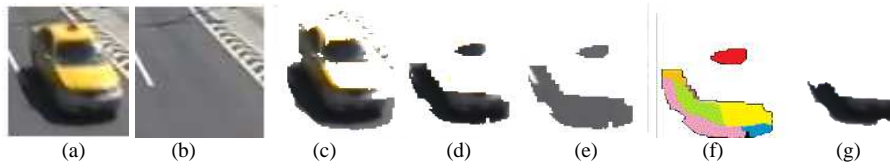


Figure 2. (a) Current image, (b) background image, (c) dark region, (d) after blue ratio test, (e) the corresponding region in background, (f) segmentation result, (g) cast shadow.

5. SHADOW MODEL UPDATING

In the Gaussian mixture model (Equation (3)), most of the weights except for the matched η function are decreased according to Equation (4). After having the shadow color value, we can locate the corresponding η function and stop to decrease its weight. Other pixels in the same material region with the detected shadow region are added η function for shadow. To locate the material region, we segment the background image so that the pixels with similar feature values are collected in the same region.

The segmentation method begins by dividing the input image I into four sub-images r_i ($i=1, 2, 3, 4$) equally on the horizontal and vertical directions; their attribute vectors θ_i are calculated separately. For each sub-image r_i , let $\theta_i = (A_i, T_i)$, where A_i is the average of chromatic characteristics of pixels in r_i and T_i is the average of texture characteristics of pixels in r_i . We transform the pixel color into another domain to calculate the chromatic characteristics that are the most stable and suitable for the vision application. The texture characteristics are estimated using steerable filters that are basis filters for texture representation [11].

The attribute vector is included of the chromatic and the texture characteristics. The chromatic characteristic is computed by the average of the pixel values in RGB channel. The texture characteristic is computed by the average of the gradient values in the sub-image. We can convolute the sub-image with the θ

direction derivative of Gaussian, G^θ , in all direction. The texture characteristic T_i is defined as the θ where we have the maximum average convolution values between sub-image r_i and G^θ :

$$T_i = \operatorname{argmax} \sum r_i * G^\theta$$

However, testing all of the θ value to obtain T_i is not efficient. Fortunately, it has been proved that the G^θ filter is a steerable filter [35], so that it can be synthesized by taking a linear combination of several basis filters: $G^\theta = \cos(\theta)G^{0^\circ} + \sin(\theta)G^{90^\circ}$, where G^{0° and G^{90° are the basis filters. Because the convolution is a linear operation, we can synthesize a convoluted image $R^\theta = r * G^\theta$ using R^{0° and R^{90° : $R^\theta = \cos(\theta)R^{0^\circ} + \sin(\theta)R^{90^\circ}$. To obtain the maximum average value of R^θ , the corresponding θ value, θ_d , can be computed as $\theta_d = \tan^{-1}(\frac{R^{90^\circ}}{R^{0^\circ}})$.

Comparing the attribute vector Θ_i , if all Θ_i , $i=1..4$, have the similar values, we can say that the original sub-image is homogeneous, and we do not need to divide that. Or the sub-image will be divided into more to obtain more sub-images until the sub-image is tested as being homogeneous or the sub-image size is small than a threshold. To prevent merging small objects into the background, we divide the image into 6×5 sub-images as our initialization. The merged result of Fig. 3(a) is shown in Fig. 3(d).

For each pixel (x, y) in the shadow region, we apply updating algorithm proposed in Section 3. Besides, the matched or the initialization η function is denoted as η_b function and we give that a special handling. Because η_b function could not be matched usually, its weight will be set as a static value without applying Equation (4). For other pixels (x', y') in the same segmentation of the pixel (x, y) , we also update $P^{x', y'}$ as the above processing for $P^{x, y}$. Our shadow updating helps to construct the shadow information in the background model even no shadow appear in all place.

In the Gaussian mixture background model for the ground surface, we have at least two η functions for one pixel: one for shadow (η_b function) and another one for non-shadow. Given its mean values, μ_i and μ_s , the body color vector of the pixel can be defined as $V_b = \mu_i - \mu_s$ (in the form of a 3-dimensional vector). When the corresponding pixel of the current image is determined as foreground (Equation (5)), the μ_s must be subtracted from the pixel value as computing its body color V'_b . If $d(V_b, V'_b)$ is smaller than a threshold, the current pixel value will be considered as penumbra value.

6. EXPERIMENT

We test our process on several video sequences, which has asphalt road, concrete floor, and grassland. Fig. 3 shows the background generation result. The pixel values are defined as the mean value of the η function with largest weight value. However, each pixel may have more than one mean value to represent its background value. For example, we may have two values to represent the background value on the waving three and on the shadow region.

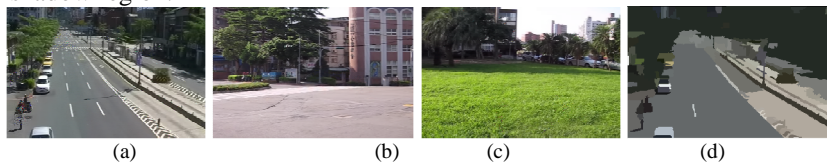


Figure 3. The test videos: (a) asphalt road, (b) concrete floor, (c) grassland, and (d) segmentation result of (a).

Fig. 4 shows the foreground object detection with shadow and without shadow. When we subtract the ordinary background model from the current image, the shadow would be casted beside the foreground object. Using our proposed Gaussian mixture background model with shadow information, we can subtract the shadow at the same time.

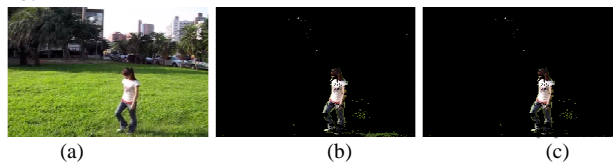


Figure 4. Foreground object extraction (b) with shadow and (c) without shadow.

Fig. 5 shows the foreground objects extracted using our background model. Those objects may be broken because some parts of the object are similar to the background or the shadow model. This problem is appeared in background subtraction method. We can apply some morphological operations or enclosing method as that proposed in [4] to obtain a whole objects.

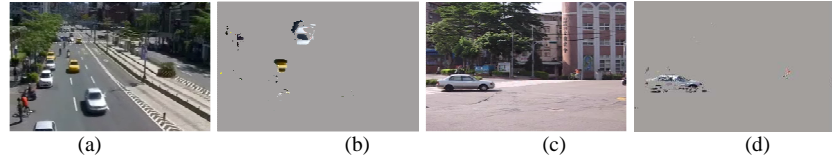


Figure 5. Foreground object extraction (b)(d) with shadow removal from the current image (a)(c).

7. CONCLUSION

Shadow effect is an important issue in background subtraction method and can significantly influence the performance of an application system. The proposed technique requires no particular information about camera installations, and can automatically examine the shadow information to embed into the background model. Based on the segmentation result, we can embed shadow model in the whole ground surface even no shadow appear in the current time. That process helps us to handle the shadow problem in the following time.

The proposed method has the assumption of white illumination in the scene. However, it may be yellow illumination in the nightfall or blue illumination in the cloudy day. We don't need to worry about the second case because there is no shadow in the cloudy day. The current technique has to be improved to be able to apply to the nightfall case.

REFERENCES

- [1] J. M. Wang, S. Cherng, C. S. Fuh, and S. W. Chen, "Foreground Object Detection Using Two Successive Images," *Int'l Conf. on Advanced Video and Signal Based Surveillance*, pp. 301-306, Santa Fe, 2008.
- [2] M. Colin, *Cognitive Psychology: A Neural-Network Approach*, Brooks/Cole Publishing Company, California 1991.
- [3] C.J. Chang, W.F. Hu, J.W. Hsieh, and Y.S. Chen, "An Automatic Traffic Surveillance System for Tracking and Classifying Vehicles", *Proc. of 15th IPPR Conf. on Computer Vision, Graphics and Image Processing*, pp. 382-387, 2002.
- [4] Y.C. Chung, J.M. Wang, and S.W. Chen, "Progressive background images generation", *The 15th IPPR Conference on Computer Vision, Graphics and Image Processing*, Hsinchu, Taiwan, 858-866, Aug 2002.
- [5] M. Fathy and M.Y. Siyal, "An Image Detection Technique Based on Morphological Edge Detection and Background Differencing for Real-Time Traffic Analysis", *Pattern Recognition*, Vol. 16, pp. 1321-1330, 1995.
- [6] J.M. Scanlan, D.M. Chabries and R.W. Christiansen, "A Shadow Detection and Removal Algorithm for 2D Images", *Proc. of Int'l Conf. on Acoustics, Speech, and Signal Processing*, pp. 2057-2060, 1990.
- [7] L. G. Shapiro and G. C. Stockman, *Computer Vision*, Upper Saddle River, NJ: Prentice Hall.
- [8] T. Bouwmans, F. E. Baf, B. Vachon, "Background modeling using mixture of Gaussians for foreground detection-a survey," *Recent Patents on Computer Science*, vol. 1, pp. 1-19, 2008.
- [9] C. Stauffer, W. Grimson, "Adaptive background mixture models for real-time tracking," *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol.2, pp. 246-252, 1999.
- [10] S. Nadimi and B. Bhanu, "Physical models for moving shadow and object detection in video," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, pp. 1079-1087, 2004.
- [11] Y. C. Chung, S. Cherng, R. R. Bailey, and S. W. Chen, "Intrinsic image Extraction from a Single Image," *Journal of Information Science and Engineering*, vol. 25, no. 6, pp. 1939-1953, Nov. 2009

A SIMILARITY MEASURE BASED ON GEOMETRIC CONSISTENCY FOR IMAGE RETRIEVAL USING LOCAL DESCRIPTORS

A. Najjar and E. Zagrouba

RIADI Laboratory, University LaManouba

Team of research SIIVA, Institut Suprieur d'Informatique, Ariana, Tunisia

ABSTRACT

The content based image retrieval (CBIR) issue can be formulated as a problem of research in the primitive space. These primitives generally describe the similarity at the photometric level. In this work, we focus on the study of geometric consistency to measure the similarity between images in order to build an image retrieval system. In fact, we compute similarity by considering the geometric positions of the points of interest (SIFT). Then, we compare our method to a RANSAC algorithm based method as an alternative to compute geometric consistency. The evaluation of results using two subsets of image (MPEG7 and COIL) shows that the proposed method allows obtaining a better retrieval result.

KEYWORDS

CBIR, Geometric consistency, RANSAC, Similarity measures, SIFT.

1. INTRODUCTION

Content-Based Image Retrieval (CBIR) describes the process of retrieving images based on the image content rather than the metadata such as keywords, tags, and/or descriptions manually associated with the image. The literature shows that CBIR systems use global or local features to describe the image content. In the global approach, the image is characterized by texture, color, and shape attributes. In the local approach points of interest are computed to characterize details and objects. The general scheme of a typical CBIR system is as follows: First, each image on the collection is indexed using global or local features and the computed descriptors are stored. Then, the query image is indexed. Finally, a comparison step between descriptors is performed to obtain the most similar images to the query. CBIR systems based local descriptors are applied for object and scene recognition (Low, 2004) and content-based copy detection (Joly et al, 2005). In such systems, matching the detected feature points is a crucial step. This may be insufficient because it doesn't perform well on a heterogynous image database. To enhance the retrieval result, we consider the geometric distribution of detected feature points. For this, we present a similarity measure based on geometric consistency for image retrieval using local descriptors. This paper is organized as follows: in section 2, the proposed method for image comparison is described. In section 3, the RANSAC algorithm is provided to compare our method for geometric consistency computing. Experiments and evaluation results are presented in sections 4. Finally, some conclusions and perspectives of this work are given.

2. ROPOSED METHOD

The proposed method consists on computing the geometric consistency using the following steps: indexing the query image and database images using local SIFT (Scale-Invariant Feature Transform) descriptors proposed by David G. Lowe (Lowe, 2004), matching the query image with the database images, estimating the data model for each image and finally computing the distance between estimated models. In our work, "geometric consistency" refers to the distribution homogeneity of points of interest among two images. Considering two sets $E1$ and $E2$ of points of interest, detected respectively on two images $I1$ and $I2$, our goal

is to decide if the feature points in $E1$ and $E2$ are dispersed in the same way. So, a similarity measure must be applied between these two sets of points. The problem is relative to the different image properties (size, brightness, zoom, rotation...), therefore the computed properties (location, associated descriptors) on detected features are also different. Our aim is to find a way to measure the consistency without considering these properties. The proposed solution is divided into two stages. The first stage consists to solve the problem of heterogeneity between the descriptors of the points of interest. So, for each detected point in $I1$, the most similar one in $I2$ is determined. This procedure is called "matching". Thus, we eliminate the outliers. In the second stage, a normalization scheme is used to allow the comparison between geometric coordinates. In fact, the similar images can have different sizes, different resolutions, taken from different angles... So, rather than comparing the feature points coordinates, we propose to estimate their relative positions (by estimating the data model for each set $E1$ and $E2$). Then, the similarity between the estimated models is measured.

2.1 Indexing Images using Local Descriptors

In our work, we used local features SIFT to describe the content of images. David G. Lowe presented a framework to detect and describe points of interest. The associated descriptor to each detected SIFT point of interest is a 3D histogram describing the gradient localization and orientation. The localization is quantified with a 4×4 window, while the gradient angle is quantized with 8 directions. The different steps to generate a set of SIFT features are: Scale space Extrema Detection, Keypoint Localization, Orientation Assignment and Keypoint Descriptor Computing. The choice of SIFT to index images is due to two reasons: First, the SIFT are detectors and descriptors in the same time. So, we obtain not only the feature point localizations but also a representation of their characteristics. Next, SIFT has several interesting properties. They are invariant to the scale factor, partially invariant to the changes of luminance, to the changes of the camera perspective and they are well localized in both spatial and frequency domains. This reduces the effects of occlusion and noise. Finally, the primitive SIFT are repeatable. Several works showed the effectiveness of SIFT as descriptors, especially for object recognition (Tao and Chi, 2010).

2.2 Point Matching

After the indexing step, we proceed to the matching step. This algorithm has as input two sets of points of interest detected respectively on two images. The aim is to found for each point of interest in the first image (query image), the most similar one in the second image (image from the database). Thus, we discard the outliers and maintain the couples of points of interest having similar descriptors. We will use the selected points for similarity computing in subsection 2.3. David Low (Lowe, 2004) proposed algorithm is used in order to match feature points by choosing a high threshold S . A description of this algorithm is as follows: a couple of points are selected as a valid match if it satisfies two conditions. First, it must be considered as the best match by minimizing the distance between the descriptors of its points. Then, the Euclidean distance between those descriptors must be below by the threshold S to the distance between the descriptors of the second best match. In the case where a couple of points of interest verify the first condition and not the second one, it is rejected and the point from the first image is considered unmatched. To add more reliability to the algorithm described above, the result is refined using a filtering step. This step aims to eliminate the double matching. This situation occurs when two distinct points in the first image are matched with the same point in second one. In this case, we keep the couple that minimizes the Euclidean distance. The asymptotic complexity of matching and filtering steps are $O(N_p \times N'_p)$ and $O(N_C^2)$, respectively. With N_p and N'_p are the numbers of SIFT interest points detected, respectively, on first and second images. N_C is the number of matched couples before the filtering step.

2.3 The Similarity Computing

In our work, similarity measures are applied to compare two probability density functions (pdf) modeling the distribution of SIFT geometric coordinates extracted from two images. Since no general knowledge about the data space is available, we used a nonparametric model for pdf estimation. We opted for the kernel estimator, called also Parzen window (Parzen, 1962). This method is efficient and widely used in many based-image applications such as pattern recognition and classification (Duda and Stork, 2000). To apply the Parzen

window estimator, a kernel function must be selected. So, we applied the most popular one, the Gaussian kernel. The asymptotic complexity of this estimator is $O(N \times N')$ with N is the number of geometric coordinates and N' is the number of random points chosen to perform estimation. We selected five distances to compare probability densities. In the literature, the similarity measures performance was the subject of several studies (Kok'are and Biswas, 2003), (Zhang and Lu, 2003). Two types of measures were chosen: correlations measures such as the correlation ratio and the mutual information (which are quantities to maximize), and divergence measures such as Bhattacharyya, Hausdorff and Kullback-Leibler distances (which are quantities to minimize). The correlation ratio (CR) is a measure of association of non-linear relationships between quantitative variables. It takes the value 1 when all the elements of the population are equal and it is equal to 0 when there is dispersion between elements. The authors of (Liu and Uren, 2008) used the correlation ratio given by the equation (1) as a similarity measure to determine the functional dependence between multimodal images X and Y to solve the image registration problem. The mutual information (MI) (equation 2) of two random variables is a quantity measuring the statistical dependence of these variables. Informally, we say that two variables are independent if the realization of one provides no information on the realization of another. For a couple of images (A, B) , it corresponds to the amount of information that the image B has on the image A and vice versa (Tourassi et al, 2010). The Bhattacharyya distance (Bhattacharyya, 1943) measures the similarity between two discrete probability distributions. For two probability distributions p and q , Bhattacharyya distance is given by equation (3). Kullback-Leibler distance (K-L) (Kullback and Leiber, 1951) is a dissimilarity measure between two probability distributions w_1 and w_0 . This measure (equation 4) is equal to 0 if and only if the distributions $P(. / w_1)$ and $P(. / w_1)$ are equal. The Hausdorff distance (Hangouet, 1995) measures the distance between two subsets X and Y . The definition of this distance is given by equation (5).

Table 1. Distances and correspondent equations

Distance	Equation	
Correlation Ratio	$\eta(Y X) = Var[E(Y/X)]/Var(Y)$	(1)
Mutual Information	$I(A, B) = H(B) - H(B/A) = H(A) - H(A/B)$	(2)
Bhattacharyya	$DB(p, q) = -\ln(BC(p, q))$ with $BC(p, q) = \sum_{x \in X} \sqrt{p(x)q(x)}$	(3)
Kullback-Leibler	$D(w_1 w'_1) = \sum_{w_2} P(w_2/w_1) \log P(w_2/w_1)/P(w_2/w'_1)$	(4)
Hausdorff	$D(X, Y) = \max\{\sup_{y \in Y} \inf_{x \in X} d(x, y), \sup_{x \in X} \inf_{y \in Y} d(x, y)\}$	(5)

The matching step has a great influence on the similarity measures computing. Indeed, not only the quality of the matching which has to be considered, but also the quantity of matched pairs. So, we weighted the used distances by the number of matched pairs N_B . So, we weighted the CR and MI (quantities to maximize) by the number of matched pairs N_B . However, we weighted Bhattacharyya, Hausdorff and K-L distances (quantities to minimize) by the inverse of the number of matched pairs N_B .

3. ESTIMATION USING THE RANSAC ALGORITHM

The experiments on two image collections, presented in section 4, show that K-L distance gives the best result. So, we compare our method by considering this distance, to an existing algorithm called Random Sample Consensus (RANSAC) (Fischler and Bolles, 1981). RANSAC is an iterative method to estimate parameters of a mathematical model from a set of observed data which contains outliers. It does not treat all the sample data at the same time. Iteratively, parameters of the model are reconstructed from a subset of selected data. The remaining data are then tested against the computed model. If a point fits well to the estimated model it will be considered as an inlier. After a fixed number of iterations, the final model is chosen if it maximizes the number of inliers over all computed models. This method is non-deterministic and robust to the presence of outliers. In our case, RANSAC is used to compute an optimal rigid transformation between two sets of SIFT points coordinate. Since the algorithm is non-deterministic, the estimated model may not fit with the original data. So, we deduce a measure of dissimilarity given by the mean of all errors

between the positions of the SIFT points coordinates on the real model and the estimated one. The RANSAC asymptotic complexity is $O(N_i \times N)$ with N_i is the iterations number and N is the size of the sample data.

4. EXPERIMENTS AND EVALUATION

To evaluate our method efficiency, we draw the Recall/Precision curves. Recall is a measure of the ability of a system to present all relevant items. It is computed as the fraction of the number of the retrieved images that are relevant by the number of relevant ones in the collection. While precision is a measure of the ability of a system to present only relevant items. It is the fraction of the number of the retrieved images that are relevant by the total number of retrieval ones. The first evaluation set of image collection is extracted from the MPEG-7 collection. It is composed of 216 images with 18 different shapes and 12 images from every shape. The purpose of using this type of collection is to eliminate the photometric effect and consider only the geometric aspect. The second evaluation set is composed of 140 images extracted from COIL collection. The selected set contains 7 different objects. To draw the precision/recall curves, we made several queries. We focus, then, on the first k images returned by each query. The different points on the curves are computed by varying the parameter k . We define a point coordinates on the curve by computing the mean values of recall and precision, across all queries and for a fixed value of k . Figure 3-a and Figure 3-c present the Recall/Precision curves using different similarity measures, respectively for the COIL and MPEG-7 collections. We note, first, that K-L distance gives the best results and Bhattacharyya gives the worst. Then, CR and MI have comparable results. To evaluate our method, we compare our proposed retrieval scheme using K-L distance to a RANSAC based retrieval scheme. The research result using the queries in Figure 2-a and Figure 2-d is shown respectively in Figures 2-b, 2-c and Figures 2-e, 2-f. As illustrated in these figures, our method gives more pertinent and accurate results than RANSAC algorithm. This has been proved by Recall/Precision curves provided on Figure 3-b and Figure 3-d. This result can be justified by the fact that RANSAC is a parametric-estimator and can't approximate non rigid transformation.

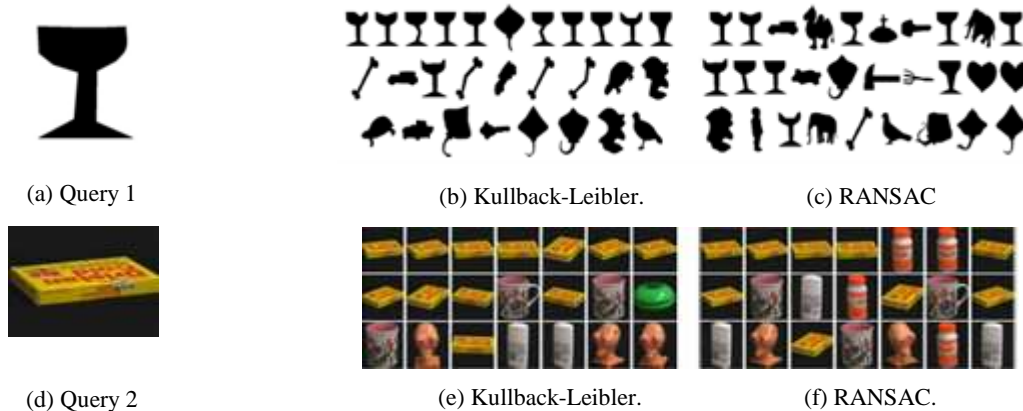


Figure 2. Retrieval result by the query 1(a), MPEG7 Collection (b,c) and by the query 2 (d), COIL Collection (e,f).

5. CONCLUSION

In this paper, we are interested in studying the similarity between the images based on the geometric positions of their detected SIFT point. First, a retrieval scheme is developed by computing geometric consistency using a non parametric estimation. Our method is based on four steps which are: image indexing based on local SIFT descriptors, query image matching with the database images, estimating the data model for each image and computing the distance between estimated models. Second, we compare our method to a RANSAC algorithm based method. So, to be able to compare images using RANSAC, we deduced a measure of dissimilarity given by the average of all errors between elements of the real and the estimated models. Experiments made on MPEG7 and COIL collections proved that using the geometric consistency to

compute similarity for image retrieval gives good results. As perspectives, we will study the quality retrieval result according to the quality of the used descriptors in indexing step.

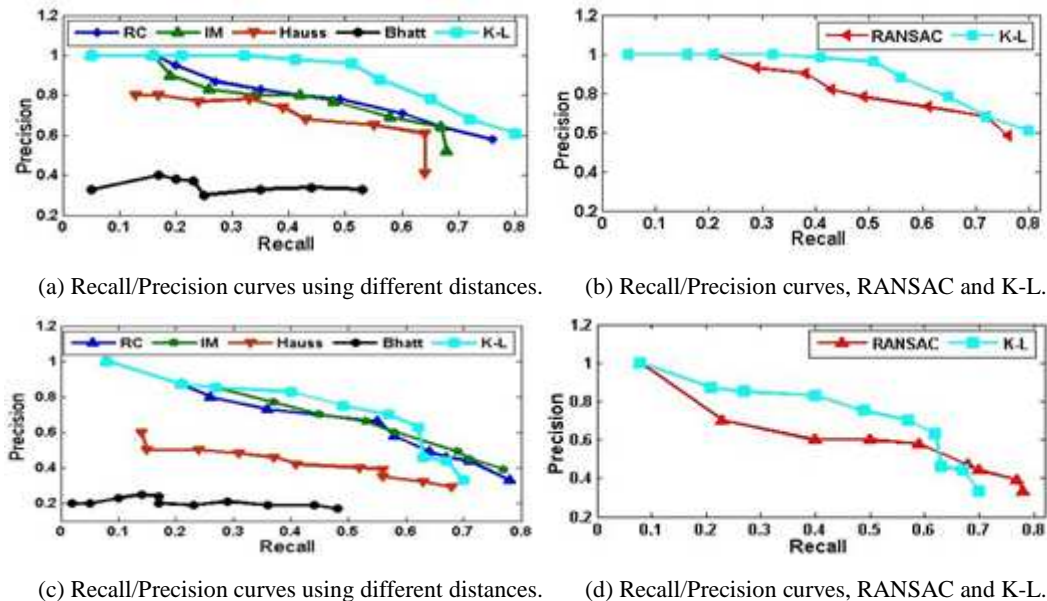


Figure 3. Recall/Precision curves for COIL collection (a-b) and for MPEG7 collection (c-d).

REFERENCES

- Bhattacharyya, A., 1943. On a measure of divergence between two statistical populations defined by their probability distributions. *Bulletin of the Calcutta Mathematical Society*.
- Duda, R., et al, 2000. *Pattern Classification*. Wiley-Interscience, 2nd edition.
- Dorkó, C and Schmid C., 2004. Object class recognition using discriminative local features. *IEEE Transactions on Pattern Analysis and Machine Intelligence*.
- Fischler, A. et al, 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *In Communications of the ACM*, Vol 24(6), pp 381–395.
- Hangouet, J.F, 1995. Computation of the Hausdorff distance between plane vector polylines. *Twelfth International Symposium on Computer-Assisted Cartography*, Vol 4, pp 1-10.
- Joly, A. et al., 2005 Content-based video copy detection in large databases: A local fingerprints statistical similarity search approach. *In Proceedings of the Int. Conf. on Image Processing*.
- Kok'are, M. et al, 2003. Comparison of similarity metrics for texture image retrieval. *In Conference on Convergent Technologies for the Asia- Pacific Region*, Vol 2, pp 571–575.
- Kullback, S. and Leibler, R., 1951. On information and sufficiency. *Annals of Mathematical Statistics*, Vol. 22, pp 79–86.
- Lowe, D. G., 2004. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, pp 99–100.
- Parzen, E., 1962. The estimation of a probability density function and the mode. *The Annals of Mathematical Statistics*, Vol 33, pp 1065–1076.
- Tao, Y. et al, 2010. Performance evaluation of sift-based descriptors for object recognition. *In International MultiConference of Engineers and Computer Scientists*, Vol 2.
- Tourassi G.A., Frederick E.D., Markey M.K., and Floyd C.E, 2001 Application of the Mutual Information Criterion for Feature Selection in Computer-aided Diagnosis. *Medical Physics*, Vol 28.
- Zhang, D. and Lu, G. , 2003. Evaluation of similarity measurement for image retrieval. *In IEEE international conference on Neural Networks and Signal Processing*, Vol 2, pp 928–931.

COMPUTING THE EUCLIDEAN DISTANCE TRANSFORM ON THE EPUMA PARALLEL HARDWARE

Ingemar Ragnemalm and Andréas Karlsson
Dept. of Electrical Engineering, Linköping University

ABSTRACT

The ePUMA architecture is a novel parallel architecture being developed as a platform for low-power computing, typically for embedded or hand-held devices. As part of the exploration of the platform, we have implemented the Euclidean Distance Transform. We outline the ePUMA architecture and describe how the algorithm was implemented.

KEYWORDS

Euclidean distance transform parallel hardware SIMD.

1. INTRODUCTION

The ePUMA (embedded Parallel DSP with Unique Memory Architecture) is a parallel architecture in development. It exists as a cycle-accurate simulator running assembly level programs. It is primarily been developed for embedded systems, combining low power consumption with significant computing power.

As part of the development of the architecture, relevant algorithms are implemented, with the dual goal to prove usability and/or find desirable improvements to the design. The main tracks for such explorations are two, communication infrastructures and media. The latter includes video coding, graphics and image processing. The present work investigates an ePUMA implementation of the Euclidean Distance Transform.

2. EUCLIDEAN DISTANCE TRANSFORM ALGORITHMS

A Distance Transform (DT) is a well known operation proposed by Rosenfeld & Pfaltz (1966, 1968). Given a binary image, it produces the distance to the nearest feature pixel for every pixel. The original operation was based on scalar values. These *scalar* DTs are extensively researched, e.g. by Borgefors (1986).

The Euclidean Distance Transform (EDT) by Danielsson (1980) uses vectors. The sequential EDT occasionally produces insignificant errors. Error-free algorithms have been developed by Rutowitz (1990), Ragnemalm (1990), Paglieroni (1992), Saito & Toriwaki (1994), Eggers (1998) and Lucet (2009).

EDT algorithms can be divided into different classes, including raster-scanning algorithms (Danielsson 1980; Ragnemalm 1989), ordered propagation algorithms (Ragnemalm 1990; Eggers 1998), independent row/column processing algorithms (Paglieroni 1992; Saito & Toriwaki 1994), Minkowski operations (Shih and Mitchell 1992) and Voronoi-based algorithms (Rutowitz 1990; Schouten & van den Broek 2004).

Concerning parallelism, the earliest parallel algorithms were proposed by Danielsson (1980) and Yamada (1984). These algorithms process all pixels in every iteration, wasting computation in unchanging areas.

As shown by Ragnemalm (1989), raster scanning algorithms, including scalar ones, are easy to parallelize. Schneider et. al. (2009) implemented this algorithm on a GPU. Furthermore, other parallel GPU computing algorithms have been presented by Rong et. al. (2006) and Cao et. al. (2010).

3. THE EPUMA ARCHITECTURE

The ePUMA architecture (Liu 2008) is a master-SIMD architecture in development. Its simulator, silicon cost estimation and power cost estimation exist on which exploration work is performed.

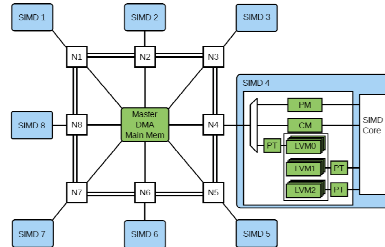


Figure 1. Overview of the ePUMA master-SIMD architecture.

In ePUMA, a master processor controls 8 SIMD processors (Figure 1), each with 8 16-bit data paths for most processing. The SIMD processors and the master are connected by a star-ring DMA network. Computations are pipelined in 12-13 or 9-10 steps for different kinds of operations.

Each SIMD processor holds three *Local Vector Memory* banks (LVM) of 80 kB each (5k vectors with 16 bytes per vector) as fast local memory. Typical LVM handling is illustrated in Figure 2 (left). Two LVMs take turns, one providing input data and space for output, and the other being used for DMA transfers to and from the host, while a third (m0 in the figure) holds persistent data, accumulated output or static input data over many iterations. This scheme overlaps processing with input and output, hiding the cost of memory access.

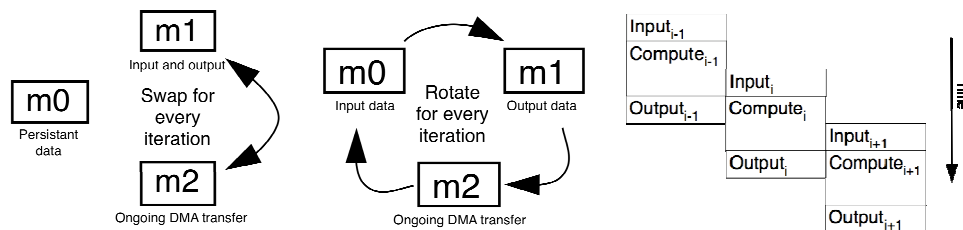


Figure 2. Switching roles of the LVMs, allowing overlapping computations and global memory access.

An alternative approach is shown in the middle of Figure 2, with two LVM's switching roles, is to use a cyclic scheme for all three. For such a case, there should be no persistent data. This is highly suitable for our problem at hand since the output from each iteration (one row/column) is the input to the next.

Image processing is highly relevant for the ePUMA project. Ongoing work include linear operators (such as Sobel operators), nonlinear operators (e.g. median and max) as well as vital components for video codecs (Liu 2008). The task of rendering graphics has also been investigated (Ragnemalm and Liu 2010). However, no prior research on distance transformations has been performed for the platform.

4. RASTER SCANNING EDT FOR SIMD PROCESSING ON THE EPUMA

We have chosen to study raster scanning algorithms, being straightforward, simple, and relatively easy to convert to parallel form. We use the four-scan algorithm proposed by Ragnemalm (1989), more recently implemented on GPUs by Schneider et. al. (2009). We focus on the unsigned 4-scan, 8-neighbor version. Every pixel carries three values, the squared distance d^2 and a two-component vector \mathbf{v} (v_x and v_y). As illustrated by Figure 3, each mask is applied as a separate scan, row by row (top and bottom masks) or column by column (left and right masks). For a sequential algorithm, this means a double "for" loop for each, accessing three neighbors, calculating a candidate value for d^2 and comparing it to the d^2 of the center pixel.

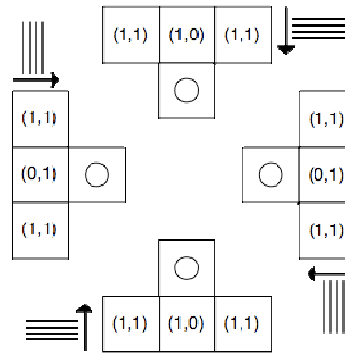


Figure 3. SIMD compatible 4-scan EDT (unsigned version).

All applications of the mask are independent within a row (column) with arbitrary order of operation. This makes the algorithm suitable for SIMD implementation. Each row must be processed in sequence, but the computations within one row can be made in parallel. In pseudo code, a mask is applied like this:

```
for all pixels p in a row
  for each neighbor q on the next row
     $d^2_{\text{candidate}_q} := d^2_q + \text{change due to offset}$ 
    if  $d^2_{\text{candidate}_q} < d^2_p$  then
       $d^2_p := d^2_{\text{candidate}_q}; v_{p\_} = v_q + \text{offset}$ 
```

The d^2 values are calculated incrementally using only additions and shifts, an acceleration trick rarely noted about distance transforms, useful for low-level implementations such as ours. For example:

$$\|(x+1, y+1)\|^2 = (x+1)^2 + (y+1)^2 = x^2 + 2x + 1 + y^2 + 2y + 1 = \|(x, y)\|^2 + 2x + 2y + 2$$

For the ePUMA, we must use local memory, and use all the eight SIMD units. To do this, we upload two rows split into sections, 1/8 row plus one pixel padding, into LVMs. The previous row, which data is read from, should be in one LVM, and the destination, in another, while the third is busy writing out the result of the previous iteration and reading in the next row, which suits the rotating LVM usage of Figure 2 (middle).

Note that after processing a row, the edge pixels must be communicated to the neighbor SIMD. This can be done efficiently with a messaging system built into the system, handling small messages over the ring bus.

The ePUMA is programmed in assembly code. The following instructions are relevant for our purposes:

```
COMPW, CoMPare Words
COPY, copy data and COPY.ULT, Vector COPY if Unsigned Less Than
LSLW, Logical Shift Left Word (for 2*)
ADDW, ADD Word and ADDW.ULT, ADD Word if Unsigned Less Than
```

All operations operate on 8 16-bit integers in parallel, which is what we use in our implementation. Thus, in order to process 8 pixels with a neighbor, we need to (with appropriate addressing):

- ADDW and LSLW to get $d2 + v.x*2 + v.y*2 + 2$ or $d2 + v.y*2 + 1$
- CMPW between $d2$ and the $d2$ of the center pixel.
- COPY.ULT and ADDW.ULT $d2$, $v.x+1$ and $v.y+1$ to the center pixel.

Operations need to be rearranged to avoid pipeline stall. We have performed such rearranging below to get realistic performance. The full code for the algorithm (optimized, for 128 pixels at a time) is as follows:

```
// load 8 d2-values for comparison
copy vr0 m0[ar0+1].v
repeat 16 REPEAT_END // repeat 16 times

// Left pixels
lslwq vacr0 m1[ar0].v 1 // 2x
2 * addwq vacr0 m1[ar1].v vacr0 // 2x + 2y
addwq vacr0 m1[ar2].v vacr0 // d2 + 2x + 2y
addwq vr1/vacr0 2 vacr0 // d2 + 2x + 2y + 2
cmpwq vr0 vacr0 // compare with d2 + 2x + 2y + 2 -> first flags

// Start with middle pixels
lslwq vacr0 m1[ar0+1].v 1 // 2x
addwq vacr0 m1[ar2+1].v vacr0 // d2 + 2x
addwq vr1/vacr0 1 vacr0 // d2 + 2x + 1
cmpwq vr0 vacr0 // compare with d2 + 2x + 1 -> second flags

// First flags done, update: Conditionally copy/add if ult (unsigned less than)
copy.ult m0[ar2].v vr1
```

```

addwq.ult m0[ar0].v m1[ar0] 1
addwq.ult m0[ar1].v m1[ar1] 1

// Right pixels
lslwq vacr0    m1[ar0+2].v 1    // 2x
2 * addwq vacr0    m1[ar1+2].v vacr0 // 2x + 2y
addwq vacr0    m1[ar2+2].v vacr0 // d2 + 2x + 2y
addwq vr1/vacr0 2 vacr0 // d2 + 2x + 2y + 2
cmpwq vr0 vacr0 // compare with d2 + 2x + 2y + 2 -> third flags

// Second flags done -> Update
copy.ult m0[ar2].v vr1
copy.ult m0[ar0].v m1[ar0+1].v
addwq.ult m0[ar1].v m1[ar1+1].v 1
copy vr0 m0[ar0 + 9].v // Load 8 new d2-values for next iteration

// Third flags done -> Update
copy.ult m0[ar2+=8].v vr1
addwq.ult m0[ar0+=8].v m1[ar0+2].v 1
addwq.ult m0[ar1+=8].v m1[ar1+2].v 1

REPEAT_END:
10 * nop

```

This sums to 26 cycles per loop or a total of $16 \cdot 26 + 10 = 426$ cycles. This will process 128 pixels. We will need four similar passes over the image for a complete transform, plus the cost for a square root of 17 cycles for 8 pixels (Tomasson 2010) or 272 cycles for 128 pixels. The cycle cost per pixel will thus be $(426 \cdot 4 + 272) / 128 = 15.4$ cycles per pixel. With 8 SIMD units this gives a total cycle count of $15.4/8 = 1.9$ cycles per pixel! To this should be added a slight overhead for the messaging between row segments and transposing the image once in order to access data row by row even in the vertical passes. The former is negligible, while the latter can be done very efficiently, at data transfer cost with full bandwidth usage, $8/3$ cycles per pixel.

The total processing cost is in good balance with the time consumption for memory transfer. This shows both that the algorithm fits the architecture and that it can be implemented with very good performance.

A limitation in our implementation is the use of 16-bit arithmetics. For the vectors (vx and vy) this is not a problem, but more so for the squared distance value (d2). However, the ePUMA supports saturation arithmetics, where overflowing values will simply result in the maximum value (65535 for unsigned integers). The ePUMA can also work in 32-bit mode, using 4 lanes instead of 8.

5. RESULTS AND FUTURE WORK

Making this implementation, we encountered limitations and thereby desirable enhancements to the design, which is the primary result of this work. In this section, we outline the findings and suggested improvements.

The bit vector field set by e.g. the VCMP instruction is somewhat limited. Currently, it will set a single bit for a scalar comparison, and all 8 bits by a vector comparison. We found that this does not properly support mixing of 16- and 32-bit arithmetics. A particularly important addition is the ability to allow the combination of two 4-element vector comparisons to be applied on 8-element vector copying. Then we could work with the d2 array in 32 bits per element but 16-bits for vx and vy, allowing 32-bit precision where needed but still enabling full 8-element operations for all other computations. The cost of those changes are expected to be minor while the value for the possibility to optimize other algorithms is significant.

Also, our implementation indicates very high performance even for a low power ePUMA system.

6. CONCLUSION

We have presented a parallel implementation of the Euclidean Distance Transform for the ePUMA architecture. The algorithm we base our work on is a fairly well-known variant of the Euclidean Distance Transform, but had to be adapted for the ePUMA memory system in order to be efficient. The algorithm is the first inherently sequential image processing algorithm so far to be implemented on this platform.

The algorithm will produce one pixel every 2.3 cycles. For a 500 MHz ePUMA this indicates a processing time of 1.2 ms for a 512x512 image, indicating that more operations as well as larger data sets can be processed by the ePUMA even in low-power configurations.

Based on this work, we have discovered some desirable enhancements for the architecture, for more powerful conditional operations and better mixing of 16- and 32-bit operations.

ACKNOWLEDGEMENT

The ePUMA project is funded by SSF, the Swedish board for strategic research.

REFERENCES

- Borgefors G. 1986, "Distance transformations in digital images," *Computer Vision, Graphics, and Image Processing: An International Journal*, vol. 34, no. 3, pp. 344–371.
- Cao T.T, Tang K., Mohamed A., and Tan T.S., 2010, "Parallel Banding Algorithm to Compute Exact Distance Transform with the GPU", *The 2010 ACM Symposium on Interactive 3D Graphics and Games*, Maryland, USA, pp. 83-90.
- Danielsson P.-E., 1980, "Euclidean Distance Mapping," *Comp. Vision, Graphics, and Image Processing* 14, pp. 227-248.
- Eggers H., 1998, "Two Fast Euclidean Distance Transformations in Z Based on Sufficient Propagation," *Computer Vision and Image Understanding*, vol. 69, pp. 106-116.
- Liu D., 2008, *Embedded DSP Processor Design, Application Specific Instruction set Processors*, Elsevier, USA.
- Lucet, Y., 2009, "New sequential exact Euclidean distance transform algorithms based on convex analysis," *Image and Vision Computing*, vol. 27, no. 1–2, pp. 37–44.
- Paglieroni D.W., 1992, "A Unified Distance Transform Algorithm and Architecture," *Machine Vision Appl.* 5, pp. 47-55.
- Ragnemalm I., 1990, "Contour Processing Distance Transforms", in: Cantoni et.al. eds. *Progress in Image Analysis and Processing*, World Scientific, Singapore.
- Ragnemalm I., 1989, "The Euclidean Distance and its implementation SIMD architectures," *Proceedings, 6th Scandinavian Conference on Image Analysis*, Oulo.
- Ragnemalm I., Liu D., 2010, "Towards Using the ePUMA Architecture for Hand-held Video Games", *Proceedings, Computer Graphics, Visualization, Computer Vision and Image Processing 2010*, Freiburg.
- Rong G., Tan T.S., 2006, "Jump flooding in gpu with applications to Voronoi diagram and distance transform". *ACM Symposium on Interactive 3D Graphics and Games*, pp. 109–116.
- Rosenfeld A., Pfaltz J.L., 1966, "Sequential operations in digital picture proc.," *Journal of the ACM* 13 (4), pp. 471-494.
- Rosenfeld A., Pfaltz J.L., 1968, "Distance Functions on Digital Pictures," *Pattern Recognition*, vol. 1, pp. 33-61.
- Rutowitz D., 1990, "Efficient processing of 2-D images", in: Cantoni et.al. eds. *Progress in Image Analysis and Processing*, World Scientific, Singapore.
- Saito T., Toriwaki J.-I., 1994, "New Algorithms for Euclidean Distance Transformation of an n-dimensional Digitized Picture with Applications," *Pattern Recognition*, vol. 27, pp. 1551-1565, 1994.
- Schneider J., Kraus M. and Westermann R., 2009, "GPU-based Real-time Discrete Euclidean Distance Transforms with Precise Error Bounds", in *Proc. VISSAPP* (1), pp.435-442.
- Schouten T.E., van den Broek E.L., 2004, "Fast Exact Euclidean Distance (FEED) Transformation," in *Proceedings ICPR 2004*, vol. 3, Cambridge, United Kingdom, pp. 594–597, 2004.
- Shih F.-C., Mitchell O., 1992, "A mathematical morphology approach to Euclidean distance transformation", *IEEE Transaction on Image Processing* 1(2), 197-204.
- Tomasson, O., 2010, *Impl. of Elementary Functions for a Fixed Point SIMD DSP Coprocessor*, Linköping University.
- Yamada H., 1984, "Complete Euclidean Distance Transformation by Parallel Operation," *Proc., 7th ICPR*, pp. 69-71.

AUTOMATIC FUZZY SEGMENTATION OF A COLOR IMAGE – AN APPLICATION FOR ROBOCUP VISION SYSTEM

Juan M. Calderón, Andrea K. Pérez, Andres Gómez and J. A. López
Santo Tomás University

ABSTRACT

This paper presents two methods of segmentation based on fuzzy logic, designed to provide recognition of nine main colors that are developed for different categories of RoboCup. It explains a first method of manual segmentation based on fuzzy logic and a second method similar to the first but with an automatic color extraction. Both methods are applied to images extracted from the different RoboCup leagues and finally we make an analysis of the results.

KEYWORDS

Image segmentation, fuzzy systems, robot vision systems, RoboCup.

1. INTRODUCTION

The development of robotics has been promoted in recently years from different fields of knowledge, and has stimulated research in several parts of the world. It has resulting in a significant increase of the interested community in these developments.

In this context, the RoboCup world championship is very important, because it is a global initiative to promote research and teaching of robotics and artificial intelligence.

This initiative was launch in Japan in 1993, after a thorough analysis of financial viability, technological and social impact of using soccer to promote science and technology. Thus it is concluded that the project is viable and although at first it was a project launched in Japan, the initiative was accepted in different countries, becoming an international project, for which it has been suggested, that a mid-century a team of fully autonomous humanoid robots can compete with the world champion of soccer organized by FIFA.

Initially there were three categories, two wheeled robots designed for small and medium-size and one for simulation, but now they have opened more categories that allow broader context of competition, RoboCup is currently classified into three broad categories : Soccer, rescue, and Junior @ home.

These categories present challenges that develop skills of cooperation in the context of the design of artificial intelligence, robotics, electronics, and others. However, it is important to note that every challenge has different stages, so is necessary to develop optimal processes to achieve the goals.

In this way, the artificial vision systems needs to be supported by image segmentation algorithms, these algorithms are responsible for object recognition in most of the categories comprising the RoboCup World Championship.

We propose two methods to image segmentation based on fuzzy logic, applied to the specifically needs present in RoboCup. We developed a first method that initially defines membership sets for each condition and a second method, with automatic definition of membership functions (Wei 2007).

2. VISION SYSTEMS

There are two vision systems: local and global vision, each one is applied to different categories according to the characteristics in every case. The case of global vision, it is external to the robotic agents of competition;

it is the case of the soccer category of Small Size, as shown in Figure 1(a). The cameras over the game field, allow the capture of information that is collected in a system designed for this purpose. This system distributes the information to each team, but it is necessary to clarify that in this category is also possible to implement local vision system.

The local vision systems are those that are included within the robotic agents, such as the categories of humanoid soccer and Middle Size, as shown in Figure 1(a). Every robotic agent has cameras, so it is internally performed the process of artificial vision.



Figure 1. (a) Global vision system, Small Size league. (b) Local Vision of Humanoid league

3. FUZZY COLOR EXTRACTOR

3.1 Fuzzy Color Extractor

After a capture of images, it's necessary to perform the image processing. The color segmentation is the first step towards the recognition of elements within the image (Sahoo 1988), (Sezgin 2004).

Depending on lighting conditions, images can differ in color shade, so the segmentation process should encompass a range of representative colors for the color you want to extract. Due to the variety in the segmentation, we propose a color fuzzy extraction method that allows identify the desired color (Wei 2008).

The starting point of segmentation is based on a pattern of color components determined in red-green-blue (RGB) color range where a layer has a maximum of 255 levels.

The method establishes a group of fuzzy sets that classify the color range of each RGB layer of the image, dividing the color layer in lower (μ_L), medium (μ_M) and high levels (μ_H). For each group using a fuzzy set determined by the following equation.

$$\mu_X(z) = \frac{1}{1 + \left(\frac{z - c}{a}\right)^b}$$

$$X = \{(z, \mu_X(z)) | z \in Z\}$$

Where c is the center of the bell, a is width and b the slope.

The figure 2(a) shows three basic membership functions that are used to set rules for each color pattern.



Figure 2.(a) Input membership functions, (b) Output membership functions

The choice of rules is based on the color pattern. So for the red color pattern RGB (255, 0, 0) is defined the following rule.

IF red component is high **AND** green component is low **AND** blue component is low
THEN YES, color is red. **ELSE NO**, color isn't red.

Each combination of input functions corresponding to a rule of a set of 27 rules. According to the desired pattern the corresponding rule is chosen for the output bell function YES, the other rules are assigned to the output bell functions NO.

The output functions are defined as shown in Figure 2(b). When analyzing an image pixel, RGB values are taken and evaluated for the rules. Once this process is done, it is calculated the minimum between the resulting value and the output membership function appropriate for the rule.

In evaluating the appropriate rules and taking the out area for each rule, it's determined the maximum of these areas, resulting in a total output area.

Then we get a value from the resulting area by the method of defuzzification of bisector of area (BOA) described in the next equation.

$$\int_{\alpha}^{z_{BOA}} \mu_A(z) dz = \int_{z_{BOA}}^{\beta} \mu_A(z) dz$$

Where $\alpha = \min \{z \mid z \in Z\}$, $\beta = \max \{z \mid z \in Z\}$ and z_{BOA} is a vertical line that divides the region between α and β in two sections with the same area.

For segmentation, the output value obtained by the defuzzification, must exceed a defined threshold, provided by next equation, otherwise the pixel is discarded.

$$BOA > \varphi$$

Where φ is threshold.

Due to the variety of colors, φ can be modified for a better selection. Also it's possible to change the center and slope of input membership functions in order to cover more combinations of colors within the segmentation.

3.2 Automatic Fuzzy Color Extractor

The color extraction methods can present problems in the calibration process due to the number of variables that can be modified. The adaptation of membership functions for a correct segmentation can be time consuming and experience therefore proposes an automatic method of extracting color to adjust the parameters of the input functions, according to the expected target region (Laurent 1994),(Been 2002).

First we take a color sample and calculate the difference between values of the image layers (RGB), as indicated by following equation.

$$\begin{cases} dif_{R-G} = R - G \\ dif_{G-B} = G - B \\ dif_{R-B} = R - B \end{cases}$$

Where R is red layer component of the image, G is green layer component and B is blue layer component.

This method determines a membership function for each different component, for every rule there are three possible conditions for each difference and therefore three membership functions.

Based on the estimated differences, we calculated the mean ρ (dif) and standard deviation δ (dif) for each of the three differences. Subsequently each function is defined as a bell membership function with center ρ (dif) and width δ (dif). Defined by next equation:

$$\mu_X(z) = \frac{1}{1 + \left(\frac{z - \rho}{\delta}\right)^b}$$

To evaluate a pixel of the image, we calculate the difference between values of each color component. So for the pixel p (x, y) have differences d1, d2, d3, defined at the next equations:

$$\begin{cases} d_{R-G} = p_R(x, y) - p_G(x, y) \\ d_{G-B} = p_G(x, y) - p_B(x, y) \\ d_{R-B} = p_R(x, y) - p_B(x, y) \end{cases}$$

The rule applied to these input membership function is described below:

IF d_{R-G} is difR-G **AND** d_{G-B} is difG-B **AND** d_{R-B} is difR-B
THEN YES, (analyzed color). **ELSE NO**,(other color).

For each color in the segmentation process is generated a rule, then for n- colors are derived n-rules . For segmentation of a color is chosen the relevant rule in the output membership function YES (is color), the

others rules in the output membership function NO (other color)

For example, for specific case of green, there are three membership functions as shown in Figure 3.

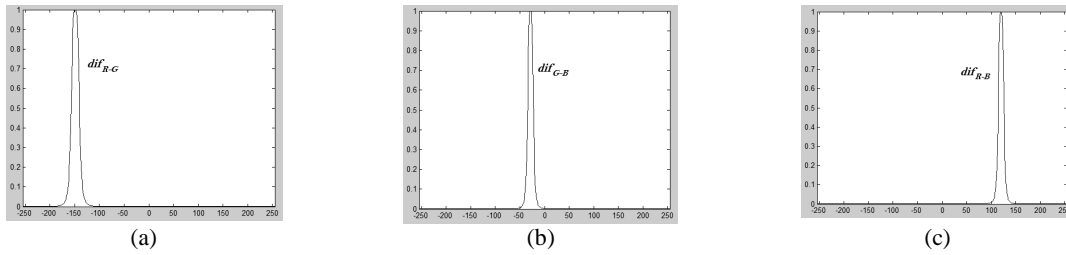


Figure 3. (a) Membership function dif_{R-G} . (b) Membership function dif_{G-B} . (c) Membership function dif_{R-B} .

When we evaluate a pixel, we calculate member value pixel into the three membership functions obtained, with this value is determined the minimum between it and the appropriate output function.

The pixel is valued in each of the rules implementing the process described above, then determines the maximum of the areas resulting from each rule and uses the BOA for defuzzification process.

If another sample pixel, membership functions change according to the color that is evaluated, so the above process must be done again , obtaining a new value of defuzzification.

4. APPLICATION METHOD

The proposed segmentation methods have focused on the challenges of the different categories of RoboCup, in this way there are nine possible colors, which are those that are working on these challenges. Those colors are: yellow, blue, white, cyan, magenta, orange, black, green and dark green.

Applying the method of fuzzy color extractor or segmentation with manual setting of membership functions, it was defined a fixed threshold for each segmentation color and is applied the explained method, however it is necessary to ensure that the selected region is the desired region and this region does not contain tones of other colors. As shown in the figura4 (a). A picture was taken of the Humanoid category, where was established a threshold of 0.5 for all colors, and the obtained image is showed in the figure 4(b).

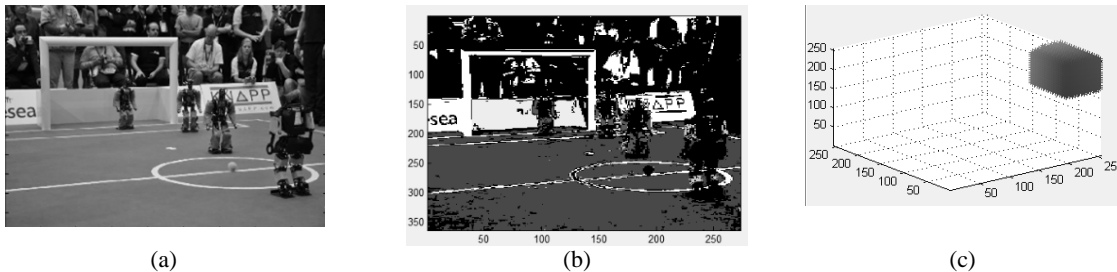


Figure 4. (a) Original image. (b) Segmented image. (c) color range appropriate to the magenta color

As an example, the magenta color is chosen to display the region that is generated inside the RGB cube. This color establishes a cubic region, as in Figure 4 (c).

For another case, a picture of the Small Size League was used, as shown in Figure 5 (a) and performs the segmentation process redefining the membership functions for each color that we want to target, the result after applying the same method is showed in Figure 5(b). The color range covered and the membership functions defined for this case, as shown in Figure 5 (c).

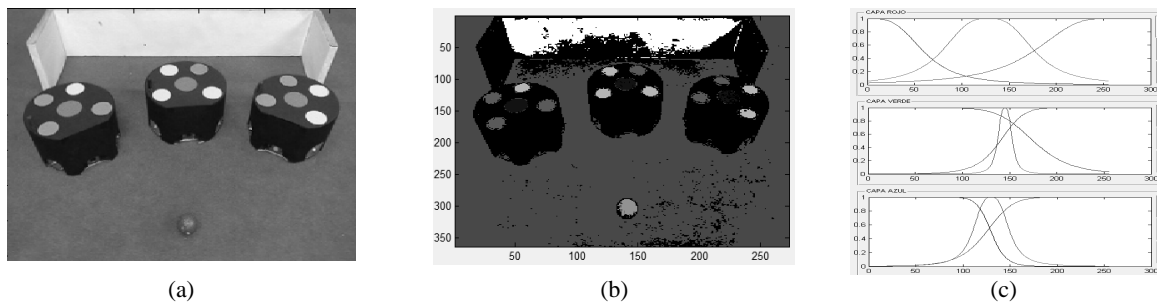


Figure 5. (a) Original image. (b) Segmented image. (c) membership functions low, medium and high layers for red, green and blue, respectively

On the other hand, the automatic segmentation method has an important feature, it takes less processing time compared with the first proposed method. In addition to generating functions, considering the difference between the RGB components showed less tendency to erroneous segmentations.

Additionally, as there are nine possible colors, the nine rules are evaluated and the output is obtained by competition, so that, choose the highest output value of all rules evaluated by segmentation process is more selective, unlike the first method in which the selection is done by setting a fixed threshold.

At the making a new test of the automatic method, we take the same image of the Small Size League, which was previously segmented with the first proposed method. Figure 6 (b) shows the result. The first method did not identify fully segmented regions, but the second method was able to fully cover the segmentation process. The region that is generated within the RGB color cube as a result of the identification of targeted color, unlike the other method is a non-cubic region, it is generating a cube cut irregular, containing a color distribution along the RGB space, as shown in Figure 6 (c)

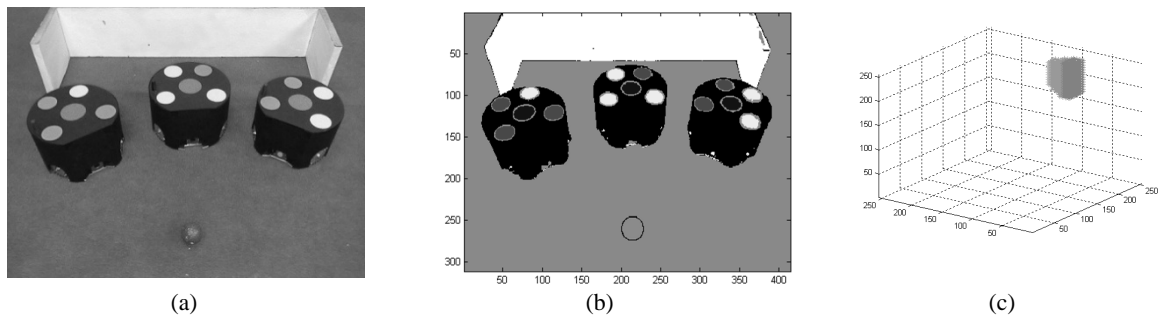


Figure 6. (a) Original image. (b) Segmented image. (c) Region corresponding to the magenta color

5. FUTURE WORK

Although this work is not entirely novel, represents a first step in a studio of fuzzy logic algorithms for image processing and color segmentation. At future work is expected to make a comparison between these results with other techniques based on color models such as HIS, YUV, among others.

6. CONCLUSION

According to the applications made by the proposed methods, you can see some difficulties faced in implementing them, however the results obtained are satisfactory, as they reach the objective, performing a properly colors segmentation for colors used in different RoboCup categories.

For the case of segmentation method to define membership functions manually, is possible to define the regions that we want to segment as cubes contained in the RGB color space, considering a full range of color.

But it does not appropriate in all cases, due to it can take inappropriate regions and overlapping it with other colors, thus preventing the proper segmentation.

By the last reason, the automatic segmentation method gives better results compared to those obtained with the first presented method, due to defining the membership functions according to the RGB components of the desired target region; it can generate selectively colors range, obtaining irregular regions contained in the RGB color cube and avoiding the overlapping of colors.

REFERENCES

- Wei Li, 2007. An Iterative Fuzzy Segmentation Algorithm for Recognizing an Odor Source in Near Shore Ocean Environments. *Proceedings of the IEEE International Symposium on Computational Intelligence in Robotics and Automation*, pp. 101-106, June 2007.
- Sahoo, P. et al, 1988. A Survey of Thresholding Techniques, *Computer Vision, Graphics and Image Processing*, vol. 41, pp. 233-260
- Sezgin M. and Sankur B., 2004. Survey over Image Thresholding Techniques and Quantitative Performance Evaluation. *Journal of Electronic Imaging*, Vol. 13, pp. 146-165.
- Wei Li, Yunyi Li, Jianwei Zhang. Fuzzy color extractor based algorithm for segmenting an odor source in near shore ocean conditions. *IEEE International Conference on Fuzzy Systems*, 2008. FUZZ-IEEE 2008.
- Laurent Wendling. Et al, 1994. Fuzzy Segmentation and Sample-Based Object Recognition in Remote Sensing Images. *International geoscience and remote sensing symposium (IGARSS)* 1994.
- Been-Chian Chien, Ming-Cheng Cheng, 2002. A color image segmentation approach based on fuzzy similarity measure. *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, FUZZ-IEEE'02.

DEFORMATION EXCHANGE BETWEEN LARGE MESHES

Benjamin Duplex¹, Gilles Gesquiere², Marc Daniel² and Fabien Perdu³

¹CEA, DEN, DER/SSTH/LDAL, Cadarache, F-13108 Saint-Paul-lez-Durance, France

²LSIS UMR CNRS 6168, ESIL Case 925, 163 Avenue de Luminy, F-13288 Marseille cedex 09, France

³CEA, DEN, DER/SSTH/LDAS, 17, rue des Martyrs, F-38054 Grenoble, France

ABSTRACT

To improve nuclear knowledge, physical simulations are made. Physical specific solvers, so called codes, are developed. Interactions between them must be done to increase the simulation accuracy of physical experiments. This work focuses on the transfer of geometrical deformations computed by a code to another code. The deformations result from irradiations and imposed constraints, and lead to large volume of data exchange. Our method mix a simplification process based on QEM and a RBF network allowing to exchange large geometrical modifications between physical codes.

KEYWORDS

Radial basis function, thin-plate spline, mesh simplification, quadric error metric, deformation exchange, code coupling.

1. INTRODUCTION

Some computer programs, called physical codes, are made to simulate physical phenomena. The French Atomic Energy and Alternative Energies Commission (CEA) develops physical codes specific to nuclear domains: neutronics, mechanics, thermo-hydraulics, fuel behaviour and so on. To improve simulation result accuracy, the CEA develops interactions between these codes through data exchanges. Works dealing with the exchange of physical values have already been achieved (Bonaccorsi *et al.* 2007, Perdu and Vandroux 2008, Vaiana 2009). They do not concern geometrical modification transfer. A deformation is calculated by a mechanical code, a program which focuses on mechanical behaviour, due to mechanical forces, pressure, temperature or irradiations. This paper addresses the problem of transferring deformations computed by a physical code to another one. Physical codes are based on geometrical representations which can be Constructive Solid Geometries (CSG) but are mainly meshes. We will restrict ourselves to the transfer of mesh deformations between two codes. The meshes are different but model the same (or a part of the same) geometry. Meshes could be a surface or a volume, with one or several kinds of shapes like tetrahedra or hexahedra. Mechanical codes used at the CEA generally calculate with meshes having less cells than other codes. The problem is finally to transfer a deformation computed on a coarse mesh to a finer one, without common shared nodes. Figure 1 summarizes it. Meshes noted (1) and (3) are geometry representations. Mesh (2) is the result of the mechanical code computation. Mesh (4) must be computed using the three others, even if the node number in meshes is very large, up to millions of nodes. We have developed a new function based on a simplification process and an interpolation method.

The remainder of the paper is structured as follows: section 2 reviews related work. Section 3 presents the method to transfer mesh deformations. A practical experiment will be studied in detail in section 4.

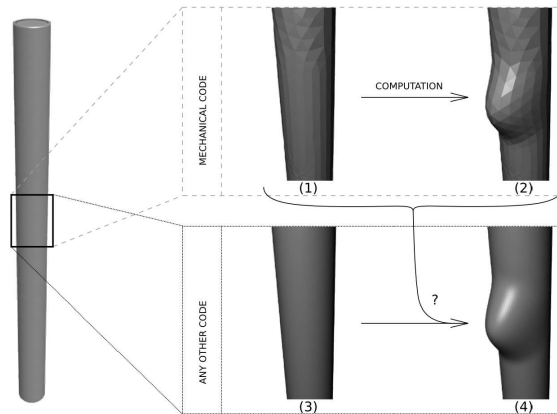


Figure 1. Summary of the principle of our method

2. PREVIOUS WORK

2.1 Scattered Data Interpolation

Methods are split into two groups (Wendland 2005): methods based on a mesh, like B-Spline method (Kim *et al.* 2008), or meshless methods, only based on point clouds. The last one offers the possibility to compute an interpolation function F at any point $P \in \mathbb{R}^D$. We note $\{L_j\}_{j=1,\dots,n}$ the node set of the initial mesh, $\{L'_j\}_{j=1,\dots,n}$ the node set of the resulting mesh computed by the mechanical code. n is the number of nodes in meshes. Interpolation condition of function F is: $F(L_j) = L'_j, j = 1, \dots, n$. Such functions contain a representation of the deformation and could be apply to another mesh extracted from the same geometry. Methods specific to noisy meshes have been proposed. They imply many control parameters and interpolation errors (Bozzini *et al.* 2010). Others methods are complex, as in (Johnson *et al.* 2009). Authors use a combination of B-splines and wavelet spaces. Their results are quite the same as the well-known approach using Radial Basis Functions (RBFs) (Wendland 2005) which are easy to implement while providing good interpolation results. A RBF network is a multivariate scattered data interpolation method composed with a linear combination of RBFs. At any point $P \in \mathbb{R}^D$, the network is defined by:

$$F(P) = \sum_{k=1}^n \lambda_k \cdot \phi(\|P - L_k\|) + \mathcal{L}(P)$$

where ϕ is a RBF centred on point $L_j \in \mathbb{R}^D$, $\lambda_j \in \mathbb{R}^D$ a network coefficient, n the number of points in the network and \mathcal{L} a low degree polynomial function in \mathbb{R}^D (the degree is less than those of ϕ , see (Iske 2003) for more details). RBF centres are known value positions that function F must interpolate. In the following, the Euclidean norm is used. Physical code meshes are in \mathbb{R}^3 . Several formulations of RBFs were proposed: Gaussians ($\phi(r) = \exp(-(r - \alpha)^2/\beta^2)$), Thin-plate splines (TPS) ($\phi(r) = r^{2k} \cdot \log(r)$), polyharmonic splines ($\phi(r) = r^{2k+1}$), multiquadrics ($\phi(r) = (r^2 + \alpha^2)^\beta$), Compactly supported radial basis functions (CS-RBF).

Even if parameter β of the multiquadric functions is often set to 1/2 or -1/2, the choice of parameter α is still an open problem. Gaussian functions need to have a uniform repartition of nodes. Our surfaces do not have this restriction. CS-RBFs are functions with a compact support. They imply to fix its influence area. Some works were done (Ohtake *et al.* 2005) but it is still an open problem. In (Bozzini and Rossini 2002), authors show that polyharmonic spline $\phi(r) = r^3$ and TPS function $\phi(r) = r^2 \cdot \log(r)$ are the best choice. The first one has a better result in terms of local and global accuracy, nearly followed by the second one. Otherwise, (Bozzini and Rossini 2002) proves that $\phi(r) = r^3$ has a higher matrix condition number, justifying the choice of TPS function. For TPS, parameter k is used to specify the continuity of ϕ function which is C^{2k-1} continuous (Boztosun *et al.* 2002). CEA mechanical codes solve second order equations so that C^2

TPS function is chosen. In (Iske 2003), author proves that the $\phi(r) = r^{2k} \cdot \log(r)$ order of TPS, with $k \in \mathbb{N}$, is $m = k + 1$. C^2 TPS implies $k = 2$ so $m = 3$. The order of the polynomial function \mathcal{L} must be less than this value. Usually, a linear polynomial is used to represent global deformations of first order (Beckert and Wendland 2001). We note $P^{(i)}$ component i of point P . The polynomial function is then:

$\mathcal{L}(P) = \lambda_{n+1} \cdot P^{(1)} + \lambda_{n+2} \cdot P^{(2)} + \lambda_{n+3} \cdot P^{(3)} + \lambda_{n+4}$ To take into account conditions from additional polynomial terms (Iske 2003), we impose:

$$\sum_{k=1}^n \lambda_k \cdot L_k^{(i)} = \sum_{k=1}^n \lambda_k = 0, \quad \forall i = 1, 2, 3$$

RBF network coefficients are computed by solving the linear system induced by the interpolation condition. RBF calculation performance relies on the number of points $\{L_j\}'$. Despite good results, interpolating scattering data using RBF's is computationally expensive and requires lots of memory. To overcome this problem, some approaches approximate coefficients (Botsch and Kobbelt 2005) but we need to interpolate mesh displacement. Others prefer to reduce the number of points $\{L_j\}'$ (Dyn *et al.* 2008). We propose a new method based on mesh simplification process. Before detailing our method, simplification methods are quickly presented. Due to mesh size and heterogeneity, we must consider methods efficient on large meshes and mesh independent, in terms of dimension (surface or volume) and cell type (triangle or quadrangle).

2.2 Simplification Function for Surfaces and Volumes

(Joy 2009) shows that no method can handle all kinds of meshes. Edge contractions are powerful surface simplifying methods (Kraus and Ertl 2002). They consist in sorting edges (or more generally any node pair) with a cost criterion. The point pair fitting the criterion is contracted into a new node and the cost of the others is updated. This process is made until a stop condition is reached: for instance, the number of nodes in the final mesh or an exceeded maximal cost. The cost function could be based on an energy in (Hoppe 1996), or a QEM (Quadric Error Metric) in (Garland and Heckbert 1997). QEM allows to simplify preferably flat areas and zones with important curvatures remain precise. In (Bac *et al.* 2007), authors simplify very large meshes, more than 10 millions of points, with the help of a preliminary clustering method. Although it improves processing time, this step adds errors. The process merges nodes inside cluster cells without taking into account the mesh information, as the curvature.

Our RBF network will be based on the simplified mesh nodes, so that our method accuracy is linked to data kept during the simplification step. QEM permits to simplify preferably flat areas and to keep curvature information. It can only work on triangular meshes. All faces are previously split. We note $Q(P)$ the QEM of the point P . First, all QEM of nodes in the mesh are initially computed. Then, for all point pairs, the contraction cost of $(P_1, P_2) \rightarrow \bar{P}$ is calculated by $\Delta(\bar{P}) = \bar{P}^T \cdot Q(\bar{P}) \cdot \bar{P}$. The position of \bar{P} could be P_1 , P_2 , the middle point or totally computed to minimize an energy.

3. A NEW METHOD

We introduce a new notation: $F_\alpha(P)$ denotes the RBF network after a simplification which has kept α nodes. We note $\{L_k\}_{k=1, \dots, \alpha}$ (resp. $\{L'_k\}_{k=1, \dots, \alpha}$) the node set in the initial (resp. deformed) simplified mesh.

3.1 Modifications of Simplification Method

The aim of these modifications is to be able to apply the simplification method and the RBF network for any kind of large meshes. To begin, we impose \bar{P} to be equal to P_1 or P_2 , depending on costs, to keep the link between meshes before and after simplification step. The method, selected to reduce mesh complexity, must simultaneously work on both meshes. It must contract the same edges in both meshes because the initial mesh might contain particularities which are missing in the deformed mesh and vice versa. We propose to blend quadrics of both meshes in the global cost function. This modification does not add calculation and allows to keep all details. The final quadric is the maximal value between the quadric in the initial and the deformed mesh to ensure.

Garland’s criterion performs the simplification of edges that involves the smallest error, *i.e.* low curvature areas. Because the method is very powerful, node repartition inside simplified mesh could be non-uniform. RBF method does not need a uniform node distribution but such a simplification step could generate bad results due to the lack of information to interpolate. To overcome this problem a second criterion for simplification is added to preserve the homogeneity. The idea is to take into account the face areas to keep nodes. We define a new function $H(P)$ which is the sum of the set of faces containing point P in the initial and the deformed mesh.

With the help of both criteria, a new cost function is computed by: $C(P) = \beta \Delta(P) + (1 - \beta)H(P)$, $\beta \in [0, 1] \in \mathbb{R}$. The β parameter allows to modify the importance of both criteria in the final cost. Meshes at millimetre order have to set β lower than meshes at metre order. Function H calculates area on faces, so that areas depend on the mesh order. Figure 2 shows possible simplifications on a half-circle depending on the value of β . If $\beta = 1$, every nodes inside the half-circle will be deleted in order to maintain the half-circle aspect. If $\beta = 0$, the aspect is not kept but nodes will be conserved inside the mesh. A middle value, like for $\beta = 0.5$, will mix these effects. During tests, β was fixed to 0.5.

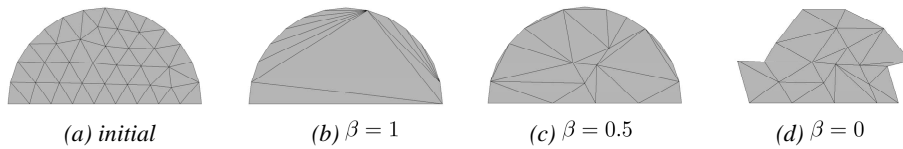


Figure 2. Simplification step using Garland’s method

3.2 Application of F_α

After the F_α computation, function is used to modify an other mesh. The RBF network is a space deformation function (Wendland 2005). For all nodes $\{P_j\}_{j=1,\dots,m}$ of this mesh, new positions $\{P'_j\}_{j=1,\dots,m}$ be computed:

$$F_\alpha(P_j) = \sum_{k=1}^{\alpha} [\lambda_k \cdot \phi(\|P_j - L_k\|_2)] + \mathcal{L}(P_j) = P'_j, \quad j = 1, \dots, m$$

The same network can be applied to several different meshes. The simplification method involves errors. On the other hand, all nodes cannot be kept. The computing time decreases when the node number decreases. For this reason, the impact of the simplification step is discussed below.

4. RESULTS

The RBF network interpolates the node displacements. The error introduced by the simplification process is viewable by applying function F_α to the initial mesh and by analysing the difference between the mesh computed by the mechanical code and those computed our function. This analysis is evidently only valuable to illustrate the method quality. Two criteria are used. The first one is local and exhibits the maximum error. The second, the well-known Root Mean Square (RMS), is global and informs about the average interpolation quality. Let L'_j be the j^{th} node computed by the mechanical code and L''_j the displaced node computed by F_α .

$$E_{max} = \max_{j=1}^{\alpha} (\|L''_j - L'_j\|_2) \quad E_{RMS} = \sqrt{\frac{1}{\alpha} \sum_{j=1}^{\alpha} (\|L''_j - L'_j\|_2)^2}$$

The experiment considered to test our method is a half steam generator mock-up called "CLOTAIRE" (Campan and Bouchter 1988). Its simplified geometry is a half-cylinder open and fixed at the bottom. A force can be applied to this object at the top. Subject to mechanical constraints, the object is deformed. The modification is calculated by a mechanical code and our method is used to deform the mesh of another code based on the same geometry. We apply our method to modify the initial mesh (figure 1) and to calculate errors due to the simplification step. We restrict ourselves to meshes with only 30,500 nodes to be able to compute the errors without simplification. Available memory space on our computer is 8GB and the RBF network requires about 7.2GB to store the interpolation matrix of 30,500 nodes.

4.1 Impact of the Simplification Step on Errors

The evolution of errors due to the simplification step is shown in figure 3. Peak between simplification at 100 nodes and 1,000 nodes on both graphs are the illustration of the lack of information to interpolate. One can notice that the introduction of $H(P)$ helps to reduce this effect. On this example, except for peaks, errors with and without $H(P)$ are quite the same. Obviously, they depend on parameter β which is mesh dependent.

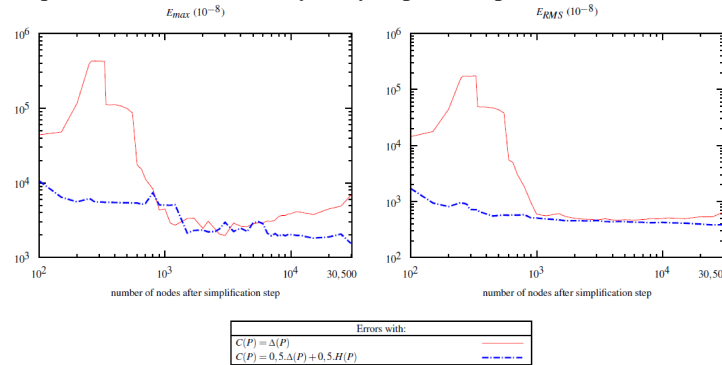


Figure 3. Principle of the error computation due to the simplification step

4.2 Processing Time and Memory Space

The simplification step also modifies the processing time and memory space required. The linear system resolutions are computed by the direct Gaussian elimination. System matrices are very dense with TPS RBF. Every computation has been made with double precision on Linux Debian Etch, with a 64-bit processor using only one core at 2.6GHz and 8GB of main memory. Influence of the simplification step on performances is shown in table 1. It contains the processing time of the whole method (computing time and application time), the peak of required memory and errors E_{RMS} and E_{max} with parameter $\beta = 0.5$ and $\beta = 1$. Our method is in $O(n^3)$ for solving the system and in $O(n^2)$ for the new position calculation. Acceptable errors depend on the study and input data. A compromise has to be found between the accuracy and the total computing time and memory space required. For physicists, an acceptable method has to perform this transfer with a lower order in time than their code computing time, as for instance few hours if the code will compute during two days.

4. CONCLUSION

Physical codes need up-to-date meshes. For this purpose we propose a novel two-steps method. First, we simplify the meshes using linked QEM cost metric. Secondly, an interpolation method using a RBF network is employed. Some modifications were made to mix them. A use case is presented and our method is tested to analyse the simplification method impact. We have demonstrated that the proposed method is effective and allows physicists to transfer mesh deformations between physical codes. Parameter β is mesh dependent. A further study is still needed to find an optimal value. We are now working to consider CS-RBF and to solve the system by an iterative method to decrease the processing time and memory space requirements.

ACKNOWLEDGEMENT

This research is supported by the French Atomic Energy and Alternative Energies Commission (CEA). The authors would like to thank the reviewers for their help and their remarks.

Table 1. Influence of the simplification step on performances

Number of nodes	Application time	Memory peak				
500	5s	52 Mo	566	43.387	5.420	98.638
1.000	9s	52 Mo	508	582	5.048	4.473
3.000	27s	109 Mo	459	467	2.962	1.975
6.000	55s	318 Mo	427	467	2.829	2.923
10,000	1min 38s	808 Mo	423	498	2.032	3.868
20,000	3min 07s	3174 Mo	395	535	1.888	4.482
30,500	5min 09s	7209 Mo	0	0	0	0

REFERENCES

- Bac, A. *et al.*, 2007. *A hybrid simplification algorithm for triangular meshes*, E_{RM} SiCon'2007, pp. E_{max} 4.
- Beckert, A. and Wendland, H., 2001. *Multivariate interpolation $f_0\beta = 0.5ruct\beta = 1era\beta = 0.5bler\beta = 1g$ radial basis functions*, *Aerospace Science and Technology* 5, pp. 125 – 134.
- Bonaccorsi, T. *et al.*, 2007. *Data exchange interface and model for coupling softwares in nuclear reactor simulations*, International conference in Product Lifecycle Management, Milano, Italy, pp. 207–216.
- Botsch, M. and Kobbelt, L., 2005. *Real-time shape editing using RBFs*, *Comp. Graphics Forum*, pp. 611–621.
- Boztosun, I. *et al.*, 2002. *Thin-plate spline radial basis function scheme for advection-diffusion problems*, *Electronic Journal of Boundary Elements BETEQ* 2001 (2), pp. 267–282.
- Bozzini, M. and Rossini, M., 2002. *Multivariate Approximation and Interpolation with Applications*, Vol. 20 of *Monografias de Zaragoza*, chapter Testing Methods for 3D Scattered Data Interpolation.
- Bozzini, M. *et al.*, 2010. *Polyharmonic splines: An approximation method for noisy scattered data of extralarge size*, *Applied Mathematics and Computation* 216(1), pp. 317–331.
- Campan, J. and Bouchter, J., 1988. *Steam generator experiment for advanced computer code qualification: Clotaire international program*, NURETH3, Seoul, South Korea.
- Chen, Y. *et al.*, 2006. *Meshless methods in solid mechanics*, Springer Verlag.
- Dyn, N. *et al.*, 2008. *Meshfree thinning of 3dpoint clouds*, *Foundations of Comp. Mathematics* 8(4), pp. 409–425.
- Garland, M. and Heckbert, P. S., 1997. *Surface simplification using QEM*, *SIGGRAPH*, pp. 209–216.
- Hoppe, H., 1996. *Progressive meshes*, in H. Rushmeier (ed.), *SIGGRAPH96*, pp. 99–108.
- Iske, A., 2003. *Radial basis functions: Basics, advanced topics, and meshfree methods for transport equations.*, *Rendiconti del Seminario Matematico, Universita e Politecnico Torino*, pp. 247–286.
- Johnson, M. J. *et al.*, 2009. *Scattered data reconstruction by regularization in b-spline and associated wavelet spaces*, *Journal of Approximation Theory* 159(2), pp. 197–223.
- Joy, K., 2009. *Massive data visualization: A survey*, *Mathematical Foundations of Scientific Visualization, Computer Graphics, and Massive Data Exploration*, pp. 285–302.
- Kim, S.-H. *et al.*, 2008. *A method for modifying a surface model with non-uniformly scattered displacement constraints for shoe sole design*, *Adv. Eng. Software* 39(9), pp. 713–724.
- Kraus, M. and Ertl, T., 2002. *Simplification of nonconvex tetrahedral meshes*, *Hierarchical and Geometrical Methods in Scientific Visualization*, pp. 185–196.
- Ohtake, Y. *et al.*, 2005. *3d scattered data interpolation and approximation with multilevel compactly supported RBFs*, *Graphical Models* 67(3): 150 – 165. Special Issue on SMI 2003.
- Perdu, F. and Vandroux, S., 2008. *System/CFD Coupling for Reactor Transient Analysis. An Application to the Gas Fast Reactor with CATHARE and TRIO U*, ICAPP'08.
- Vaiana, F., 2009. *Couplage Neutronique - Thermohydraulique : Application au Réacteur à Neutrons Rapides refroidi à l'Hélium*, PhD thesis, Institut polytechnique de Grenoble.
- Wendland, H., 2005. *Scattered Data Approximation*, *Cambridge Monographs on Applied and Computational Mathematics* (No. 17), Cambridge University Press.

DEPTH VALUE DEDUCTION USING OPTICAL FLOW FOR REVERSE ENGINEERING

Rahmita Wirza Rahmat and Suhail Azmi

Faculty of Computer Science and Information Technology - Universiti Putra Malaysia, Serdang, Selangor, 43400, Malaysia

ABSTRACT

Optical flow is actually a tracking algorithm used for detecting moving features in images (at least two). With optical flow, we can get the motion magnitude and vector of the tracked features. In this paper, we proposed an algorithm to acquire 3D depth information from images using optical flow magnitude, while we filtered out the noise by using its vector. The idea; when rotating an object in front of sensors, nearer parts of the object will move more than further parts, which mean, the nearer part has a bigger magnitude than the latter.

In this algorithm we are trying to achieve full automation in detecting 3D depth of the tracked features by exploiting the optical flow detected. To achieve this, we prepared an optimum environment for the implementation. Camera is placed at a specific location perpendicular to a black turntable with black background. With two images, one taken at zero degrees and one at 2.5 degree rotation, we apply a Pyramidal Implementation of Lucas Kanade optical flow to track detected features. Experiments are done to validate the 3D points constructed; three objects were tested quantitatively. The implementation of this framework succeeded in extracting features, estimating depth information and visualizing 3D result. Comparing with the real objects, we can conclude that the results are similar but not as accurate as conventional reverse engineering techniques. However, it will be a step forward to have a framework and a technique that greatly reduces cost and time, aside from being portable. Future research should be carried out in perfecting this framework because of its advantages and prospects.

KEYWORDS

Reverse engineering, optical flow

1. INTRODUCTION

Reverse engineering a geometrical model basically consists of extracting 3D shape or contours from the objects of interest and reconstructs the similar 3D model before send the model to prototype department (Wirza, 2002). The processes involves in reverse engineering are varied upon the problems caused by the nature of the objects. In other words, ways to reverse engineer certain objects are different from each other. There are contact-based and non-contact based to reverse engineer. In this paper, we propose a non-contact based techniques of reverse engineering by using images of the object. The term reverses engineering is carried in many field of studies: machinery, software, biology, semiconductor, electrical, mechanical, architectural, historical artifacts and many more. There are lots of definitions about reverse engineering depending on the field that uses it. But generally, geometrical reverse engineering is; to study about the functionality and design of the object and try to reproduce the same or enhanced version of the object when the original documentations are unavailable. Reverse engineering is motivated by few reasons; mainly because the information needed to produce an existed product or object is unavailable, documentation may be lost to fire or natural disaster or some other destruction, no available resources such as blueprints, enhancing existing products where the information of the product is confidential by other parties (Wirza, 2006). However, in this paper we focus on non-contact reverse engineer where this category does not involved touching or having a physical contact towards the object of interest. Aside from not being as accurate as contact based techniques, non-contact based can easily measure objects that cannot be measured by Coordinate Measuring Machine (CMM), so we can capture the shape and structure without damaging or change the shape of the object (Creehan, 2006).

Meanwhile structure from motion is one way of obtaining 3D information from 2D images. Humans perceive a lot of information about 3D structure in their environment by moving through it. When the observer moves and the object around him move, information is obtained from images sensed over time (Bradski, 2008). Basically our approach starts with finding correspondence between images, features such as corner points (edges with gradients in multiple directions) to be tracked from one image to the next. The feature trajectories over time are then used to reconstruct the object 3D position and camera motion (Jean-Yves Bouguet, 1998). Technically, the goal of optical flow is to track certain features of images (edges, corners, gradients- depending on the algorithm) and track the direction of their motion to the next image. We need at least two images in order to produce the optical flow. Two main processes in optical flow are; first, to search for traceable features, and second track these features in the next image. There are many type of features we can detect such as gradients, edges and corners, which identified as easily distinguished, not mistakenly detected as other features and exist in the next image. In the meanwhile, Image Based Modelling (IBM) is basically reconstructing 3D models out of 2D images (Jean-Yves Bouguet, 1999b). This method was widely used in recovering 3D structure of architectural objects where most of the techniques developed exploit the nature of the structures such as perpendicularity, orthogonality and parallelism or combined with image-based modeling techniques (hybrid technique)(Remondino, 2006).

2. PROPOSED SOLUTION

The idea of this propose solution is simple; if something is rotated, the center of rotation will have the shortest magnitude comparing to the magnitude of location farther away from the center of rotation. As we are going to use optical flow to detect the movement of our targeted object, we need to understand the nature of optical flow. How it detects the features, and how it track the features. However, the Lucas-Kanade optical flow is based on several assumptions, which are brightness constancy, temporal persistence and spatial coherence (Jean-Yves Bouguet, 1999a). Optical flow is highly dependent on the lightings. When dealing with images or capturing images, it's a general knowledge that the things we see around us is the reflected lights cast upon the objects. Most of computer vision algorithm cannot work well without proper lightings (Bradski, 2008). When we have an optimized lighting for capturing the images, what we need to make sure is that these lightings are as constant as possible (from experiment to experiment). The idea that we come up with is to conduct the experiment in the very same room and very same location with the same amount of lights shone to the object.

Before running the experiment, there are some preparations need to be done in order to get proper images. The first thing to do is create the ideal environment for taking pictures of the object of interest, as in Figure 1. The rotating plane and surrounding of the object need to be black in color; more accurately 'flat black' as we need to absorb as much light as possible in order to create a high contrast between the object and the background. From few experiments, we can see that the feature tracker will perform better in determining the feature of the object if the background is black. This happens because the light received by the camera from the object is more than the surrounding, making the object 'stand-out'. After captured the first picture using a webcam, we rotated the platform 2.5° so that the object of interest will be slightly rotated.

By implementation of the optical flow method introduced by Jean-Yves Bouguet (1999), we are able to collect the good features to track in the first image, and the corresponding good features to track in the second image, plus the optical flow of the image which connect features in first and second images. This information will be written into an external file. From the design experiment, we assumed that the furthest point will be $z=390\text{mm}$, and the optical vector magnitude linearly dependent with the distance of the point to the camera, similar with world optical vector magnitude (MW). With this assumption we proposed this algorithm.

- i. Put the object in the proposed environment and capture two images with and without rotation
- ii. Process the object's 1st image and measure in pixel values for the width of the object's image
- iii. Calculate the scaling vector, $S_p = \frac{\text{object real width}}{\text{object pixel width}}$;
- iv. Apply Lucas-Kanade algorithm and extract features good to track
- v. Locate each feature point pairs of the object's image before and after rotation and save their coordinates $(xv_{1,i}, yv_{1,i})$ and $(xv_{2,i}, yv_{2,i})$ in an output data file

- vi. Filter any data points that are further from the object's image.
- vii. For each pair of points, calculate:
 - a. The magnitude optical vector ($MV_i = \sqrt{(xv_{2,i} - xv_{1,i})^2 + (yv_{2,i} - yv_{1,i})^2}$)
 - b. The world coordinate for each pair of points ($xw_{1,i} = xv_{1,i} \times S_p, yw_{1,i} = yv_{1,i} \times S_p, xw_{2,i} = xv_{2,i} \times S_p, yw_{2,i} = yv_{2,i} \times S_p$)
 - c. The World Magnitude Optical Vector ($MW_i = MV_i \times S_p$)
- viii. Calculate $z_i = \frac{\text{Width of the object}}{\text{Max MW} - \text{Min MW}} \times (\text{Max MW} - MW_i) + (390 - \frac{\text{width of the object}}{2})$, finally
- ix. Display the coordinate values of world coordinate, $\{(xw_{1,i}, yw_{1,i}, z_i) | i \in \{1, \dots, N\}\}$.

3. VALIDATION

To validate the proposed algorithm we implemented it onto a tin can with colorful surface, refer to figure 2. With $i=1, \dots, N$ (total number of optical points), width of the object=radius of the tin, and by assuming that the camera able to capture front semi-cylinder image, we proceed with the validation task. As the diameter of the tin can at 66 mm, and the diameter of the tin can image at 205 pixels, we compute that, $S_p=66/205=0.322$. To simplify our calculation we simplify the z formula, by eliminating the (390-diameter of the cylinder) in the proposed formula. This can be done, as it only gives the relative locations and the importance of this project is to measure the dimension of the object. After checking the reliability of the z formula, we precede the experiment with the proposed algorithm. Through the observation of the output display, it seems that there is no data point that can be considered as noise, yet the accuracy still far from satisfactory. So we presume that we need to observe the MW distribution. From the observation of MW distribution, we decided to add the filtering rules by filtered the points which are far from the standard deviation values. Once satisfied the experiment proceed with the validation procedure. To validate this experiment we only consider data points that are within $100 \geq y \geq 40$, and their errors were calculated using $\frac{\sqrt{x^2 + (z-33)^2} - 33}{33}$. We found out that the maximum error is 0.313139 located at $y=56.47006$, the minimum error is 0.001291, standard deviation for error is 0.037538 and the mean error is 0.05596. Even though we not really satisfied with the error analysis, never the less observing figure 3, we can see the shape of the tin was preserved.

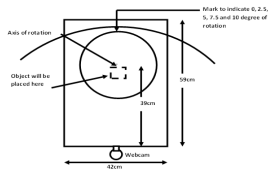


Figure 1. Environment set-up



Figure 2. The first image with the optical flow vector.

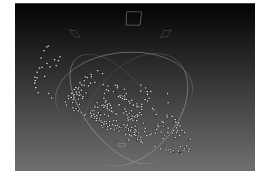


Figure 3. Cloud of points displayed using Meshlab.

4. CONCLUSIONS

We conclude that this technique is a very sensitive technique and heavily weighted on Optical Flow Technique. To achieve higher accuracy, we need to choose a random color object and specular lighting effect. With maximum error 0.313139, minimum error 0.001291, and mean error 0.05596 for the Tin Can, we consider it acceptable even though we need to enhance the Optical Flow section. We have done the validation using Toy ball and Rubik's cube, and the results are similar (refer to figure 4,5,6 and 7). To get better smooth surface points without scarifying the accuracy, we proceed implementing a simple basic algorithm onto the 3D data points. This simple basic algorithm re-arranged the points; reconstructed the surface by implementing Ball and smooth the surface with Laplacian surface, which are available in Meshlab as in figure 8. This paper shows that optical flow technique can be used for geometrical reverse engineering,

and prove of concept, that there is possibility that we can deploy this approach to reverse engineered complex shape of biological object, such as anatomical implant.



Figure 4. Ball with black pen sketch for features tracking

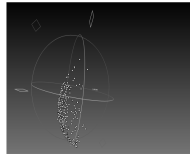


Figure 5. 3D cloud of points using MeshLab-side view for ball

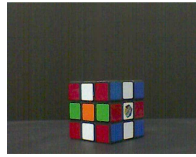


Figure 6. Rubik's cube as a sample.

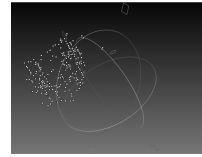


Figure 7. 3D cloud of points using MeshLab-side view for Rubik's cube

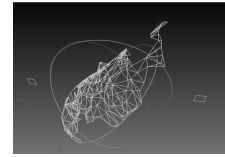


Figure 8. Laplacian surface in Meshlab.

REFERENCES

- Bradski, G. et al, 2008. *Learning OpenCV: Computer Vision with the OpenCV Library*. O'Reilly Media, California, USA.
- Creehan, K. et al, 2006. Reverse Engineering: A Review & Evaluation of Non-Contact Based Systems in Rapid Prototyping, *Manufacturing Systems Engineering Series*, Vol. 6, Part 1, pp 87-106
- Elham Bagherian. et al, 2008. Facial Features Extraction for Face Recognition: A Review. In *Proceedings of International Symposium on Information Technology 2008*, Vol. 2, pp 1-9.
- Jean-Yves Bouguet, 1998. 3D photography on your desk. In *Proceedings of the IEEE International Conference on Computer Vision*, pp 43-50
- Jean-Yves Bouguet, 1999a. Pyramidal Implementation of the Lucas Kanade Feature Tracker Description of the Algorithm. Unpublished document, Intel Corporation, Microprocessir Research Labs.
- Jean-Yves Bouguet, 1999b. 3D Photography using Shadows in dual-space Geometry. In *International Journal of Computer Vision*, Vol 35, Issue 2, pp 129-149
- Remondino, F. et al, 2006. Image-Based 3D Modeling: A Review. In *The Photogrammetric Record Journal*, Vol. 21, No. 115, pp 269-291
- Wirza R. et al, 2002. Inspection Strategies for Complex Curved Surfaces Using CMM. In *Lecturer Notes in Computer Science*, Vol. 2331, pp 184-194.
- Wirza R. et al, 2006. Complex Shape Measurement Using 3D Scanner. In *Jurnal Teknologi UTM, Series D, Bil. 45*, pp 97-112.

BEZIER CLIPPING METHOD IMPROVEMENTS FOR DIRECT RENDERING OF BEZIER PATCHES

Sergey Fedorov, Mikhail Letavin and Maxim Shevtsov
Intel

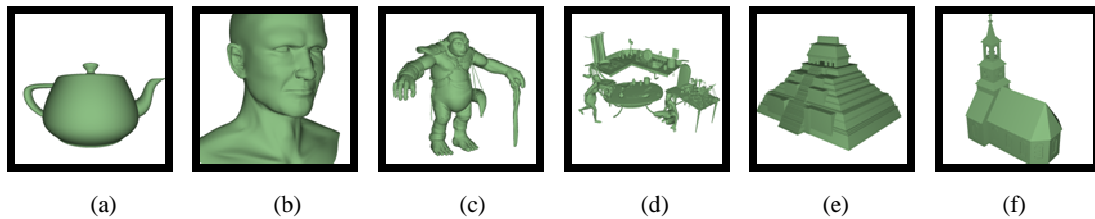


Figure 1. Teapot (a): 32 patches, camera FOV $6\pi/5$, 28fps. Head (b): 6186 patches, camera FOV $2\pi/5$, 14fps. Tahoro (c): 172572 patches, camera FOV $3\pi/10$, 11fps. Kitchen (d): 33497 patches, camera FOV $7\pi/10$, 11fps. Pyramid (e): 614 patches, camera FOV $3\pi/5$, 9fps. Kirche (f): 4481 patches, camera FOV $\pi/2$, 15fps. Image resolution: 1024x1024 pixels. Test workstation: Intel Core i7 975 3.33 GHz (4 cores, 8 threads), 6GB RAM, Win7.

ABSTRACT

We present an improved method for direct (without tessellation) rendering of rational Bezier patches that allows efficient parallel implementation. More numerically robust method for adaptive computation of termination criteria in Bezier clipping algorithm is presented. In turn, performance improvement for the method is achieved via clever preclipping patches on scanlines. Finally our implementation achieves real-time performance for the complex scenes with up to hundreds of thousands of patches on the commodity CPU.

KEYWORDS

Computer graphics, parametric surfaces, rational Bezier surface, direct Bezier rendering, Bezier clipping method.

1. INTRODUCTION

Modern GPUs have specialized hardware that allows patch tessellation and the resulting triangles are processed by the traditional triangle pipeline. However, this approach suffers from creating many tiny triangles that might degrade performance. A much more flexible and compact approach is to use higher order surface representations like Bezier surfaces. This way higher order surfaces have already entered the field of custom software pipelines. However their rendering is extremely computationally demanding.

One solution is to generate polygonal approximation instead, either offline or directly on the fly. We do not consider offline methods further. Generating polygonal approximation on the fly can provide sufficient accuracy by using view-dependant methods but it requires an additional preprocessing stage. Finally, both types suffer from being extremely memory consuming especially when complex objects are rendered in high resolution.

From a quality perspective the most reliable is the so called direct approach: subdivide patch until its boundaries won't be within pixel. The obvious downside is the performance since this method is really slow due to expensive subdivision operations performed for each pixel. This paper addresses this exact issue, making it possible to achieve real-time performance on the commodity CPU. Further we discuss two primary ways we developed to optimize direct rendering:

- The first way is to develop new termination criteria that would stop subdivision iteration when subpatch is proved to be within a pixel tolerance, thus skipping most of expensive subdivision iterations;
- The second is grouping of the first subdivision iterations for adjacent pixels to reduce overall number of subdivision iterations.

Modern approaches to direct Bezier rendering fall into two categories. Methods based on Newton’s iteration algorithm of finding a line-patch intersection belong to the first category. Newton’s iteration method has significant drawback of being dependant on the initial guess what makes it impractical. The basic approach of detecting parameter ranges where this method is guaranteed to converge was found by Toth [1985] but it was proved to be very time consuming. Another method called Bezier clipping was proposed by Nishita et al [1990]. This method doesn’t suffer from initial guess problem but has another bottlenecks investigated by Campagna, Slusallek[1996] and Efremov[2005]. Wang et al [2000] presented modifications to improve algorithm efficiency in case of multiple roots.

2. BODY OF PAPER

2.1 Algorithm Overview

First algorithm for Bezier clipping method was initially proposed by Nishita et al [1990]. While implementing the Nishita’s original approach we incorporated improvements by Efremov et al [2005]. Refer to Efremov et al [2005], Nishita et al [1990] for the detailed explanation of these algorithms.

We present our improvements for calculating view dependent termination criteria in section 2.1.1 and describe preprocessing optimization based on scanline clipping of Bezier surfaces in section 2.1.2. Section 2.1.2 also describes our implementation of preclipping Bezier surfaces based on Hilbert-order sequences.

2.1.1 Adaptive Computation of the Termination Criteria

The original Bezier clipping algorithm used the termination criteria determined in 2D (u,v) parameter space of a projected patch Nishita et al[1990]. This might lead to underestimation or overestimation of the result in 2D (x,y) parameter space.

Efremov et al [2005] has shown how to change the termination criteria to work in the 2D (x,y) coordinate space:

1. On each Bezier clipping iteration when calculating the signed distances $d_{i,j}$ we determine the minimum and maximum values d_{min} and d_{max} . That gives us the maximum distance span $(d_{max} - d_{min})$.
2. Termination criteria is now applied to the size of the maximum distance span: once its value becomes less than ϵ we stop Bezier clipping iterations in this direction.

Basic idea of computing the termination criteria ϵ is depicted in the figure 2:

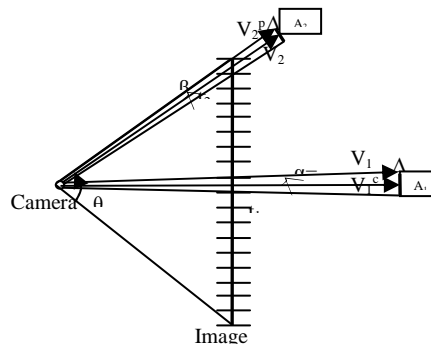


Figure 2. Scheme of termination criteria computation. Distances t_1, t_2 from the ray origin to the object AABB are the same but the intervals A_1, A_2 that are mapped to the center and corner pixels respectively are different: the corner one is smaller. That means that the value of the termination criteria for the corner pixels will be smaller (more precise).

Let t be the distance from the ray origin to the point of intersection with the object’s AABB, θ is the camera’s field of view parameter.

Then value ε of termination criteria can be computed by the following equation:

$$\varepsilon = k \cdot t \cdot \sin \alpha \quad (3.2.1)$$

Where α is an angle between the ray, which goes through the center of a pixel and the ray, which goes through the pixel's border (Figure 2). The coefficient $k \in R$ is used to control accuracy and can be set to any value from the interval $(0, 1]$. If antialiasing techniques are not used we can set k to 1 (Efremov et al[2005]).

Efremov et al [2005] suggested considering the smallest possible angle α which will correspond to the pixel on the corner of the virtual screen.

If we use the smallest possible angle α for all pixels we will overestimate the results for all pixels except the border one thus wasting time.

We suggest considering the pixel coordinates to calculate termination criteria by using the following equation:

$$\varepsilon_k = k \cdot t \cdot |\sin \beta| \cdot \frac{\|V_i^c\|}{\|V_i^p\|}, k \in \{x, y\} \quad (3.2.2)$$

Where V_i^c – vector from the ray origin to the pixel center, V_i^p –vector from the ray pixel to the pixel border (see Figure 2).

This equation lets us get the precise value of the termination criteria but calculating the sinus for each pixel might be expensive. To reduce overhead we suggest precomputing values ε_{min} , ε_{max} for center and corner pixels and then using linear interpolation to get ε for all intermediate pixels:

$$\varepsilon_k = \varepsilon_k^{\max} + (\varepsilon_k^{\min} - \varepsilon_k^{\max}) \cdot \frac{2 \cdot |k - \text{width}/2|}{\text{width}}, k \in \{x, y\} \quad (3.2.3)$$

Calculations in this case will be way less expensive while the difference between precise and interpolated values doesn't exceed 5% as shown in the figure 3:

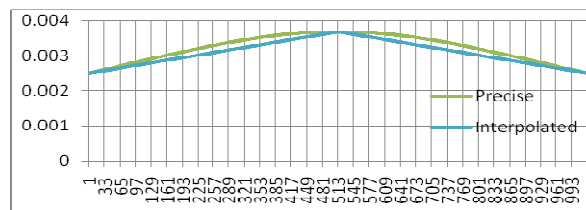


Figure 3. Difference between precise and interpolated values of termination criteria on scene Teapot (refer to Fig.1 for description, 1024*1024 pixels, camera FOV $6\pi/5$). The x-axis represents pixel index, y-axis represents value of termination criteria.

Also note that interpolated value is slightly less than the precise one thus we lightly overestimate the result and guarantee the image quality.

2.1.2 Cached Scanline-based Bezier Clipping

Ray tracing lets us get precise results but calculations are very expensive because we have to process each pixel separately. In comparison to ray tracing scanline methods give us better performance efficiency since they allow processing multiple pixels at once.

The description of scanline methods for rendering rational Bezier surfaces can be found in Nishita et al [1991], Lane et al[1980]. Nishita et al[1991] proposed an algorithm that is also based on Bezier clipping.

We suggest using Bezier clipping on scanline to preprocess Bezier patches of the image and save a couple of subdivision iterations.

The idea of this method was first mentioned by Nishita et al [1990] but this method hasn't been implemented yet.

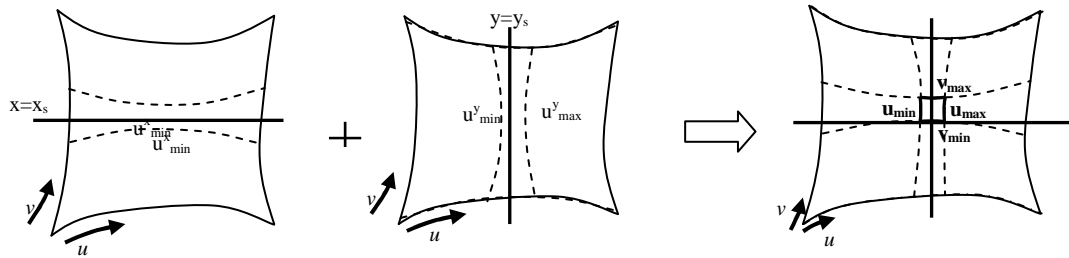


Figure 4. Clipping patch for pixel based on scanline intervals of interest. We compute intervals of interest for image scanline $x = x_s, y = y_s$, join it to get interval of interest for pixel (x_s, y_s) and then clip patch.

Patch cache-based preclipping for pixel (x_s, y_s) is described below (figure 4):

1. As described by Nishita et al [1991] for scanlines $x = x_s, y = y_s$ of rendered image calculate parameters $u_{min}^k, u_{max}^k, v_{min}^k, v_{max}^k, k \in \{x, y\}$ such that intersection will be in the interval of interest $[u_{min}^k, u_{max}^k], [v_{min}^k, v_{max}^k]$ (figure 4).

2. For pixel (x_s, y_s) subpatch bounded by parameters $[u_{min}, u_{max}], [v_{min}, v_{max}]$ is defined as follows (figure 4):

$$u_{min} = \max(u_{min}^{x_s}, u_{min}^{y_s}), u_{max} = \max(u_{max}^{x_s}, u_{max}^{y_s}), v_{min} = \max(v_{min}^{x_s}, v_{min}^{y_s}), v_{max} = \max(v_{max}^{x_s}, v_{max}^{y_s}) \quad (3.3.1)$$

3. Using De Casteljau algorithm we clip the patch in the parameters $[u_{min}, u_{max}], [v_{min}, v_{max}]$.

Steps 1-3 are repeated until the desired size of the subpatch is reached or no intersection case is being determined (if $u_{min} > u_{max}$ or $v_{min} > v_{max}$). In the former case the ray tracing Bezier clipping method described in previous section is used to find the ray-patch intersection.

This method allows us to improve the performance if the surfaces intersect 4 or more pixels on the scanline.

To use this algorithm we have to store preprocessed pairs $[u_{min}^x, u_{max}^x], [v_{min}^x, v_{max}^x]$ ($[u_{min}^y, u_{max}^y], [v_{min}^y, v_{max}^y]$) for each scanline/patch pair. It's obvious that storing of these values for all pairs will require a lot of memory space. Thus we have to use some data structure with moderate memory usage that makes it possible to extract those values for currently rendered pixel (x_s, y_s) .

A structure of cache table item for storing information about intersection of x- (or y-) scanline with a patch is presented in the listing 1.

```

struct CacheLine
{
    int scanline;//scanline number
    int patchIndex;//Bezier patch index
    pair<float, float> uPair;//range of interest[u_min, u_max]
    pair<float, float> vPair; //range of interest [v_min,v_max]
    bool missed;//flag indicating intersection or no-intersection case
}
    
```

Listing 1. Data structure to store patch/scanline intersection in the cache table

In our approach we use the cache tables of fixed size for x- and y- scanlines that store the values used latest. We locate the position of the (x, y) pixel in the cache table using hashing algorithm described by Dietrich, Slusallek [2007]. There are cases when we have several candidates to the same cache table row; we solve that collision by storing only last used pixel row or column. In our experience, this approach didn't give any performance hit due to low number of collisions. Thus we dynamically replace old unused entries with current ones that most probably will be used again.

To maximize cache table hit rate we pass through image pixels in a special way, using properties of Hilbert curves. Detailed description of Hilbert curves and their properties can be found in Warren [2002]. Image is divided into squared parts and pixels in each part are passed in Hilbert order.

To get intervals of interest $[u_{min}, u_{max}], [v_{min}, v_{max}]$ for the pixel (x_k, y_k) we look up for the $[u_{min}^x, u_{max}^x], [v_{min}^x, v_{max}^x]$ in the x- cache table, $[u_{min}^y, u_{max}^y], [v_{min}^y, v_{max}^y]$ in the y-cache table and then use formulas (3.3.1).

If preprocessed interval of interest can't be found in the x- or y- cache table we recompute it and store in the table.

Using this approach we can get very high hit rate by using cache tables of relatively small size even for images with very high resolution. Empirically we came up with 4096-8192 optimal common cache table size (256-512 cache table size works great for per-thread cache tables). The cache size/hit rate dependency table for test scenes of different sizes is shown in the following table (Table 1):

Table 1. Cache size / cache hit rate dependency.

Cache table size	Hit rate (%)		
	Teapot	Head	Tahoro
1	37.44	37.1	35.61
16	85.44	78.65	70.45
256	96.93	89.15	76.49
1024	98.91	92.13	79.28
4096	99.21	93.71	80.96
16384	99.29	94.39	81.77
65536	99.29	94.53	82.14

2.2 Results and Discussion

We implement our method using SSE to utilize CPU's SIMD units. We also implemented efficient tiling (binning) based parallel version.

We tested our algorithm on the different models. "Head" and "tahoro" models were converted from Catmul-Clark models (copyright CGNetworks). The rest are free SPatch models from various artists.

Using adaptive termination criteria gives us 1-4% performance improvements (depending on the scene geometry and camera field of view) but also gives us the guarantee of image quality within given tolerance. In the unoptimized method we used approach proposed by Efremov et al. [2005]. It is interesting to note that usually user tends to specify more conservative tolerance value compared to adaptive one so typical speed up can be of 10-15%.

Scanline-based patch preprocessing gives more impressive performance improvements (6 - 45%) shown in table 2. This improvement works better on objects with big patches. Speed up from caching increases with bigger image resolution as patch covers more pixels and we save more Bezier clipping iterations. Good speed up was reached not only because of patch subdivision but also because of detecting no-intersection case for multiple pixels. It's worth mentioning that due to the efficient binning and robust z-buffering performance doesn't depend greatly on the number of patches in the scene.

Table 2. Performance comparisons

Scene	Unoptimized method,[ms]	Using adaptive termination criteria with interpolation [ms] /speed up [%]	Cached Bezier clipping method [ms] / speed up [%]	Both optimizations [ms] / speed up [%]
teapot	55.24	54.61/1.1%	35.19/36.3%	35.19/36.3%
kirche	85.08	84.20/1.0%	64.89/23.7%	64.12/24.6%
pyramid	207.12	198.97/3.9%	112.88/45.5%	108.03/47.8%
kitchen	114.9	113.72/1.0%	88.27/23.2%	88.01/23.4%
tahoro	94.23	92.70/1.6%	88.92/5.6%	88.28/6.3%

3. CONCLUSION

In this paper we introduced several important improvements of the original Bezier clipping method for direct rendering of rational Bezier surfaces.

The presented method for adaptive calculation of termination criteria ϵ allows us to control the accuracy of output image and guarantees that the result will be within the given tolerance. We developed the way to speed-up the process of rendering by introducing new termination criteria that greatly reduced the number of iterations in the original Bezier clipping method.

We also evaluated and improved the method for preprocessing of the input Bezier patches by clipping on scanline, initially mentioned by Nishita et al [1990]. Our implementation gave us significant performance improvements up to 45%.

As a result we achieved real-time performance for high image resolution on the commodity CPU even for complex scenes with hundreds of thousands of patches. This is the best performance on the CPU we are aware of.

While the performance of our algorithm might be smaller for simple scenes rendered on GPU (Eisenacher et al [2008]) it can be used for rendering extremely complex scenes.

Our future research incentives include OpenCL* implementation to directly compare performance of our method on various OpenCL enabled devices like CPUs and GPUs.

REFERENCES

Book

David Salomon, 2006. *Curves and Surfaces for Computer Graphics*. Springer, USA.

Henry S. Warren, 2002. *Hacker's Delight*. Addison-Wesley, San Francisco, USA.

Journal

Nishita et al, 1991, Scanline algorithm for displaying trimmed surfaces by using Bezier clipping. *The Visual Computer*, Vol.7, No.5, pp.247-258.

Kajiya, 1983, New Techniques for Ray Tracing Procedurally Defined Objects. *ACM Transactions on Graphics*. Vol.2, Issue 2, pp.161-181.

Piegl, 1986, A Geometric Investigation of the Rational Bezier Scheme of Computer Aided Design. *Computers in Industry*. Vol.7, Issue 5, pp.401-410.

Wang et al, 2000, An Efficient and Stable Ray Tracing Algorithm for Parametric Surfaces. *Journal of Visualization and Computer Animation*. Vol. 11, No.4, pp.209-219.

Wang et al, 2002, An improved rendering technique for ray tracing Bezier and B-Spline surfaces. *Journal of Information Science and Engineering*. Vol. 18, No.4, pp.541-561.

Lane et al, 1980, Scan Line Methods for Displaying Parametrically Defined Surfaces. *Communications of the ACM*. Vol.25, Issue 1, pp.22-34.

Campagna, Slusallek, 1996, Improving Bezier clipping and Chebyshev boxing for ray tracing parametric surfaces. *The Visual Computer*, Vol.13, No.6, pp.265-282.

Conference paper or contributed volume

Efremov et al, 2005, Robust and numerically stable Bezier clipping method for ray tracing NURBS surfaces. In *SCCG '05: Proceedings of the 21st spring conference on Computer graphics*. Budmerice, Slovakia, pp. 127-135.

Nishita et al, 1990, Ray tracing trimmed rational surface patches. In *SIGGRAPH '90: Proceedings of the 17th annual conference on Computer graphics and interactive techniques*. Dallas, TX, USA.

Nishita, 1998, Applications of Bezier Clipping Method and Their Java Applets. In *SCCG: Proceedings of Spring conference on computer graphics*. Budmerice, Slovakia, pp.3-15.

Kajiya, 1982, Ray tracing parametric surfaces. In *SIGGRAPH'82: Proceedings of the 9th annual conference on Computer graphics and interactive techniques*. Boston, MA, USA.

Toth, 1985, On ray tracing parametric surfaces. In *SIGGRAPH'85: Proceedings of the 12th annual conference on Computer graphics and interactive techniques*. San Francisco, CA, USA.

Whitted, 1978, A scan line algorithm for computer display of curved surfaces. In *SIGGRAPH'78: Proceedings of the 5th annual conference on Computer graphics and interactive techniques*. Atlanta, GA, USA.

Christian Eisenacher et al., 2009 Real-Time View-Dependent Rendering of Parametric Surfaces. *I3D '09: Proceedings of the 2009 Symposium on Interactive 3D Graphics and Games*, Boston, USA, 2009.

Dietrich, Slusallek, 2007, Adaptive Spatial Sample Caching. *IEEE Symposium on Interactive Ray Tracing*, Germany, 2007.

3D DOCUMENTATION OF NATURAL HERITAGE FOR VIRTUAL ENVIRONMENTS AND WEB — CASE STUDY: VALLEY OF GEYSERS, KAMCHATKA

Andrey Leonov¹, Alexander Aleynikov², Dmitriy Belosokhov³, Alexander Bobkov³, Evgeny Eremchenko⁴, Pavel Frolov⁴, Andrey Klimenko¹ and Stanislav Klimenko⁴

¹*Vavilov Institute for the History of Science and Technology of the RAS, Moscow, Russia*

²*R&D Center ScanEx, Moscow, Russia*

³*Moscow Institute of Physics and Technology, Dolgoprudny, Moscow region, Russia*

⁴*Institute of Computing for Physics and Technology, Protvino, Moscow region, Russia*

ABSTRACT

The article presents a project on a 3D documentation of the Valley of Geysers, Kamchatka (UNESCO Natural Heritage). The 3D document serves for virtual ecotourism, ecological education, scientific modeling and visualization.

KEYWORDS

Virtual Reality, Natural Phenomena Modeling, Scientific Visualization, Neogeography, Situational Awareness

1. INTRODUCTION

This paper presents a project of development of the 3D document “Virtual Valley of Geysers”. The terrain is modeled using high resolution satellite imagery. The content includes texts, photos, videos, vector models and dynamic animations. Scientific data like seismic activity, geothermal system structure and thermal water outflows catalogue is visualized.

The principal scheme of a 3D document was enhanced to support two representations of the 3D document, one for public Web access and the other — for Virtual Environment (VE) stereo projection systems. Both versions represent generally the same information but use different terrain models. The VE version is developed using open source software OpenSceneGraph (OSG) [OSG] and AVANGO [AVANGO]. The Web version is based on Google Earth and is publicly available at www.valleyofgeysers.com.

The project contributes to the development of approaches for building Virtual Natural Heritage environments. It also provides a shared cyberspace for effective interdisciplinary communication of researchers from different branches of science.

2. 3D DOCUMENTATION OF NATURAL HERITAGE

The principal scheme of a 3D document is shown on 0Generally, it is a 3D model of an object and an information system linked to the 3D model. Set of information also often includes a large amount of semi-structured data, embedded in the 3D model directly or via linking files.

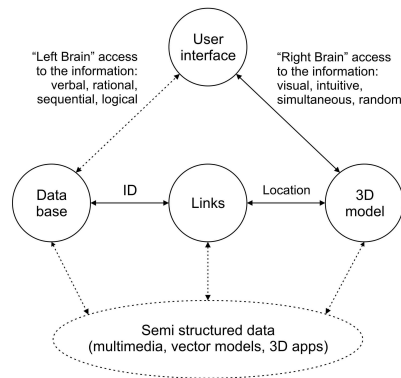


Figure 1. The principal scheme of a 3D document.

The concept of a 3D document is presented in the publications of Dieter Fellner [Fellner, 2007], Fabio Remondino [Remondino, 2010] and others. Methods of 3D documentation are being developed most actively in the field of Virtual Cultural Heritage — virtual museums, archaeological reconstructions, visualization of complex architectures [Havemann, 2009], [Remondino, 2009]. The same principle could be implemented for the development of Virtual Natural Heritage applications.

First of all, Natural Heritage site is a landscape. So the 3D document in this case is based on a 3D model of a terrain. The terrain can be reconstructed precisely using historical topographic maps, satellite data, aerial photogrammetry or aerial laser-scanning depending on technical and financial constraints [Boehler, 2001]. Aerial methods are more precise and thus better, especially for small areas [Buckley, 2008]. Reconstruction of the terrain for large areas usually involves satellite imagery [Chen, 2010], [Deng, 2008], [Mao, 2008]. The methods can be combined to create a 3D model of a large territory with enhanced resolution for some particular areas of interest.

Given the precise and geo-referenced 3D terrain is built, further process of 3D documentation of a natural area involves development of an information system connected with the 3D terrain and building user interface to provide access to the 3D document.

3. RELEVANCY

The Valley of Geysers is the second largest geyser field in the world [Sugrobov, 2009], [Bryan, 1991]. It is located in Kamchatka peninsula on the Russian Far East, in the Kronotsky Reserve, that is a part of the UNESCO World Heritage Site “Volcanoes of Kamchatka” (N 765 bis). In 2008 it was voted as one of the seven Wonders of Russia.

The Valley of Geysers is a renowned tourist destination. But its average attendance is only about 3000 people a year because of its remote location and reserve status. It is only a tiny percentage of those who would like to see it. It makes necessary to develop virtual ecotourism for this place.

The massive landslide on June 3, 2007 has dramatically changed the landscape of the Valley of Geysers. About half of all geysers were destroyed. A large set of interdisciplinary research was started after 2007 in order to analyze and forecast further landslide hazard [Pinagina, 2007], [Leonov, 2007], [Lobkov, 2008], [Dvigalo, 2009], [Kugaenko, 2010]. Careful 3D documentation of scientific data as well as physical modeling and visualization is needed to provide effective interdisciplinary communication.

As some of the UNESCO World Heritage sites change irretrievably due to natural processes, they can be preserved only in the form of evolving 3D documents. Precise reconstruction with use of Neogeography [Turner, 2006] and VE will allow saving a lot of data about this areas and their dynamics.

Some results of the project were published in our previous article [Leonov, 2010].

4. FUNCTIONAL SCHEME

Precise and visually realistic reconstruction of a landscape is a necessary condition to achieve good immersion effect in VE system. But higher resolution means larger amount of data and thus more difficult Web access. It is a basic contradiction in the process of 3D documentation of large natural areas.

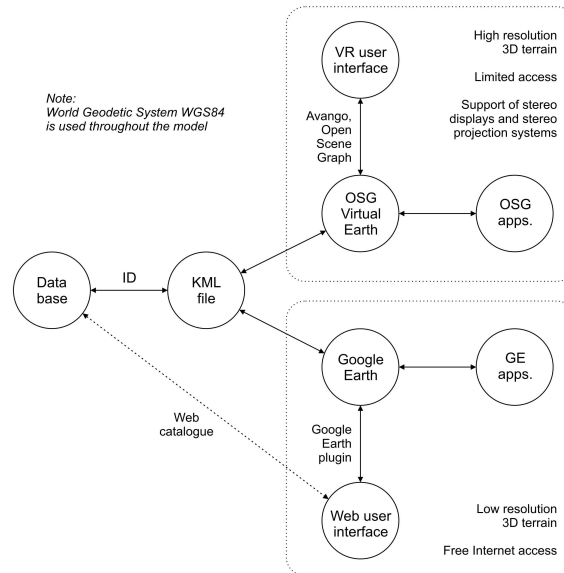


Figure 2. The scheme of a 3D document with dual representation.

To solve this contradiction, we have enhanced the principal scheme of a 3D document to support dual representation of the 3D document, one for Web access and the other for VE systems. We use precise terrain model for VE application on the local stereo projection system, and we use open Google Earth terrain to provide worldwide access to the information. The information is linked with terrain models by a KML file that provides correct positioning of data on both virtual globes (OSG and Google Earth). Both versions represent generally the same information, with some additional applications in each version, 0

5. VIRTUAL TERRAIN

The virtual terrain was built using GeoEye and Cartosat satellite imagery. The resolution of the digital elevation model (DEM) is 2.5 m and the texture resolution is 0.5 m. The imagery and DEM are courtesy of R&D Center ScanEx and GeoEye Foundation, Inc. The terrain was precisely geo-referenced in the world coordinate system ITRF/WGS84 using field GPS measurements. The coordinates of the twelve reference points were measured with an absolute accuracy of about 0.1 m. Differential corrections were implemented with use of the two backbone GPS base stations of KEMSD GS RAS. The accuracy of positioning of the terrain model in the world coordinate system reached 0.2 m on the horizontal plane and 3 m on the Z axis.

6. WEB AND VE REPRESENTATIONS

A web representation is based on the Google Earth virtual terrain, 0 Public access is organized on our website www.valleyofgeysers.com using the Google Earth plug-in. Web control menu is developed in JavaScript. The menu is generated automatically on the web page using information from the KML file. The menu allows control of the 3D document and switching on/off visibility of layers and objects. The size of the initial KML linking file is only about 500 KB, all other information is downloaded on demand (after a layer is set visible or an object description is opened). It makes it easy to browse the 3D document on-line. Rough terrain

model is the main drawback of the public Web representation. Google Earth terrain data (SRTM 90 m [SRTM]) cannot be refined by the user. Low resolution of the virtual terrain reduces dramatically the effect of immersion. Moreover, Google Earth software does not support a stereo mode. The main aim of the public Web representation is to provide free worldwide access to the information about the Valley of Geysers.

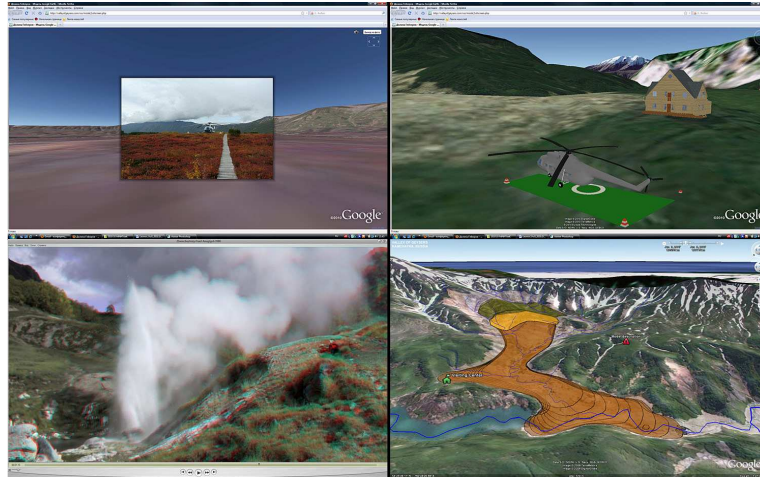


Figure 3. Examples of data in the Web version: georeferenced photo, 3D models, 3D video, and landslide animation.

A VE representation is developed in OSG and AVANGO software. It uses the high resolution virtual terrain model embedded in the OSG Virtual Globe. The KML file links the terrain model and the web information system. Thus users of the VE application have full web access to the same information as Web users. A 3D mouse with six degrees of freedom is used for free travel through the terrain. Limited access is the main drawback of the VE representation. A large amount of the high resolution terrain data requires dedicated web server to provide comfortable web access. This version is intended for stereo presentation and provides good immersion effect. It is now demonstrated in the public museum of the Kronotsky Reserve. Several specific 3D applications such as geothermal system structure, 3D geyser activity animation, landslide models, and local seismic activity visualization are also being developed in the VE representation.

7. CONCLUSIONS

We presented a dual approach to 3D documentation of a complex Natural Heritage site with the single information system both for VE applications and for Web access. This approach was implemented for a case study of high scientific and cultural importance: the Valley of Geysers, Kamchatka. The developed 3D document is used for virtual ecotourism, ecological education, scientific modeling and visualization.

ACKNOWLEDGMENT

The project is run by a team based on the Institute of Computing for Physics and Technology (ICPT), Russia. We are supported by the Kronotsky State Natural Biosphere Reserve, the Institute of Volcanology and Seismology of Far Eastern Branch of RAS (IVaS FEB RAS), the Kamchatkan Experimental and Methodical Seismological Department of Geophysical Service RAS (KEMSD GS RAS), R&D Center ScanEx. Financial support is granted by the Russian Foundation for Basic Research (RFBR). Important satellite imagery was granted by GeoEye Foundation, Inc. We wish to express our gratitude to all the people supporting us.

REFERENCES

- ACANGO website: <http://www.avango.org/>
- Boehler, W. et al, 2001. Topographic Information in Cultural and Natural Heritage Visualization and Animation. *Archives of ISPRS*, Vol. XXXIV-5/W1, pp. 56-61.
- Bryan, S. T. et al, 1991. *The geysers of the Valley of Geysers*. A special report of GOSA Transactions. B&J Printing, California, USA.
- Buckley, S. J. et al, 2008. Oblique helicopter-based laser scanning for digital terrain modeling and visualisation of geological outcrops. *Archives of ISPRS*, Vol. XXXVII-B4, pp. 493-498.
- Chen, B. et al, 2010. VCUHK: integrating the real into a 3D campus in networked virtual worlds. *Proc. of 2010 International Conference on Cyberworlds*, IEEE CS CPS, pp. 302-308.
- Deng, B. et al, 2008. Remote sensing analysis of the status of the Beijing-Hangzhou Grand Canal. *Archives of ISPRS*, Vol. XXXVII-B5, pp. 231-236.
- Dvigalo, V. N. and Melekestsev, I. V., 2009. The geological and geomorphic impact of catastrophic landslides in the Geyser Valley of Kamchatka: Aerial photogrammetry. *In Journal of Volcanology and Seismology*, Vol. 3, No. 5, pp. 314-325.
- Fellner, D. W. et al, 2007. Guest Editors' Introduction: 3D documents. *In IEEE Computer Graphics and Applications*, Vol. 27, No. 4, pp. 20-21.
- Havemann, S. et al, 2009. The Arrigo Showcase Reloaded — towards a sustainable link between 3D and semantics. *In Journal on Computing and Cultural Heritage*, Vol. 2, Issue 1, p. 1-13.
- Kugaenko, Yu. A. et al, 2010. Local Seismicity Within the Valley of the Geysers: Results from the 2008-2009 Field Investigation. *In Bulletin of Kamchatka Regional Association "Educational-Scientific Center", Earth Sciences*, No. 1 (15), pp. 90-99.
- Kugaenko, Yu. A. et al, 2010. Deep Structure of the Region of the Uzon-Geyser Volcanic-Tectonic Depression Based on the Data of Microseismic Sounding. *In Doklady Earth Sciences*, Vol. 435, Part I, pp. 1460-1465.
- Leonov, A. V. et al, 2010. Virtual story in cyberspace: Valley of Geysers, Kamchatka. *Proc. of 2010 International Conference on Cyberworlds*, IEEE CS CPS, pp. 247-253.
- Leonov, V. L., 2007. Geological preconditions and capability to forecast the landslide of June 3, 2007 in the Valley of Geysers, Kamchatka. *Proc. of scientific and technical conference "Geophysical Monitoring and Problems of Seismic Safety of Russian Far East"*, Geophysical Service RAS, Vol. 1, P.-Kamchatskiy, Russia, pp. 91-95.
- Lobkov, E. G. and Lobkova, L. E., 2008. Ecological effects of the landslide in Valley of Geysers June 3, 2007 (first season after). *Proc. of VIII international scientific conference "Conservation of biodiversity of Kamchatka and coastal waters"*, P.-Kamchatski, Russia, pp. 114-140.
- Mao, F. et al, 2008. The research and application of spatial information technology in cultural heritage conservation — case study on Grand Canal of China. *Archives of ISPRS*, Vol. XXXVII-B5, pp. 999-1005.
- OpenSceneGraph website: <http://www.openscenegraph.org/>
- Pinegina, T. K. et al, 2008. The Valley of Geysers, Kamchatka after the June 3, 2007 Disaster. *In Vestnik DVO RAN*, No. 1, pp. 33-44.
- Remondino, F. and Rizzi, A., 2010. Reality-based 3D documentation of natural and cultural heritage sites — techniques, problems, and examples. *In Applied Geomatics*, Vol. 2, No. 3, pp. 85-100.
- Remondino, F. et al, 2009. 3D virtual reconstruction and visualization of complex architectures — the "3D-ARCH" project. *Archives of ISPRS*, Vol. XXXVIII-5/W1.
- SRTM 90 m Digital Elevation Data: <http://srtm.csi.cgiar.org/>
- Sugrobov, V. M. et al, 2009. *Pearl of Kamchatka — the Valley of Geysers*. Kamchatpress, P.-Kamchatskiy, Russia.
- Turner, A. J., 2006. *Introduction to Neogeography*. O'Reilly Media, USA.

MEASURE OF INDIVIDUAL EYE-HEAD COORDINATION COMPARED IN REAL AND VIRTUAL ENVIRONMENT

Gildas Marin¹, Marine Faure¹, Marc le Renard² and Damien Paillé¹

¹*Essilor International, 57, av. de Condé, 94106 Saint-Maur, France*

²*ESIEA Ouest, rue des docteurs Calmette & Guérin, 53000 LAVAL France*

ABSTRACT

The eye-head coordination is a physiological behavior specific to each individual. This behavior is of high interest for the design of ophthalmic lenses. In this study, the natural eye-head behavior of subjects has been measured in reality and in a similar situation reproduced in a virtual reality ophthalmic lens simulator. Similar behavior has been measured in both situations: the natural behavior of people about their visual strategy is not disturbed by virtual reality, thus, validating the full potential of virtual reality as a new and complementary method of behavioral measurement in our simulator.

KEYWORDS

Virtual Reality, Ophthalmic optics, ophthalmic lens simulator, eye-head coordination.

1. INTRODUCTION

An active exploration of the environment implies eye and head movements. These constant gaze adjustments aim at maintaining a particular point of our visual environment on each eye fovea in order to get a clear and detailed vision. This task requests a precise coordination between the movements of each eye and the head. This coordination is individually acquired during the early stage of development. Each individual will develop its own strategy. Thus, to identify an object in the peripheral visual field, some people mostly move the head and very few the eyes. Others have the opposite behavior. In the literature, we find the distinction between subjects that move more the head, defined as « Head-mover » and those that move more the eyes, defined as « Eye-mover » [Afanador 1986]. These two extremes are linked with a continuum of values describing all the possible behaviors. This has been used for the conception of personalized progressive eyeglass lens [Devie 2003]. The precise visual strategy of each person is measured thanks to a dedicated tool: the VisionPrint System™ (VPS). This device measures the « eye-head coefficient », which is the ratio between the movement amplitude of the head and the eccentricity of the displayed visual target. A « stability coefficient » is also computed by the VPS, corresponding to the standard deviation of the behavior.

We have developed and validated a virtual lens simulator that reproduce the optical effects seen through spectacles on virtual scenes, with the final goal of drastically reducing the number of lens prototypes needed for the development of new designs [Marin 2006 and 2008]. Another objective is to study the physiological behaviors of spectacle wearers. For this purpose, we have developed a virtual scene, in which we can measure the eye-head coefficient, with a virtual VPS. This scene is an exact reproduction of the environment used in reality for this kind of measure.

The differences between reality and virtual environments however, such as artificial stereoscopy, photorealism, transport delay, etc... are all potential factors that could modify the immersive feelings and natural behavior of subjects [Fuchs 2003] and especially may highly impact their natural eye-head behavior.

Previous study performed in virtual environments demonstrated that the presence of a dynamic environment, when measuring the eye-head coefficient, did not change subjects' behaviors. The measurement remained stable whatever the spatio-temporal combinations used [Faubert 2007] however, they did not compare their virtual cases to real cases.

The goal of the current study was to compare eye-head coordination strategy in real and virtual environments by measuring subjects' coefficients in exactly the same conditions using both real and virtual

VPS. This experiment aims at consolidating the use of virtual reality as a new and complementary method of natural behavioral measurement, especially for the ophthalmic field.

2. MATERIAL AND PROTOCOL

2.1 The VisionPrint System™

The VisionPrint System™ measures head movements thanks to a 120Hz ultrasonic tracker worn by the subject while making saccades to peripheral targets (Figure 1a). The subject is seating in front of the central LED, looking at the stimuli line. The distance between the device and the base of the nose is approximately 40cm (the subject adjusts his position by moving away or closer of the device in order to center a position cross on the control screen). The gaze lowering when looking at the central LED should be approximately equal to 30° (Figure 2a). In this position, peripheral LEDs have an eccentricity of $\pm 40^\circ$ from the subject. Once in the correct position, the person is asked to reach the peripheral targets as naturally as possible while maintaining his position.

Sequence for a measuring session: the subject looks at the central LED. Then the central LED is turned off and a peripheral target is turned on. The task of the subject is to look the more naturally to the peripheral target. Then the peripheral target is turned off and the central target is turned on; and so on.

During the full session 25 peripheral targets (LEDs) are displayed. For the first 5 targets, the peripheral LEDs are switched on/off in a sequential order between left and right as a short learning session of the task. The next 20 peripheral targets are switched on/off in a randomized order for the measurement itself. The sequence occurs in the following manner: A beep sounds while the central LED switches on (beep sounds 1s whereas the LED remains switched on during 2s with a variation of ± 500 ms to avoid learning effect). The subject is requested to look at the central LED. At the switch off of the central target, one of the two peripheral LEDs switches on. The subject has to do a gaze saccade to the corresponding peripheral LED. A new beep sounds once the peripheral LED is switched off and the central led switches on again...

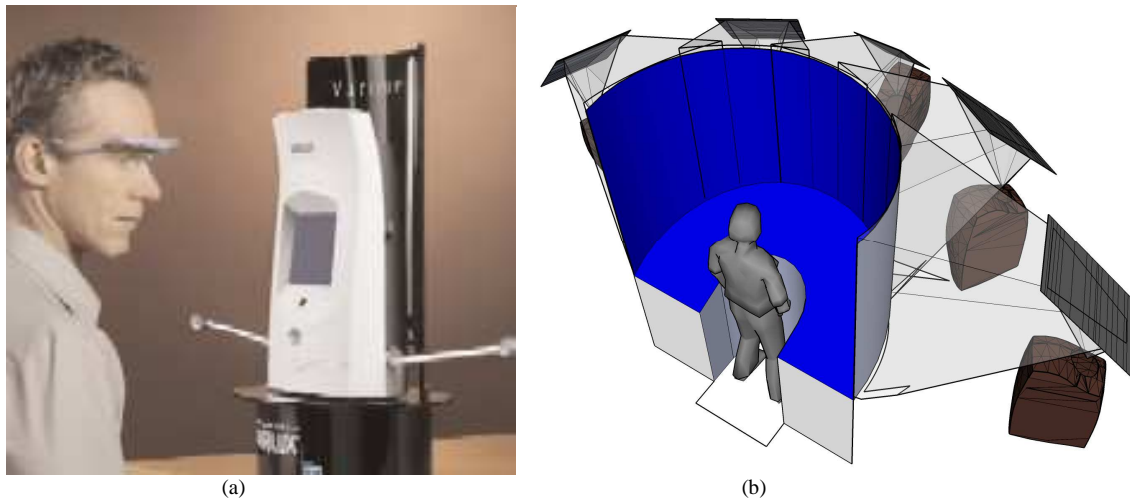


Figure 1. a) The VisionPrint System™ (VPS) and b) the display system of our simulator (Courtesy of Barco).

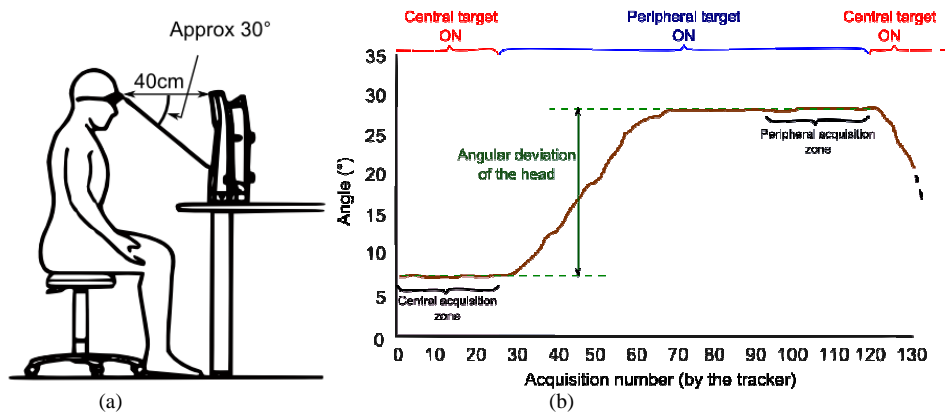


Figure 2. a) Posture for the use of the VPS. Peripheral LEDs, once the subject well-positioned in front of the device, have an eccentricity of $\pm 40^\circ$. b) Calculation of the head deviation for a peripheral target

The calculation of the eye-head coefficient value is performed in the following manner: The acquisitions of the head position preceding the switch on of the peripheral target (corresponding to the “central acquisition zone” on figure 2b) are taken into account to calculate the mean centered position of the head. The acquisitions preceding the extinction of the peripheral target (corresponding to the “peripheral acquisition zone” on figure 2b) are taken into account to calculate the final mean position after rotation of the head. The difference between those two values represents the angular rotation of the head. This makes the result independent of the angular value in central position that depends on both the positioning of the tracker on the head and the attitude of the subject.

2.2 The Virtual Reality Simulator

Our simulator has been designed to cover the full visual field, 200° in horizontal and 135° in vertical, while trying to minimize the visual discomfort generated by artificial stereoscopy. The display screen is a vertical semi-cylinder of 1 m radius, cut at the bottom by a tilted plane at 1 m of the user point of view (Figure 1). The head position is tracked at 55Hz by the Advanced Realtime Tracking (ART) infrared system.

The stereoscopic display is ensured by six full HD video projectors (1920x1080 pixels at 110Hz) using active stereoscopy with CrystalEyes shutter glasses. Our refraction room, used for the measurements in reality, has been entirely remodeled in the simulator (Figure 4).

3. EXPERIMENTAL STUDY

The eye-head coefficient and stability coefficient are measured for each subject in two different positions of the device, the VPS in front of the wall in the room (“wall” condition) vs. the device in the middle of the room (“room” condition), representing two typical conditions observed in optical shops. Those two conditions are repeated both in real and in virtual environment (Figure 4). The order of the different conditions is randomized but measurements are consecutive. All conditions have been repeated for each subject taken as its own reference.

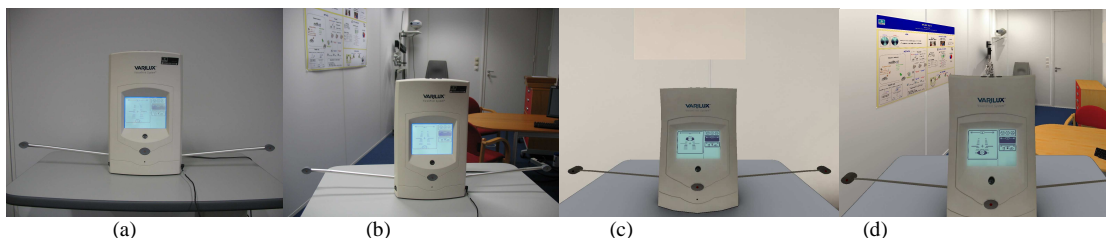


Figure 4. The refraction room in the 4 measurement conditions: real (a, b) and virtual (c, d). VPS in front of a white wall (“wall”) without interaction of the environment (a, c); VPS at the center of the refraction room (“room”) (b, d)

The population sample for this study was a group of 15 subjects (22 to 46 year old). Subjects with strabismus, scotoma, ocular pathology inducing a decrease in ocular motility capabilities, visual field or binocular visual acuity, were excluded. Were also excluded, any subject with cervical osteoarthritis, motility problems of the neck or any pathological cervical strains.

4. RESULTS

According to our statistical analysis (ANOVA), there is no significant difference between the mean of eye-head coefficients obtained in real vs. virtual conditions in both “wall” ($p=0,705$) and “room” ($p=0,935$) contexts: eye-head coefficient measured in virtual environment is statistically not different to the one measured in reality. We also don’t find any significant difference for the stability coefficients obtained in real and in virtual for either condition “wall” ($p=0,559$) and “room” ($p=0,06$). As for eye-head coefficient, the stability coefficient measured in virtual environment is not different from the one measured in reality. However, graph b) of figure 5 shows a tendency to a better stability of the measure in virtual environment. This may be due to the difference in tracking frequency between the two systems inducing a difference in the duration of each acquisition period.

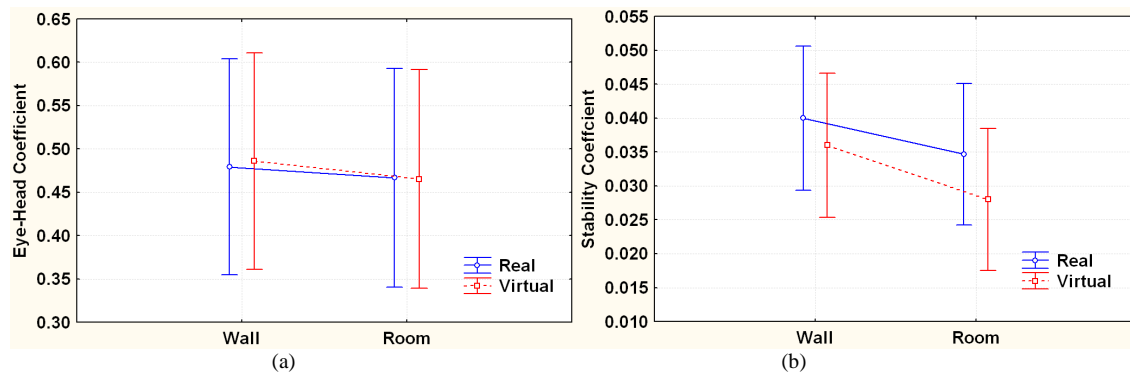


Figure 5. Mean eye-head coefficient (a) and stability coefficient (b) measured in real and in virtual for both conditions without interaction of the environment (wall) and with interaction (room)

In the real measurements, we observe no statistically significant difference between eye-head coefficients measured in front of the wall and those measured at the center of the refraction room ($p=0,239$). We get similar results for stability coefficients ($p=0,326$). Graph b) of figure 5 reveal a tendency to a better stability (lower standard deviation) while subject is looking at the center of the refraction room.

The statistical analysis of measurements carried out in virtual leads to the same conclusions than in reality: there is no significant difference neither for the eye-head coefficient ($p=0,192$) nor for stability coefficient ($p=0,08$) between the two conditions tested. However, the same tendency to a better stability while subject is looking at the center of the refraction room is observed.

The measuring environment is not impacting the eye-head coefficient but the same tendency to a better stability is observed in the “room” condition both in real and in virtual.

5. CONCLUSION

The eye-head coefficient is not different in real and in virtual environments. Moreover, the same tendencies on the stability coefficient are observed. This shows that the natural behavior of people about their visual strategy is very well preserved by virtual reality in our simulator. This measurement, moreover, could be combined with the simulation of virtual lenses in future studies. The virtual reality has now become an essential tool for our research projects because it combines efficient and fast methods with precisely controlled and varied environments. These convincing results will even reinforce the use of virtual reality for new studies aiming at a better knowledge of spectacle wearers and their physiological needs, as the example of the visual strategy.

REFERENCES

- Afanador J. et al, 1986. Eye and head contribution to gaze at near through multifocals: the usable field of view. *Am. J. Optom. Physiol. Opt.*, Vol. 63, No. 3, pp 187-192.
- Devie P. and Jouvenceau C. 2003. Varilux® Ipeo™ Eye/Head strategy and physiological personalisation. *Point de vue*. No. 49.
- Faubert J. et al, 2007. Eye/Head coordination in a full immersion virtual reality environment. Study on the influence of a dynamic environment. *Point de Vue*. No. 57, pp 23-27.
- Fuchs P. and Moreau G., 2003. *Le traité de la réalité virtuelle vol 1: Fondements et interfaces comportementales*. Les Presses de l'Ecole des mines, France.
- Marin G. et al, 2006. Validation of a lens simulator in virtual reality. *3rd European Meeting in Physiological Optics*, City University, London.
- Marin G. et al, 2008. Compared Distortion Effects between Real and Virtual Ophthalmic Lenses with a Simulator. *15th ACM Symposium on Virtual Reality Software and Technology*, Bordeaux, France.

BUBBLING IN LIQUIDS

H. Sato and Y. Kawaguchi
The University of Tokyo

ABSTRACT

A technique has been developed to model gas in fluid on the basis of a moving particle simulation (MPS) method, using the example of bubbles rising in sodas that are dissolved in gas. The MPS method does not need grids, so can handle topology changes naturally, which is appropriate for animating fluids. Additionally, we applied a discrete element method (DEM) to the collision interaction of discrete bubbles. In addition, the motion of the discrete bubbles includes buoyancy, cohesive forces, and drag interactions with the liquid. Moreover, the amount of gas transferred from a continuum fluid model to the discrete bubble model at nucleation sites varies in accordance with the impact of collision between particles. The amount affects the size and shape of bubbles. As a result, our technique can animate the realistic pouring of a glass of sodas.

KEYWORDS

Moving particle simulation (MPS), discrete element method (DEM), bubbles

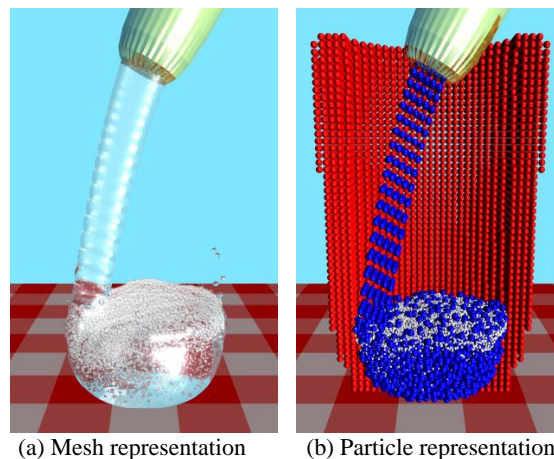


Figure 1. Results of simulating bubbles

1. INTRODUCTION

Many researchers have simulated liquids not only in computer graphics, which focus on visual effects, but also engineering, which focuses on numerical analysis. There are two major approaches for making computer graphics of fluids: the grid based method and the particle based method. The particle based method can produce simple simulations of fluids like this case that have large deformations since the particle method does not need a grid. Thus we adopted the moving particle simulation (MPS) method, which is a particle method.

In computer graphics, we usually can see two types of bubbles: air bubbles and bubbles from gas dissolved in the fluid. In the former case, bubbles are created by trapped air in real life, but in a simulation they are computed by a particle method using two types of fluid particle that have different densities. On the

other hand, the latter uses particles that have gas concentrations and computes water and bubbles separately through volume force. When topology changes, especially for air bubbles, a grid is needed. However, bubbles are approximated as a fixed shape in this case. In the MPS method, influence radius r_e of weight function can change easily. Thus, it is applied to a differential equation easily.

In accordance with the above, we use MPS to simulate the action of bubbles in fluid. In this paper, this method is used to realistic animate a realistic glass of soda being poured.

2. PREVIOUS WORK

Smoothed-particle hydrodynamics (SPH) is another particle method commonly used in computer graphics to simulate complex liquid. Müller et al. [2005] modelled fluid-fluid interaction on the basis of SPH and simulated air-water interaction. Cleary et al. [2007] simulated bubbles being generated from gas dissolved in fluid, and Ihmsen et al. [2011] presented a physical-based multiphase model for simulating water and air bubbles with SPH. Also, Hong et al. [2008] proposed a hybrid method by incorporating a bubble model based on SPH into an Eulerian grid simulation and animated lively moving bubbles.

On the other hand, MPS has been proposed as a particle method by Koshizuka et al. [1995; 1996] and used in a wide sphere. There has also been the Lung Deformation Simulation [Ito et al. 2010] and numerical analysis of shipping water onto a moving ship [Shibata et al. 2009]. In computer graphics, Tanaka et al. [2006] presented the rigid body simulation, and Kondo et al. [2007] simulated elastic objects.

To animate bubbles, there are other ways to simulate not only their motion but also their shape. Zheng et al. [2006] presented a novel framework based on a continuous fluid simulator for general simulation of realistic bubbles. There have also been techniques for simulating and rendering liquid foams [Kück et al. 2002], simulating boiling [Mihalef et al. 2006], and a method for including the visual effect of bubbles [Greenwood and House 2004]. Hong and Kim [2003] combined the volume of the fluid method and front-tracking method to naturally manipulate of topology changes and remove the numerical diffusion.

3. FLUID DYNAMICS

We summarize the moving particle simulation (MPS) method. The MPS method discretises the governing equation of continuous body using an interparticle interaction model corresponding to differential operators. Details of the method are explained by Koshizuka et al. [1995] and Koshizuka and Oka [1996]. Conservation of momentum is the governing equation of the MPS methods:

$$\rho \frac{D\mathbf{u}}{Dt} = -\nabla p + \nu \nabla^2 \mathbf{u} + \mathbf{f}, \quad (1)$$

Where ρ is the density, \mathbf{u} is the velocity, p is the pressure, ν is the coefficient of viscosity, and \mathbf{f} is the aggregate of the external forces including gravity.

3.1 MPS Model

We introduce a weight function w . This weight function is used in the interparticle interaction model.

$$w(r) = \begin{cases} \frac{r_e}{r} - 1, & (0 \leq r < r_e) \\ 0, & (r_e \leq r) \end{cases} \quad (2)$$

In this function, r is the interparticle distance and r_e is the influence radius. As r_e shrinks, the number of interacting particle reduces. Accordingly, computational time decreases. However, computation becomes unstable if r_e is too small.

A fluid is represented by a set of particles $i \in [1..N]$ with position \mathbf{r}_i , and additional attribute ϕ_i . The particle number density at coordinate \mathbf{r}_i where particle i is located is defined as follows:

$$n_i = \sum_{j \neq i} w(|\mathbf{r}_j - \mathbf{r}_i|). \quad (3)$$

In this equation, the contribution from particle i itself is not considered.

MPS approximates the gradient and Laplacian of ϕ_i at a location of particle i by the following model:

$$\langle \nabla \phi \rangle_i = \frac{d}{n^0} \sum_{j \neq i} \left[\frac{\phi_j - \phi_i}{|\mathbf{r}_j - \mathbf{r}_i|^2} (\mathbf{r}_j - \mathbf{r}_i) w(|\mathbf{r}_j - \mathbf{r}_i|) \right] \quad (4)$$

$$\langle \nabla^2 \phi \rangle_i = \frac{2d}{\lambda n^0} \sum_{j \neq i} [(\phi_j - \phi_i) w(|\mathbf{r}_j - \mathbf{r}_i|)] \quad (5)$$

Where d is the number of dimensions, n^0 is primary particle number density, w is the weighting function, and λ is denoted as follows:

$$\lambda = \frac{\int_V w(r) r^2 dv}{\int_V w(r) dv} \quad (6)$$

3.2 Calculation

Adding Eqs. (4) and (5) to the governing equation (1) and symmetrisation yield the following particle body forces

$$\begin{aligned} \mathbf{f}_i^{pressure} &= -\nabla p \\ &= -\frac{d}{n^0} \sum_{j \neq i} \left[\frac{p_j - p_i}{|\mathbf{r}_j - \mathbf{r}_i|^2} (\mathbf{r}_j - \mathbf{r}_i) w(|\mathbf{r}_j - \mathbf{r}_i|) \right], \end{aligned} \quad (7)$$

$$\begin{aligned} \mathbf{f}_i^{viscosity} &= \nu \nabla^2 \mathbf{u} \\ &= \nu \frac{2d}{\lambda n^0} \sum_{j \neq i} [(\mathbf{u}_j - \mathbf{u}_i) w(|\mathbf{r}_j - \mathbf{r}_i|)]. \end{aligned} \quad (8)$$

The pressures p_i are computed via the ideal gas equation [Müller et al. 2003; Ihmsen et al. 2011] using particle number density

$$p_i = c_s^2 (n_i - n_0). \quad (9)$$

Where c_s denotes the speed of sound.

4. BUBBLE

We adopted the method of Cleary et al. [2007] and improved it by using by the weight function of the MPS method. Furthermore, the nucleation site, which is where bubbles generate, is located on the surface of the wall, and the size of bubbles is decided automatically by the rate of impact with which the fluid particle collides with the wall particle.

4.1 Buoyancy

Bubbles are positively buoyant because of the large density difference between the bubble and the water. Therefore bubbles have the buoyancy force that propels the bubble in the direction of the liquid surface. To make larger bubbles rise faster than smaller ones, the buoyancy forces should account for the volume of the air bubbles V_{bub} . This leads to

$$\mathbf{f}_i^{buoyancy} = k_{\text{buo}} V_{i,\text{bub}} (\rho_{\text{bub}} - \rho_{\text{liq}}) \mathbf{g} \quad (10)$$

Where k_{buo} controls the buoyancy, V_{bub} is the bubble volume, ρ_{bub} is the bubble density, ρ_{liq} is the density of background liquid, and \mathbf{g} is gravity.

4.2 Collision

We use the discrete element method (DEM) [Cundall and Strack 1979] to model the collision interaction. In this simulation, this force acts between the discrete bubbles.

$$\mathbf{f}_i^{collision} = \sum_{j \neq i, \Delta x > 0} (-k_{col} \Delta x + C v_n) \frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|} \quad (11)$$

Where Δx is the overlap between bubbles, v_n is the normal component of the relative velocity, k_{col} is the spring constant, and C is a suitable damping constant.

In this simulation, the bubbles have various sizes. Thus, a penalty method is adopted as the interaction between bubbles and the wall because bubbles cannot pass through the wall.

4.3 Fluid Interaction

In the works of Cleary et al. [2007] and Hong et al. [2008], the drag force was one of the major coupling forces that made the bubble particle follow the back ground flows. Thus, we couple the bubbles to the liquid phase via an empirical drag force. With respect to the generally high Reynolds number for bubbles, the drag force \mathbf{f}_{drag} of fluid particle j acting on bubble particle i is computed as

$$\mathbf{f}_i^{drag} = -k_{drag} A_{i,bub} (\mathbf{u}_i - \mathbf{u}_j) |\mathbf{u}_i - \mathbf{u}_j| \quad (12)$$

Where k_{drag} is a constant drag coefficient and $A_{i,bub}$ is the surface area of the bubble. In this simulation, since bubble particles are approximated as discrete entities with fixed shapes, $A_{i,bub}$ is easy to compute. Additionally, the effect of bubble momentum on liquid phase is neglected since the bubble inertia is typically small.

Ihmsen et al. [2011] suggested, we propose a drag force that is motivated by the weight function Eq. (12) taking into consideration the flow direction and the distance in Eq. (2). The drag force is defined as

$$\mathbf{f}_i^{drag} = -k_{drag} \sum_{j \in liq} A_{i,bub} (\mathbf{u}_i - \mathbf{u}_j) |\mathbf{u}_i - \mathbf{u}_j| w(|\mathbf{r}_i - \mathbf{r}_j|). \quad (13)$$

4.4 Cohesive Force

We define a cohesive force $\mathbf{f}^{cohesive}$ between dry foam bubbles in order to produce realistic rheological behaviour for the foam in which the bubbles naturally aggregate.

$$\mathbf{f}_i^{collision} = \sum_{j \neq i, \Delta x < 0} (-k_{col} \Delta x + C v_n) \frac{\mathbf{r}_j - \mathbf{r}_i}{|\mathbf{r}_j - \mathbf{r}_i|}. \quad (14)$$

The form of cohesive force is the same as the collision force, expect now the overlap is allowed to be negative.

4.5 Diffusion

The source of bubbles is dissolved gas. Therefore, we need to track the gas concentration G that the discrete fluid particles have. This turns into the discrete bubbles.

The evolution of the gas concentration G due to diffusion is given by a conventional diffusion equation:

$$\frac{\partial G_i}{\partial t} = D \nabla^2 G \quad (15)$$

Where D is the diffusivity of gas in fluid. If the MPS Laplacian model (5) is used, the evolution of the discrete gas concentration G_i sampled at the particles can be computed as

$$\frac{\partial G_i}{\partial t} = D \frac{2d}{\lambda n^0} \sum_{j \neq i} \left[\frac{G_j - G_i}{|\mathbf{r}_j - \mathbf{r}_i|^2} (\mathbf{r}_j - \mathbf{r}_i) w(|\mathbf{r}_j - \mathbf{r}_i|) \right]. \quad (16)$$

Furthermore, we introduce the gas loss constants due to nucleation and due to expansion of nearby discrete bubbles, which are represented by N_a and E_a , respectively. In this simulation, N_a is handled automatically by impact between discrete fluid particles and virtually-arranged wall particles. This makes

possible a variety of bubble sizes. These gas loss constants are used, and the change in the discrete gas concentration from G_i^n (n is time step.) to G_i^{n+1} is represented as

$$G_i^{n+1} = G_i^n - N_a - \sum_{j \in \text{bubble}} E_a w(|\mathbf{r}_j - \mathbf{r}_i|). \quad (17)$$

Where N_a is subtracted if a bubble is caused.

5. RESULTS

Figure 2 shows water being poured and bubbles being generated. The glass is empty to begin with before a stream of sodas is poured into it. On the boundary, there are around 18,000 wall particles, which interact with dry foam bubbles using cohesive force. Also, grids are constructed for the penalty method. After 6 seconds of simulation time, around 8,000 4-mm fluid particles and around 36,000 gas bubbles have also been created. In Figure 1, blue particles represent liquid, white particles represent bubbles, and red particles represent the wall. (Wall particles in the range in which z is smaller than 0 are hidden.) The surface is constructed by the marching cubes method.

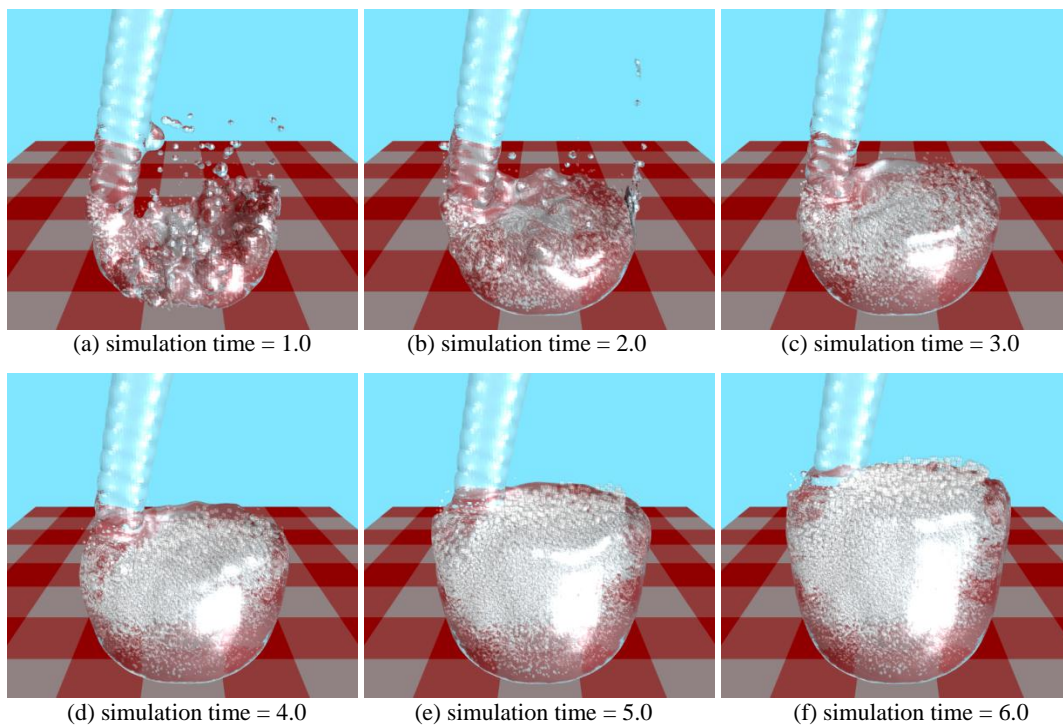


Figure 2. Results in time evolution

Table 1. Parameter

Parameter	value in this paper
Influence radius : r_e	2.1
Cohesion spring stiffness : k_{buo}	10.0
Collision spring stiffness : k_{col}	100.0
Damping coefficient : C	0.25
Drag coefficient : k_{drag}	1.0
Initial of gas concentrate: G_i	0.15
Gas loss constant due to expansion: E_a	0.001

6. CONCLUSION

We have presented a combined MPS and discrete bubble approach for modelling flow of bubbles in liquids. Using this model, we can simulate natural foam generation automatically. Also, this model is best suited to parallelization since it has explicit expression and a simple model of interactions between bubbles and fluids. Therefore this model is applicable to real-time animation.

ACKNOWLEDGEMENT

This investigation was partially supported by JST CREST.

REFERENCES

- Paul W. Cleary, Soon Hyung Pro, Mahesh Prakash, and Bon Ki Koo. Bubbling and frothing liquids. *ACM Transactions on Graphics*, 26(3):3, July 2007.
- P. A. Cundall and O. D. L. Strack. A discrete numerical model for granular assemblies. *Geotechnique*, 29(1):47–65, 1979.
- S. T. Greenwood and D. H. House. Better with bubbles: Enhancing the visual realism of simulated fluid. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2004.
- M. Sakai A. Haga T. Onoe H. Yamashita H. Ito, S. Koshizuka and K. Nakagawa. Lung deformation simulation using mps method for high accuracy radiotherapy. *WCCM/APCOM 2010 Proceedings*, (183), July 2010.
- Christian Vogelgsang Hendrik Kuck and Gu nther Greiner. Simulation and rendering of liquid foams. *GI 2002 Online Papers*, 2002.
- Jeong-Mo Hong and Chang-Hun Kim. Animation of bubbles in liquid. *EUROGRAPHICS*, 22(3), 2003.
- Jong-Chul Yoon Jeong-Mo Hong, Ho-Young Lee and Chang-Hun Kim. Bubbles alive. *the ACM SIGGRAPH conference proceedings*, 2008.
- Seiichi KOSHIZUKA Kazuya SHIBATA and Katsuji TANIZAWA. Three-dimensional numerical analysis of shipping water onto a moving ship using a particle method. *Journal of Marine science and technology*, 2:214–227, 2009.
- S. Koshizuka and Y. Oka. Moving-particle semi-implicit method for fragmentation of incompressible fluid. *Nuclear Science Engineering*, 123:421–434, 1996.
- Tamako H. Koshizuka, S. and Y. Oka. A particle method for incompressible viscous flow with fluid fragmentation. *Journal Computational and Engineering*, 62:29–46, 1995.
- Gizem Akinci Markus Ihmsen, Julian Bader and Matthias Teschner. Animation of air bubbles with sph. GRAPP, 2011.
- Takahiro Harada Masahiro Kondo, Masayuki Tanaka and Seiichi Koshizuka. Elastic objects for computer graphic field using mps method. *SIGGRAPH '07 ACM SIGGRAPH 2007 posters*, 2007.
- Mikio Sakai Masayuki Tanaka and Seiichi Koshizuka. Rigid body simulation using a particle method. *SIGGRAPH'06 ACM SIGGRAPH 2006 Research posters*, 2006.
- David Charypar Matthias M'uller and Markus Gross. Particle-based fluid simulation for interactive applications. *Eurographics/SIGGRAPH Symposium on Computer Animation*, 2003.
- Richard Keiser Markus Gross Matthias M'uller, Barbara Solenthaler. Particle-based fluid-fluid interaction. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2005.
- D. Metaxas M. Sussman V. Mihalef, B. Unlusu and M. Y. Hussaini. Physics-based boiling simulation. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2006.
- Jun-Hai Yong Wen Zheng and Jean-Claude Paul. Simulation of bubbles. *Eurographics/ACM SIGGRAPH Symposium on Computer Animation*, 2006.

MEDICAL IMAGE 3D VISUALIZATION BY VECTOR BASED METHODS

Katrina Bolochko, Mihails Kovalovs and Aleksandrs Glazs
Riga Technical University
Meza street 1/3, Riga, Latvia, LV-1084

ABSTRACT

This paper describes approaches and methods that are aimed to solve various medical image processing tasks, including extraction of region of interest, the extracted regions contour processing and the following 3D visualization of the extracted region. Two different approaches for control point acquisition and 3D model reconstruction using the points acquired from the region of interest are proposed. The proposed methods of 3D visualization use a vector-based surface model creation rather than raster-based systems to significantly improve computational efficiency and the quality of generated 3D models. The proposed methods were tested on the medical images of a brain acquired by computer tomography and proven to be applicable to different types of region of interest. The results can be used to provide practical improvements to the reliability of medical diagnostics.

KEYWORDS

Medical images, image analysis, image processing, visualization, modeling.

1. INTRODUCTION

The 3D visualization is an important aspect of image analysis and research. Analysis and visualization of different medical images, obtained by computer tomography (CT) or magnetic resonance imaging (MRI), is important to medical research and clinical practice. In this case, some the most important initial phases of medical image processing are tasks of extraction and analysis of different image regions (a medical object or a pathology zone) [7]. In relation to this, a necessity to solve the task of 3D visualization of the region of interest emerges in computer diagnostics. The ability to visualize the orientation, position, size of structures in medical images can be vital to researchers and physicians. In order to obtain a 3D model of the region, the medical image must first be segmented, following the region extraction from the medical image. Afterwards, the extracted region contour control points, which are further used for 3D visualization of the object must be selected.

There are many manual, semi-automatic and automatic image segmentation methods for different image types [2, 3]. However, the properties of medical images require a specific approach for segmentation, based not on the visual intensity or color of image pixels, but rather on the information about the density of every image element. Manual segmentation is a slow process, subject to operator error and bias. Automatic segmentation methods require little user interaction; however, in practice automatic methods sometimes fail and require manual correction (adjustment of the boundary that identifies the region). In this work, a semi-automatic segmentation algorithm described in [5] was implemented for region of interest extraction.

Although there are methods that allow segmentation and visualization of the medical image [8], the existing approaches of 3D visualization are not always able to provide a high-quality smooth surface of the 3D model. The resulting 3D models have a distinct aliasing (staircase) effect (Fig 1). The well-known marching cubes algorithm also has flaws, like aliasing effect and other negative aspects described in [6]

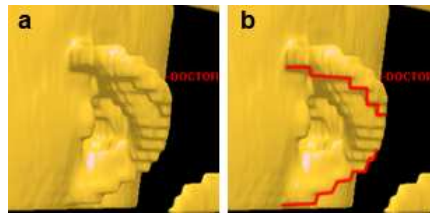


Figure 1. Aliasing effect. a) original 3D model, b) the red line shows the aliasing effect.

In this paper, two approaches of contour analysis and selection of control points are proposed, as well as three methods of 3D visualization. The first approach is based on the even control point selection, while the second approach is based on an uneven point selection. Two control point selection approaches were used in order to test whether the even or uneven control point selection has any effect on the resulting surface quality. The proposed methods of 3D visualization use a vector-based surface model creation rather than raster-based systems to significantly improve computational efficiency and the quality of generated 3D models. The proposed methods result in a better quality of surfaces of the 3D model.

2. MATERIALS AND METHODS

2.1. Contour Analysis and Visualization based on Even Point Selection

The proposed method selects control points on the contour evenly, which ensures a more precise 3D model reconstruction.

The initial image is limited by two border lines that are parallel to y axis and an initial point is selected, which is positioned on the first intersection of the contour and a line, parallel to the border lines and crossing the middle of line segment that connects the border lines. The control point selection algorithm can be described as follows:

1. A radius R is defined and from the last found control point a circle line is drawn. Afterwards, intersection points between the circle line and the contour are found.
2. If there are more than two intersections between the circle line and the contour (Fig 2.2), then, radius R is gradually decreased until there are only two intersection points (Fig. 2. 3).
3. When there are two intersections between the circle line and the contour (Fig 2.1), from these two points, the one that is the furthest from the previously found point is selected. In the beginning, when the previous point is concurrent to the initial point, any of these two points can be selected as the next point.
4. A new circle line is drawn from the last found control point (Fig. 2.4). If the radius was decreased previously and it does not equal R , then if there are only two intersection points, the radius increases until it reaches R or until it reaches a value just before the point, where there are more than two intersections. Steps 1-4 repeat, until the initial point is reached again.

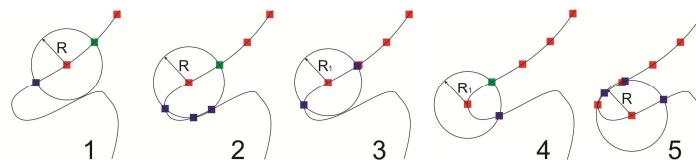


Figure 2. Even selection of control points

The algorithm described in pseudo code is shown on Figure 3.

```

VAR  $P_i$ : current control point in search,  $i = 0, 1, 2, \dots$ ;  $R$ : defined radius of search circle;  $C$ : center coordinates of search circle;
 $N$ : number of intersections between circle line and contour;

REPEAT
IF  $i > 0$  THEN  $C = P_{i-1}$  ELSE  $C = P_0$ ;
draw circle with defined radius  $R$ ;
IF  $N > 2$  THEN REPEAT decrease radius and redraw circle UNTIL  $N = 2$ ;
IF  $N = 2$  THEN
BEGIN
IF  $i > 0$  THEN  $P_i$  = intersection point that is farthest from  $P_{i-1}$ ;
IF  $i = 0$  THEN  $P_i$  = any intersection point;
END
 $C = P_i$ ;
increase  $i$ 
UNTIL  $P_i = P_0$ 

```

Figure 3. Pseudo code for even control point selection

When the algorithm finds the control points on all tomograms, excess points, which are obtained when the radius was decreased, are deleted. In order to delete the excess points, the following algorithm is used: firstly, the first point on the tomogram is taken, and the distance between this point and its neighboring points are checked. The first point, for which this distance will be equal or greater than the given radius R will be saved, all the others with lesser distance are deleted. Then the algorithm moves to the last saved point and repeats the process, and thus the algorithm checks all the points on the tomogram and saves only points with the distance between them not less than radius R .

After the acquiring of control points a 3D model can be constructed. This task is solved by triangulating the acquired point dataset, forming a mesh for the 3D model. The triangle vertices are formed from the three adjacent control points. The first two adjacent points are iteratively taken from the contour on the first tomogram and the third one is the next slice contour control point, which is the closest to the first vertex of the triangle (Fig. 4.1 and Fig. 4.3). When searching for the third control point, only points closest to the last selected third vertex are taken into consideration, in order to avoid shifts in the mesh. Initially, only 10 points are taken into consideration, but the number may increase, if the distance between these points and the last selected one is lesser than the minimal threshold, which is equal to five radiuses R . Furthermore, if there is a hole between the constructed triangle and a previous triangle, it is closed up with a triangle, where the first vertex is the lower control point of the hole and the other two are iteratively taken from the top slice (Fig. 4.2 and Fig. 4.4).

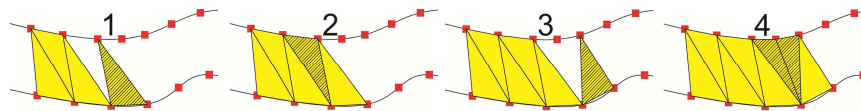


Figure 4. Triangulation of the acquired control points

2.2. Contour Analysis and Visualization based on Uneven Point Selection

For the uneven selection of the control points a method described in [4] is used. The proposed method consists of two stages, illustrated by Fig. 5: the selection of initial control points for the adaptive contour using the clockwise algorithm (Fig. 5.1) and adjustment of the adaptive contour to match the objects contour (Fig. 5.2-5.5).

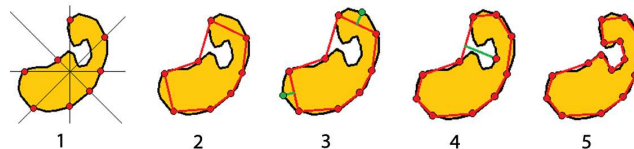


Figure 5. Uneven selection of control points

The advantage of this method is that it provides a minimum number of points, required to precisely interpolate the contour. However, due to the uneven form of the control points, there may be shifts in the 3D model during the reconstruction. To solve this problem a special parameter $t_{i,j}$ is used to mark every control

point. In this case, each j^{th} point $p_{i,j}$ of the contour on slice i is assigned a special parameter $t_{i,j}$ that is further used to create a 3D mesh from the control points. The parameter $t_{i,j}$ is calculated according to the following equation:

$$t_{i,j} = \frac{l_{i,j}}{P_i}, \tag{1}$$

where P_i – is the perimeter of the contour on slice i ;

$l_{i,j}$ – is the contour length from the point $p_{i,0}$ to point $p_{i,j}$.

As the result of this algorithm each point on the slice has its own parameter $t_{i,j}$, which is taken in consideration when triangulating points. The triangulation algorithm processes the slices in pairs. In each pair of slices, the slice with the lesser amount of points is considered to be the first slice and the algorithm processes slices in two stages:

1. The first slice is considered as the base for triangulation, meaning that two points on this slice are taken as base vertexes for the triangle and the third vertex is taken on the second slice (Fig 6.1). The base vertexes of the triangle are formed by neighboring contour points on the first slice. Then for each pair a third vertex is found on second slice. Only points where the difference between its parameter and the base vertices parameter does not exceed some given threshold Δt are taken in consideration. Among these points, the one with the minimal distance between it and the previous two vertices is chosen to be the third vertex.

2. The remaining connections between the two slices are found. The base vertexes of the triangle are formed by neighboring contour points on second slice. For each pair a third vertex is found on first slice. For the current pair of points on bottom slice, two adjacent existing triangles A and B are found. Then, the vertex p_{AB} that is shared by both triangles is added as the third vertex (Fig. 6.2).

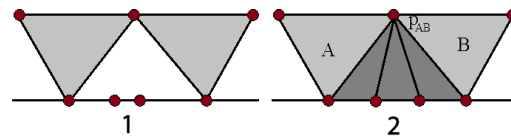


Figure 6. Triangulation of the acquired control points

3. EXPERIMENTS

The proposed approaches were tested on the medical images of a brain, acquired with computer tomography. The results of the proposed methods were compared with a 3D imaging software 3D-Doctor [1]. Fig. 5 illustrates the comparison of various methods.

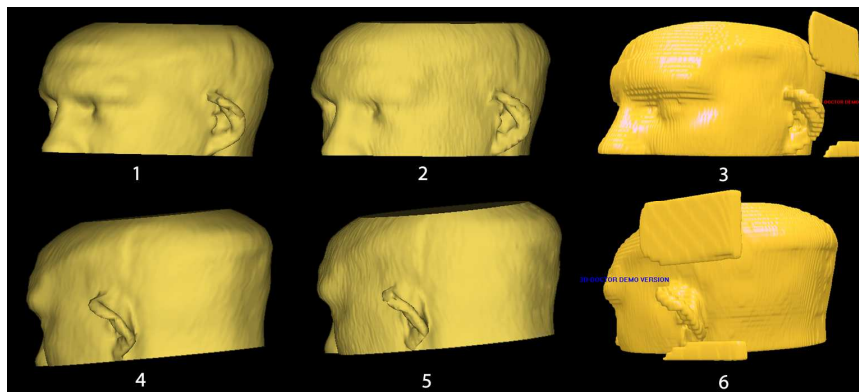


Figure 7. Comparison of different methods. 1,4) Visualization method based on even point selection, 2,5) Visualization method based on uneven point selection, 3,6) 3D-Doctor visualization.

As seen from the results of comparison, the proposed methods (Fig. 7.1, Fig. 7.4) and (Fig. 7.2, Fig. 7.5) provide a better result on image quality. The 3D-Doctor approach (Fig. 7.3, Fig. 7.6) has an aliasing effect on the surface.

The experimental results also show that the control point selection method (even or uneven) has no effect on 3D model quality. Both approaches gave similar result without the aliasing effect on the surface of the object. An additional advantage of the uneven control point selection is that it provides a minimal amount of control points necessary to interpolate the surface of the object, resulting in quicker surface generation as less calculations are necessary for surface reconstruction.

4. CONCLUSION

In this paper, methods of control point selection on the contour of region of interest and methods of 3D visualization are proposed. The methods have been tested on medical images of the brain acquired by computer tomography.

The methods have been compared with existing solutions and the experiments show that the proposed methods give a better visual result than existing method of 3D-Doctor. The surface of the model is smoother, without an aliasing effect. The experimental results also show that the control point selection method (even or uneven) has no effect on 3D model quality. Overall, vector-based methods show stable results in visualization and may be implemented in medical software to provide better 3D visual quality of the reconstructed medical object.

ACKNOWLEDGEMENTS

This work has been supported by the European Social Fund within the project «Support for the implementation of doctoral studies at Riga Technical University».

REFERENCES

1. Able Software Corp. 3D-DOCTOR, FDA 510K cleared, vector-based 3D imaging, modeling and measurement software, <http://www.ablesw.com/3d-doctor/>
2. Dastidar P., Heinonen T., Vahvelainen T., Elovaara I., Eskola H. Computerised volumetric analysis of lesions in multiple sclerosis using new semi-automatic segmentation software. *Med. Biol. Eng. Comput.*, vol. 37, 1999, pp. 104-107
3. Jiang C., Jiang L., Zhang X., Gevantmakher M., Meinel C. A New Practical Tool for 3D Medical Image Segmentation, Visualization and Measurement. *Proceedings of CCCT, 2004, Austin, USA*
4. Krechetova K., Glaz A. Contour Extraction and Processing in CT Images. *Biomed. Eng. conference proc., Kaunas, Lithuania, 2009*
5. Krechetova K., Glaz A. Development of a new segmentation method for medical images. *Biomed. Eng. conference proc., Kaunas, Lithuania, 2007, pp. 133-136*
6. Newman T., Yi H. A Survey of the Marching Cubes Algorithm, *Computers & Graphics*, vol. 30, 2006, pp. 854-879
7. Preim B., Bartz D. *Visualization in Medicine, The Morgan Kaufmann Series in Computer Graphics*, 2007.
8. Wu Y. From CT Image to 3D Model, *Advanced Imaging*, Aug. 2001, pp. 20-23

NODE COLOR ASSIGNMENT FOR DECISION TREE STRUCTURE

Jeong Joong Lee¹, Seung Gyu Kim¹, Beom Seok Lee¹, Naveed Ejaz¹, Joon Yeon Choeh¹,
Cheol Whan Lee², Myung Ho Lee³ and Sung Wook Baik¹

¹*College of Electronics and Information Engineering, Sejong University, Seoul, Korea,*

²*Oracle Corporation, Santa Clara, USA*

³*Dept of Computer Science and Engineering, Myong Ji University, Kyung Gi Do, Korea*

ABSTRACT

This paper presents a novel method to assign color information to nodes of a decision tree using YUV color model. Decision trees for data classification are easy to understand and interpret as compared to the results of other data mining methods. However, new representations of decision trees are required as their forms become more complicated when multivariate and massive data sets are processed. The main contribution of this paper is to develop a new YUV based color coordinate system and a node color assignment algorithm for new visualization approaches that can substitute decision tree styled presentation.

KEYWORDS

Data mining, Decision tree, Visualization, color assignment, classification model

1. INTRODUCTION

The understanding of the huge amount of data has been beyond human ability since information began to explode two decades ago. To deal with this massive amount of data, the analytical data mining tools have been popular among domain experts and decision makers. The results of these analysis tools have provided efficient ways to understand the implication of the raw data sets. However, since the analytical results have become too complicated and difficult to interpret into their own domain knowledge, some researchers recently began to introduce a variety of visual data mining methods, which are integrated with traditional data mining techniques. There are three different approaches for such an integrated form [Liu, 2007]. The first approach is to visualize raw data so that users can easily discover the characteristics of the given data set. In this approach, users can manipulate the visualized facilities to view subsets of partial data that they want to deal with. The second one is to visualize models/patterns obtained from data mining to provide better understanding of the derived results. It is considered the most critical research issue, because these may still look complicated to humans when raw data is huge, though such results are reduced to be abstracts of raw data. The third one [Liu, 2007] is to visualize data analyzing processes during which their results are generated. It involves users in the process of data mining, and enhances the effectiveness and efficiency of data analysis by giving them much better understanding of the model construction process.

Our work deals with the analytical model visualization [Kopanakis, 2003, Barlow, 2001, Haworth, 2010] which is the second approach abovementioned. The representation of analytical models depends upon data mining methods. Among the models of data mining methods, a decision tree styled model is one of the most intuitive representations to help human cognition. Also, a decision tree induced from the given data set can be converted to other decision forms, such as decision rule type. Such conversion is worth enough to spend extra processes for novel ways of model visualization even though the decision tree has visual patterns in itself. The reason is that the main purpose of model visualization is not to view the overview of models but to graphically represent partial implication of models for efficient decision making.

To enhance users' readability and understanding, the decision tree styled presentation is usually converted into a text report type describing decision rules. The text form of decision rules may be more appropriate for a decision making step if we assume that the relationship of classification nodes representing decision

conditions is provided. Furthermore, such text form is easier to represent graphically with new visualization approaches by breaking the traditional decision tree typed visualization. Therefore, if the structure information about the relationship of rule condition rule nodes is provided to the text form of decision rules, it is possible to represent new visualization forms which can be restored to the original form of decision tree. A successful visualization technique allows the user to have an insight into the data. The Visualization method using assignment of color must generate a visualization that is understandable by the user of a particular domain without having an in-depth knowledge of data mining. [Larrea, 2010].

This paper presents a concise method to provide the binary tree structure information to the decision rule form by including different color information for data classification to each rule condition corresponding to each node of a decision tree.

2. COLOR BASED VISUALIZATION FOR REPRESENTATION OF DECISION TREE STRUCTURE

It is straightforward to convert from the original form of decision tree to the text form of decision rules. When the conversion happens, there are some benefits in representation of decision rule form as follows:

- 1) It is possible to sort decision rule conditions in order of importance. The most important rule condition can be given high priority in a decision making step.
- 2) It is easy to represent rule conditions together or separately as needed in different graphical ways.
- 3) It is more appropriate to view or print out specific decision rules.

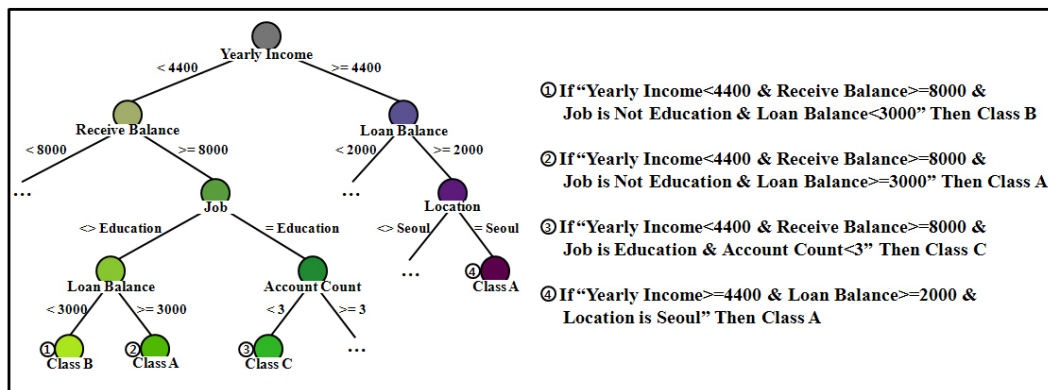


Figure 1. Decision Tree for individual credit information and the decision rules converted from it

Figure 1 presents an example of a decision tree and its corresponding decision rules form. For the connection of two forms, the assignment of node colors can be achieved on a new YUV based color coordinate system which is developed from a YUV color model [Mukherjee, 2005]. YUV color model shows better performance for assignment of colors as compared to the other color models as each of the three color components in YUV color model are independent of each other [Li, 2010]. There is relation information between nodes in the decision tree itself. To retain the structure information during the conversion of a decision tree form to a decision rules text form, the color/brightness information indicating its structure is needed in the text form. The color/brightness assignment for nodes of the given decision tree in Figure 1 has been determined after the color assignment on the coordinate plane shown in Figure 1.

Each node of the decision tree is associated with each color determined by an algorithm using the 2 dimensional UV based color model projected by variable Y. From the intersection of U and V axis on a coordinate plane up to the largest inner circle (with the radius of 0.5) within the square boundary which is the maximum range (from -0.5 to +0.5 for both U and V axes) of the coordinate plane, a circle typed color band expands along increasing radius related to U and Y values with the given incremental interval of Y value. The new YUV based color coordinate plane is presented in Figure 2, and the algorithm is performed on it as follows:

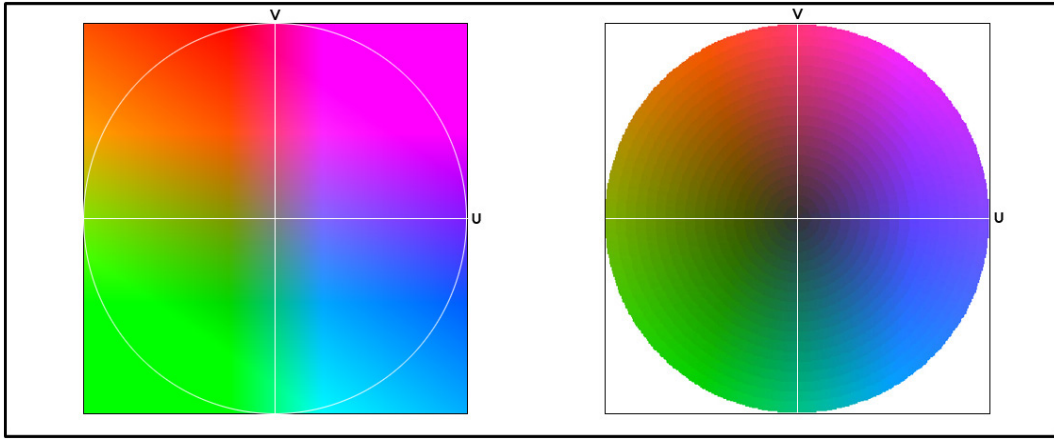


Figure 2. UV based color coordinate plane with $Y=0.5$ and the variable Y along sliced color bands

Step 1: Determine the depth of the given decision tree, which equals to the number of layers in the YUV based color coordinate plane shown in Figure 2. As the radius level (a real radius value of the plane is used to find the position of each node) of the circle typed color band increases according to the depth of decision tree, the brightness value of Y decreases by a certain interval which is a result value of $(\max Y - \min Y)/\text{depth}$.

Step 2: Each node is supposed to have both maximum and minimum angles with which we can find a specific position along the circle color band. The root node (level 0) has a color on the center of the coordinate plane and it has initial maximum and minimum angles as follows:

$$L(\text{Initial minimum angle}) = 90^\circ, H(\text{Initial maximum angle}) = 450^\circ \quad (1)$$

Step 3: To allow children nodes on the first level to have their own color, increase the radius of the color band by 1 and find their maximum and minimum angles from those of their parent as follows:

For left child node on the level 1,

$$\begin{cases} L(\text{left}) = L \\ H(\text{left}) = \frac{L+H}{2} \end{cases} \quad (2)$$

For right child node on the level 1,

$$\begin{cases} L(\text{right}) = \frac{L+H}{2} \\ H(\text{right}) = H \end{cases} \quad (3)$$

Step 4: From the next level, find their own position angle corresponding to their color position on the coordinate plane and increase the radius of the color band level.

For left child node,

$$M(\text{left}) = \frac{L(\text{left}) + H(\text{left})}{2} \quad (4)$$

For right child node,

$$M(\text{right}) = \frac{L(\text{right}) + H(\text{right})}{2} \quad (5)$$

Step 5: Find the coordinate values on the UV plane.

$$\begin{cases} U = r \times \cos\left(\frac{\text{low} + \text{high}}{2}\right) \\ V = r \times \sin\left(\frac{\text{low} + \text{high}}{2}\right) \end{cases} \quad (6)$$

Step 6: Determine a color of each node with the value of U and V on the coordinate system (CD). If we denote the color of each node by Node (c) then,

$$\text{Node}(c) = CD(U, V) \quad (7)$$

Step 7: Go to step 3 until a leaf is reached.

Figure 2 includes a UV based color coordinate plane on the left side with the fixed value (0.5) of Y in comparison with the new color model on the right side. The UV based color coordinate plane with 25 color bands with different values of brightness (Y). The value of Y gradually increases from 0.2 to 0.5. Naturally, the brightness of the color band becomes high from the center of the plane to the largest inner circle band within the square boundary.

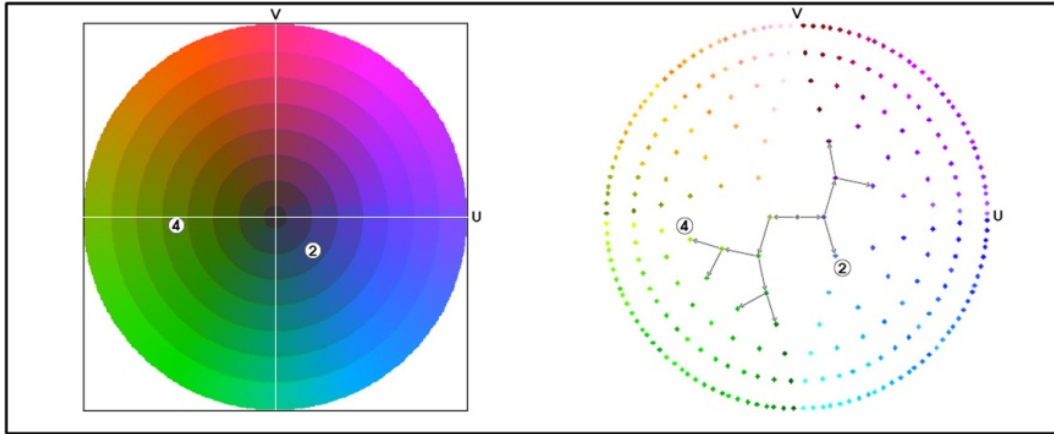


Figure 3. Color coordinate plane for the given decision tree, and color assignment on the coordinate plane

Figure 3 presents the new UV based color coordinate plane with brightness band level equal to the depth of the given decision tree and the color assignment for decision tree nodes on the coordinate plane. Each node of the decision tree can be associated with a specific color determined in the plane by our algorithm. The color/brightness assignment shown in Figure 3 is a result of applying algorithm to decision tree of Figure 1.

3. CONCLUSIONS AND FUTURE WORKS

Our technique contributes to the development of a UV based color coordinate system with sliced circle stripes of different brightness, and a color/brightness assignment algorithm for nodes of a decision tree in the system. This method can be applied to new visualization approaches that can substitute decision tree styled presentation. In future, we intend to perform usability analysis of our visualization scheme to determine its efficacy as compared to the other known schemes. We also intend to introduce a new paradigm of a representation for a predictive model which is a form of a classification tree created through decision tree learning.

ACKNOWLEDGEMENT

Corresponding author : Sung Wook Baik, Professor of College of Electronics and Information Engineering. This work is supported by the Seoul R & BD Program (JP090972M0214831).

REFERENCES

- Barlow, T. and Neville, P., 2001. Case study: visualization for decision tree analysis in data mining. *IEEE Symposium on Information Visualization 2001*. San Diego, USA, pp. 149-152.
- Kopanakis, L. and Theodoulidis, B., 2003. Visual data mining modeling techniques for the visualization of mining outcomes. *Journal of Visual Languages & Computing*, Vol. 14, NO. 6, pp. 543-589.
- Liu, Y and Salvendy, G., 2007 *Design and evaluation of visualization support to facilitate decision trees classification*, *International Journal of Human-Computer Studies* Vol.2, No.65, pp95-110

- Mukherjee, J. et al, 2005. Demosaicing of images obtained from single chip imaging sensors in YUV color space. *Pattern Recognition Letters*, Vol. 26, No. 7, pp. 985-997
- Larrea, M.L. et al, 2010. Semantics-based Color Assignment in Visualization. *Journal of Computer Science & Technology*, Vol. 10, No. 1 pp. 14-18.
- Haworth, R. et al, 2010. Visualizing Decision Trees in Games to Support Children's Analytic Reasoning: Any Negative Effects on Gameplay?. *International Journal of Computer Games Technology*, Vol. 2010, pp. 1-11.
- Li, M. et al, 2010. Support vector regression based color image restoration in YUV color space. *Journal of Shanghai Jiaotong University(Science)*, Vol. 15, No. 1, pp. 31-35

HUMAN ACTION ANALYSIS VIA VIDEO STREAMS AND WORKFLOW SCHEDULING

Anastastios Doulamis¹, Nikolaos Matsatsinis¹ and Pavlos Delias²

¹*Department of Production Engineering and Management,
Technical University of Crete, Chania, Greece*

²*Faculty of Economics, Technological Institute of Kavala
Kavala, Greece*

ABSTRACT

This paper proposes a new production scheduling algorithm that exploits (a) visual observations of industrial operations to estimate the actual completion times for tasks and (b) incremental graph partitioning-based clustering algorithms. The latter are implemented through an incremental implementation of the spectral clustering. Computer vision tools are applied to identify industrial operations via visual observations. In this case, an arduous industrial environment is considered and thus novel self corrected mechanisms are used.

KEYWORDS

Visually observed industrial operations; tracking

1. INTRODUCTION

Production scheduling/planning is probably one of the most critical managerial tasks within an industry *with many benefits for the production*. In particular, it (i) can determine whether delivery promises can be met and identify time periods for preventive maintenance, (ii) gives personnel an explicit statement of what should be done so that supervisors and managers can measure their performance, (iii) minimizes the average flow time through the system, (iv) minimizes setup times, (v) maximizes machine and/or worker utilization, (vi) can identify resource conflicts, control the release of jobs and ensure that the materials are ordered in time and (vii) can better coordinate the resources to increase productivity and minimizing operating costs.

One possible way to monitor current processes/operations is through the use of Radio Frequency Identification -RFID sensors (Fernandez et al., 2010)(Ilic et al., 2010). These, however, are efficient only for specific industrial environments, while their reliability under harsh industrial conditions is questionable. Furthermore, they prevent workers from independent acting and working (Palma et al., 2010). Another possibility is the use of cameras. Nevertheless, cameras that do not support machine learning technologies and computer vision methods actually result in a manual survey of the industrial operations fact that eliminates any possibility for a dynamic (non-rigid) industrial planning. It is impossible for the survey employees to continually concentrate on monitors that display different activities in different areas. Additionally, there is a subjective interpretation as far as humans' behaviours are concerned, let alone additional cost and exploitation of humans' resources.

To face these difficulties, we can exploit in this paper visual sensors (cameras). The recent advances in computer vision and machine learning society have endowed the cameras with smart capabilities. They can detect and salient objects, track moving entities, interpret important events taking place in the industry and finally be adapted their performance to environmental changes. Thus, they can endow modern factories with new cognition capabilities transforming them to "*smart industries*". This, in the sequel, increases scheduling efficiency making it possible to be executed on-line. Such video supervision provides a series of social/economical impact that includes (i) avoidance of fatal/severe injured accidents in the industry due to erroneous guidance of machineries, (ii) inclusions of quality assurance plans in executing the industrial processes, (iii) supervision of the predefined workflows and set alerts in case that some of these tasks are erroneously or deficiently executed since this may lead to hazardous outcomes for the whole process, and (iv)

assistance of the survey employees, if exists in the plant, to pay attention on important events rather than watching billions of frames to localize event of interests.

Most of the current approaches for industrial scheduling exploit concepts derived from computational processors (Proth, 2007)(Shaik et al., 2007)(Drotos et al., 2009). The incorporation of computer vision tools able to understand the actual (real-time) execution of industrial processes is rather limited. A survey of industrial vision systems has been reported in (Malamas et al., 2003), while extraction of salient features for defining and detecting industrial events has been discussed in (Motai, 2005). Assistance towards an efficient visual supervision of an industrial plant can be achieved by exploiting distributed cameras as well as communication issues among them. (Karakaya and Qi, 2009). Surveying industrial tasks and workflows using web services has been presented in (Idoughi et al., 2010). The main objective of this work handle complex and concurrently executed services on large-scale industries. The adoption of a Service Oriented Architecture (SOA) architecture and the use of web services enables a flexible and transparent interaction between the field devices and human operators (Doulamis and Matsatsinis, 2010).

In this paper, we propose a new production scheduling algorithm which is related on defining events, actions and workflows on visual surveyed areas of an industry. The algorithm is based on a new dynamic spectral clustering methodology. This means that instead of using the conventional spectral clustering algorithm (Bach and Jordan, 2004), we adopt a modified version of it so that dynamic arrivals and tasks' workflows and dynamic deviations of the actual to the estimated completion times are allowed (Ning et al., 2010).

2. STATIC INDUSTRIAL PLANNING VIA SPECTRAL CLUSTERING

Assuming that we have calculated the requested start (S_i) and finish times (F_i) for the i th workflow W_i (let N be the total number of operations), we can model their relations as an indicator of their overlapping, say $a_{i,j}$. Instead of (Ning et al., 2010) in which the $a_{i,j}$ is defined as binary measure, in this paper we adopt a continuous modification of the overlapping measure to make it more robust to the dynamic changes. This means that

$$a_{ij} = \begin{cases} \varphi & \text{if } W_i, W_j \text{ non-overlapping in time} \\ \varphi(1 - y_{ij}) & \text{if } W_i, W_j \text{ overlapping in time} \end{cases} \quad (1)$$

As the ultimate goal is to assign every workflow (i.e. all individual operations) to a resource so as to meet as much as possible the requested deadlines, we introduce two objective optimization criteria, which can measure the plan's efficiency with respect to the overall goal. The first one is associated with the violation of the deadlines. In our case, since meeting the deadlines stands for delivering the operations no later than the requested finish times (F_i), avoiding deadlines' violation implies avoiding any tasks' overlapping among the tasks that are executed using the same resource. The corresponding optimization criterion is mathematically formulated as

$$V_m = \frac{\sum_{i \in O_m, j \notin O_m} a_{i,j}}{\sum_{i \in O_m, j \in W} a_{i,j}}$$

Where m indicates that the optimization metric refers to the m^{th} resource, O_m is the set of the operations that is planned for allocation to the m^{th} resource, and W is the set of all the operations that require to be executed. Additionally, V_m is a marginal optimization criterion. The overall criterion is calculated by the summation of these marginal criteria over all the available resources M , and as stated before, an efficient scheduling plan should keep it at a minimum value.

The second optimization criterion concerns the thriftiness of the plan, namely how resources should be utilized so as to prevent wasteful schedules. In our case, maximizing the utilization of resources implies balancing the amount of operations that are allocated to each resource. We can guide the solution towards

this direction by favoring the maximization of the overlapping of the operations that are executed by different resources. This criterion can be mathematically expressed as

$$U_m = \frac{\sum_{i \in O_m, j \in O_m} a_{ij}}{\sum_{i \in O_m, j \in W} a_{ij}}$$

Once again U_m is a marginal optimization criterion (with respect to the m^{th} resource). The overall criterion is calculated similarly with the deadlines' violation criterion, i.e., as the sum of all the U_m over all the M available resources.

As it is shown in a previous work (Delias et al., 2011), it is possible to jointly optimize the above mentioned criteria through the use of the Ky-fan theorem (Fan, 1951). In particular, the matrix solution S , which indicates which operation is assigned to which resource is calculated as $S = D^{1/2} E$, where E is a matrix $N \times M$, the columns of which are the eigenvectors of the M largest *eigenvalues* of matrix $D^{-1/2} A D^{-1/2}$, D denotes the degree matrix of the affinity matrix A , as $D = \text{diag}\{d_1, d_2, \dots, d_n\}$, and $d_i = \sum_j \alpha_{ij}$

Actually, the solution matrix S needs an additional step to provide a schedule, as it contains real values while the assignment method requires for values which are either 1 (in the case that the task is assigned), or 0 (in the case that the task is not assigned to the particular resource). This step is about the discrete approximation of the continuous matrix S . For this step, the popular k-means algorithm is applied to cluster the N operations to M resources, based on the 'feature vectors' of operations, i.e., the rows of the matrix S .

3. SELF CORRECTED TRACKING FOR INDUSTRIAL APPLICATIONS

In this paper, we modify the self initialized tracker of (Doulamis, 2010) so as to be appropriate for industrial applications. More specifically, the architecture of (Doulamis, 2010) is only adequate for scenes that contain moving objects. This is not, however, an industrial case since workers can remain almost still for a relatively long time interval in order to fulfill some of their activities. In addition, the background complexity of an industrial environment imposes significant deviations from a good performance.

To address this difficulty, in this paper we modify both the data selector module of (Doulamis, 2010) and the object labeling adapter. In particular, a non-linear classifier is proposed as efficient background model, which in the sequel is used as a more suitable data selector. The classifier is based on a neural network structure. Neural network-based background modeling has been also proposed in (Culibrk et al., 2007). The weights, however, of the classifier do not remain static but they are modified during the video streaming so that the model can be adapted to the dynamics of the current conditions. This is achieved in our algorithm via the object labeling adapter. In (Culibrk et al., 2007), the observed statistics are exploited to dynamically modified the neural network weights, while in this paper an optimal network retraining strategy is adopted as in (Doulamis et al., 2000).

More specifically, initially the Gaussian Mixtures are used to give approximate estimates of the background content. These mixtures are automatically updated from frame to frame to capture small and/or periodic modifications of the visual content on the background. As we have stated above, the neural network classifier is actually used for separating the foreground region from the background based on a non-linear mapping of the Gaussian mixtures information with respect to the specific background content. However, since the background is modified from frame to frame the weights of the neural network classifier cannot be considered constant. Instead, they should be updated.

The generalized neural network retraining algorithm of (Doulamis et al., 2000) is adopted in this paper for an optimal updating of the network parameters. In particular the algorithm modifies the weights so that

- (i) The current statistics of the background are trusted as much as possible
- (ii) A minimal degradation of the already obtained network knowledge is achieved, i.e., the new weights trust as much as possible the already captured statistics

According to our knowledge, this will be first time that a non-linear and dynamic classifier with optimal retraining strategies will be applied for background content modeling. In addition, the exploitation of the self

initialized architecture of (Doulamis, 2010) for the purpose of an industrial plant constitutes another contribution of the proposed work.

4. ESTIMATING THE ACTUAL WORKFLOWS COMPLETION TIMES

The goal of this section is to estimate the actual completion time for an industrial process based on current visual observations as provided through the tracker. In particular, we can construct the traced trajectory for a worker when he/she executes a task, say t , and compare it with the ideal one, say f , as it has been defined by the industry senior engineer. Then, we can estimate the actual completion time for an industrial workflow, if we can find the correspondence of the actual traced trajectory t to the ideal one f .

In this paper, we modify the technique of (Cui et al., 2009) by applying guided search in order to find the most appropriate correspondence points between the two curves. This is due to the fact that the main challenging issues of our partial to whole match approach is that we do not know the last point of the partial curve but only the first (start) one. This is due to the fact that we do not know the current execution time.

In particular, we initially model both curves using the integral of the curvature measures, i.e., the integral of the norm of the second derivative,

$$C(\mathfrak{s}_1 : \mathfrak{s}_2) = \int_{\mathfrak{s}_1}^{\mathfrak{s}_2} |s''(x, y)| ds$$

Where $s''(x, y)$ is the second derivative of a curve, that is either of t or of f . It can be proven in (Cui et al., 2009) that $C(\mathfrak{s}_1 : \mathfrak{s}_2)$ is invariant under a similarity transform. Thus, this measure can be used to represent the curve complexity. However, it is still problematic to match parts of two curves, since the starting value of the integral of the above equation can be of any number.

To solve this problem, we initially consider that the curvature points, as well as their averaging, are the most characteristics points of the curve. Then, we use a cross correlation criterion as a similarity matching for both curves. In particular, if we have defined the most appropriate last point of curve f , then we could take a part of it starting from the first point to the last one and find its correlation to the traced curved t . Let us denote as $P_0(f), P_{last}(f)$ the first and the last point of curve f and as $P_0(tr), P_{last}(tr)$ the respective first and last point of curve t . Then, the correlation coefficient can be considered as a measure for their matching.

$$Corr(P_{last}(f), P_{last}(tr)) = \frac{[f(P_0 : P_{last}) - \mu(f)] * [t(P_0 : P_{last}) - \mu(t)]}{\sqrt{[f(P_0 : P_{last}) - \mu(f)]^2 * [tr(P_0 : P_{last}) - \mu(t)]^2}}$$

In the above equation, $f(P_0 : P_{last})$ refers to a part of the f curve starting from $P_0(f)$ and ends to $P_{last}(f)$. Similarly, we define $tr(P_0 : P_{last})$ curve. The function $\mu(\cdot)$ returns the average of each curve. It is worth to note that $Corr$ function actually depends on the last points of both curves since the first ones are already available and known.

Then, the optimal last point of f is found through the following minimization

$$\hat{P}_{last}(f) : \max_{P \in S_f} \{Corr(P_{last}(f), P_{last}(tr))\}$$

A genetic algorithm is adopted then to find an optimized solution to the above mentioned minimization problem.

5. INCREMENTAL UPDATING OF SPECTRAL CLUSTERING

The proposed scheduling algorithm although very efficient, is computationally expensive to be used from scratch in every case that the start or finish times are modified. There is a need to exploit the updated information that comes from the visual tracker and the curve-matching algorithm with a fast yet reliable way.

In this paper the following approach is proposed: Since the visual tracker and the curve-matching algorithms detect modifications in the normal workflow (i.e., modifications of the start and finish times of the operations), the initial affinity matrix of operations which indicates the overlapping among the operations is modified as well. In fact, only some distinct elements of the affinity matrix are affected. These changes may

induce alterations to both the degree matrix \mathbf{D} and the eigenvectors matrix \mathbf{E} . As it is proved in (Ning et al., 2010), it is possible to approximate the increment of the eigenvalues and the eigenvectors, without needing to resolve the generalized eigenvalue problem. This fact allows us to refine matrix \mathbf{E} directly and hence, to calculate the new, updated solution matrix \mathbf{S} with an inexpensive approach. The exact subprocess of the incremental update is described by the following steps:

1. Get informed about the modified start and finish times of operations (via the visual tracker and the curve matching algorithm)
2. Update the affinity matrix according to the new overlapping conditions
3. Iteratively refine matrix \mathbf{E} (eigenvectors) and \mathbf{D}
4. Apply the k-means algorithm to the updated solution matrix \mathbf{S} and get the new schedule.

6. EXPERIMENTAL RESULTS

Our experiments are carried out on a real-life industrial environment, the one of Nissan Iberica Automobile Construction company in Barcelona Spain. The dataset collected include three full days video capturing in the industry that describe any complex activity (www.scovis.eu).

In the following, we compare the performance of the proposed incremental spectral clustering algorithm with the heuristic method of Maximum Benefit First (MBF), proposed in (Doulamis and Matsatsinis, 2010) and the Earliest Deadline First (EDF) algorithm. We follow the same setup as in (Doulamis and Matsatsinis, 2010). In particular, we divide the total scheduling time horizon into 10 uniform intervals and we randomly generate 100 operations per interval, i.e., we totally create 1000 workflows. At each interval the scheduler is activated to assign to resource the already submitted workflows (the 100 newly generated and the ones that have not been assigned/executed yet). The start and finish times of the operations are uniformly distributed within three time intervals starting from the current one. We set as delivery deadline per each operation a 20% time extension beyond its finish time. Furthermore, we consider that each operation completed before its deadline yield the economic gain to the industry which also follows a normal distribution, while each violation of the operations deadline burdens with a constant compensation of 20% of the maximum gain among all workflows. This means that negative cost (damages) can be derived.

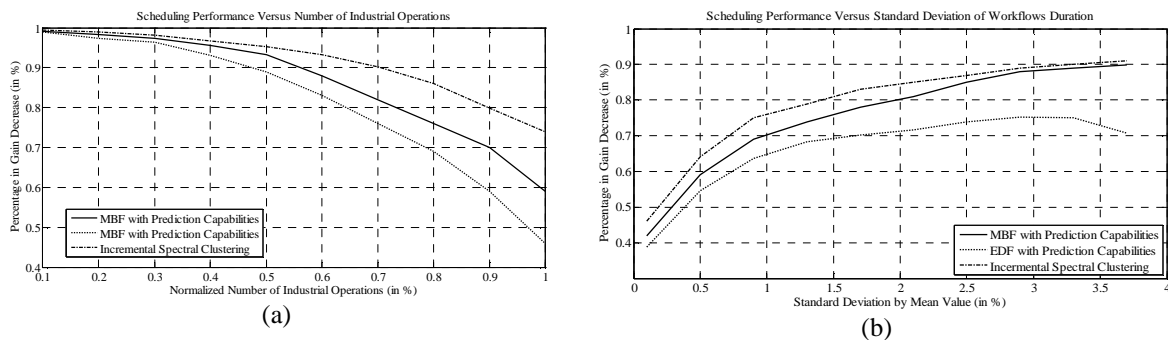


Figure 1. Comparison of scheduling performances

Figure 3(a) shows the scheduling efficiency derived by the use of the proposed incremental spectral clustering algorithm and the MBF method of (Doulamis and Matsatsinis, 2010) versus the number of operations. This number has been normalized with respect to the maximum number of 1000 for clarity purposes. The evaluation is assessed by the percentage of the decrease in total gain that will be offered to the industry if all operations will be executed within their time deadlines. We observe that the proposed algorithm better schedule the operations rather than using the method of (Doulamis and Matsatsinis, 2010). We need to recall that this algorithm deviates from (Ning et al., 2010) in the sense that, due to the different nature of our problem, incremental clustering should be followed. To emulate the effect of the computer vision tools we assume a delay on the 80% of the already executed operations which is uniformly distributed between the requested finish time and a 100% extension of it. The remaining 20% of the currently running workflows are left intact.

In these results, we have randomly generated workflows of Gaussian probability density function (pdf) present a standard deviation equal to their mean value. If we modify the standard deviation different performance will be obtained. In particular as the standard deviation takes smaller and smaller values the gain decrease will increase. On the contrary, for higher values of the standard deviation the submitted workflows will be more distributed in their durations yielding better gain. The effect of the proposed scheduling scheme versus standard deviation values is depicted in Figure 3(b). In this Figure, we have also presents results obtained apart from the MBF along with the EDF algorithm as well.

REFERENCES

- A. D. Doulamis, N. D. Doulamis and S. D. Kollias, "On Line Retractable Neural Networks: Improving the Performance of Neural Network in Image Analysis problems," *IEEE Trans. on Neural Networks*, Vol. 11, No. 1, pp. 137-155, January 2000.
- A. Doulamis, "Dynamic Tracking Re-Adjustment: A Method for Automatic Tracking Recovery in Complex Visual Environments," *Multimedia Tools and Applications*, Springer Press, 1380-7501 (Print) 1573-7721 (Online).
- A. Doulamis, N. Matsatsinis, "Visual Understanding Industrial Workflows under Uncertainty," submitted to the *Future Generation Computer Systems Journal*.
- A. Ilic, T. Andersen, and F. Michahelles, "Increasing Supply-Chain Visibility with Rule-Based RFID Data Analysis", *IEEE Internet Computing* Jan./Feb. 2010.
- D. Culibrk, O. Marques, D. Socek, H. Kalva, B. Furht, "Neural Network Approach to Background Modeling for Video Object Segmentation," *IEEE Transactions on Neural Networks*, Vol. 18, No.6, pp.1614-1627, Nov.2007.
- D. Idoughi, M. Kerkar, C. Kolski, "Towards new web services based supervisory systems in complex industrial organizations: Basic principles and case study," *Computers in Industry*, Vol. 61, No. 3, pp. 235-249, 2010.
- E. N. Malamas, E. G. M. Petrakis, M. Zervakis, L. Petit, J.-D. Legat, "A survey on industrial vision systems, applications and tools," *Image and Vision Computing*, Vol. 21, No. 2, pp. 171-188, 2003.
- F. R. Bach and M. I. Jordan "Learning spectral clustering," In S. Thrun, L. Saul, and B. Schoelkopf (Eds.), *Advances in Neural Information Processing Systems (NIPS)* 16, (long version), 2004.
- Huazhong Ning, Wei Xu, Yun Chi, Yihong Gong and T. S. Huang "'Incremental spectral clustering by efficiently updating the eigen-system," *Patter Recognition*, Vol. 43, No. 1, pp.113-127, 2010.
- J. Gonzalez Fernandez, J. C. Yelmo Garcia, Yo.-S. Martin Garcia, and J. de Gracia Santos, "Transf-ID: Automatic ID and Data Capture for Rail Freight Asset Management", *IEEE Internet Computing Mag.* Jan./Feb. 2010.
- J. W. Herrmann, *Handbook of Production Scheduling*. Springer Press, 1st edition, May 11, 2006.
- J.-M. Proth, "Scheduling: New trends in industrial environment," *Annual Reviews in Control*, Vol. 31, No. 1, pp. 157-166, 2007.
- Ky Fan, "Maximum properties and inequalities for the eigenvalues of completely continuous operators," *Proceedings of the National Academy of Sciences* pp. 760-766, 1951
- M. A. Shaik, C. A. Floudas, J. Kallrath, H.-J. Pitz, "Production scheduling of a large-scale industrial continuous plant: Short-term and medium-term scheduling," *Computer Aided Chemical Engineering*, Vol. 24, pp. 613-618, 2007.
- M. Cui, J. Femiani, J. Hu, P. Wonka, A. Razdan, "Curve matching for open 2D curves," *Pattern Recognition Letters*, Vol. 30, pp. 1-10, 2009.
- M. Drotos, G. Erdos, T. Kis, "Computing lower and upper bounds for a large-scale industrial job shop scheduling problem," *European Journal of Operational Research*, Vol. 197, No.1, pp. 296-306, August 2009.
- M. Karakaya and Hairong Qi, "Target Detection and Counting using a Progressive Certainty Map in Distributed Visual Sensor Networks," *3rd ACM/IEEE International Conference on Distributed Smart Cameras, ICDSC 2009*.
- P. Castro, H. Matos, A.P.F.D. Barbosa-Povoa, "Dynamic modelling and scheduling of an industrial batch system," *Computers and Chemical Engineering*, Vol. 26, pp. 671-686, 2002.
- P. Delias, A. D. Doulamis, N. D. Doulamis, N. Matsatsinis, "Optimizing Resource Conflicts in Workflow Management Systems," *IEEE Transactions on Knowledge and Data Engineering*, vol. 23, no. 3, pp. 417-432, Mar. 2011
- Palma, F.C. Gómez de León Hijes, M. Campos Martínez, L. Guillén Cárceles, "Scheduling of maintenance work: A constraint-based approach," *Expert Systems with Applications*, Vol. 37, pp. 2963-2973, 2010.
- Y. Motai, "Salient feature extraction of industrial objects for an automated assembly system," *Computers in Industry*, Vol. 56, Nos. 8-9, pp. 943-957, December, 2005.

POLAR MOTION STATION-WEBCAM CALIBRATION FOR SOLAR TRACKING

Miguel Romero, Rafael Lemuz, Carlos Guillén, Cesar Bautista, Irene Ayaquica and Daniel Valdés
Benemérita Universidad Autónoma de Puebla
Av. San Claudio y 14 Sur. Puebla, Pue. México. C.P. 72000

ABSTRACT

In this work we propose a calibration algorithm for a high precision solar tracking system. The main goal of the proposed system is to automatically direct a motion stage that continuously capture solar images where the sun appears in the center of the image. A calibration algorithm is proposed to accurately model the transformation from pixel displacements viewed by the camera to the angular parameters of the motion stage. A low cost solar tracking platform integrating a polar motion stage and a webcam is used to assess the accuracy of the proposed algorithm. Results are presented to show the feasibility of our approach to circumvent camera lens distortion and mechanical instabilities of the low cost tracking system.

KEYWORDS

Solar tracking, motion station-camera calibration, surface fitting.

1. INTRODUCTION

Tracking the sun using cameras has diverse applications ranging from the study of sun natural phenomena and more recently in solar energy collectors improving the efficiency of energy capture [1,2]. There are several technologies that have been applied to estimate the relative position of the sun. In [3,4] light dependent resistors and photodiodes are used to approximate position of the sun with the inconvenient that these kind of sensors are known to be sensitive to weather condition like temperature and humidity. To deal with this issue cameras embedded in motion stages can be used to improve the accuracy [5,6,7]. However, when off the shelf cameras and low cost motion stages are used motion stage- camera calibration is important to correct positioning errors due to optical lens distortion and poor electro-mechanical motion control. In this work we propose a calibration algorithm to accurate mapping from pixel displacements to angular coordinates achieving high precision solar tracking using low cost equipment. We use a low cost polar motion stage for pointing the camera to the sun by two angular parameters (azimuth and elevation). Thus, once the relative displacement of the solar region centroid from the image center is computed this information is transformed to their corresponding polar coordinates.

1.1 Solar Tracking Setup

The low cost Tracker-Pod motion stage was used to build a real tracking system. The tracker-Pod uses a RS-232 communication protocol to send motion commands from a PC. The motion range varies from -60 to 60 degrees in x axis and 45 to 45 for the y axis. Figure 1 shows the main components of the tracking system.



Figure 1. The main components of the tracking system (webcam, polar motion stage, and PC)

1.2 Sun Centre Estimation

Solar images are captured using a web camera. Then, the sun central position is calculated using image processing algorithms: first a canny edge detector finds the sun image edges. Next, the mean edge coordinates are computed. In this way, the sun's centroid displacement from the central image position can be estimated. Figure 2 shows the sun edges on a sample image, note that the image center corresponds to (0,0) image coordinates.

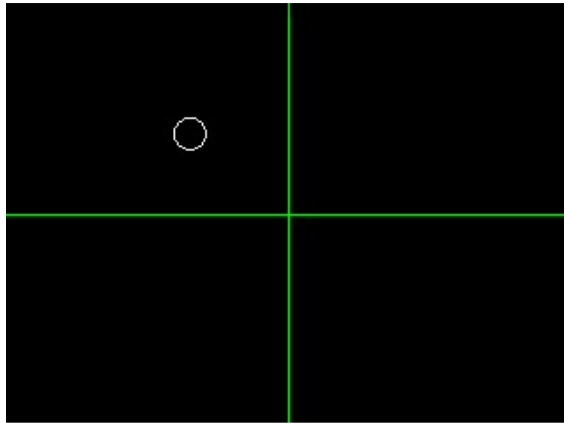


Figure 2. Sun image edges computed using the canny edge detector

During the tracking stage this displacement is automatically computed and corrected by moving the polar motion stage to keep the solar image region at the centre of the new image.

2. PROPOSED CALIBRATION ALGORITHM

Since the motion stage is controlled by the relative displacement of the sun centroid with respect to the image centre, a transformation mapping T is needed to convert pixel displacements (x,y) to angular values (θ_x, θ_y) and T is estimated by a calibration procedure as described in the next section:

2.2 Calibration Area Definition

First, the positioning system is pointed to the sun by moving the motion stage until the sun is viewed approximately in the image center estimating the sun image coordinates (x_0, y_0) and retrieving the polar motion coordinates (θ_x, θ_y) . Then a set of eight images are captured moving the polar motion stage using the known angular displacement (θ_b, θ_i) in a virtual rectangle registering the sun image coordinates (x_0, y_0) and its corresponding (θ_b, θ_i) angular values. Table 1 shows the motion displacements used in the tracking system.

Table 1. Motion displacements used for sun camera calibration

No.	Image Name	Motion Displacement
0	Central Image	(0, 0)
1	Down-Right image	(+15, -15)
2	Down-Left image I	(+15, +15)
3	Up-Right Image	(-15, -15)
4	Up-Left Image	(-15, +15)
5	Down Image	(0, +15)
6	Up image	(0, -15)
7	Right Image	(+15, 0)
8	Left image	(-15, 0)

Figure 3 shows the calibration area that consists of: a virtual rectangle formed by the angular motion and image measurements 3, 4, 5 y 6 in Table 1. Thus, during sun tracking when the sun image center is estimated and lay in this area, the corresponding motion displacement can be computed and sent it to the motion stage to point the camera to the sun's centroid.

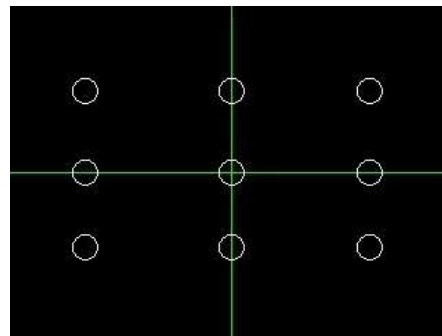


Figure 3. Solar edges for every local displacement in an over-impose single image used for motion stage-camera calibration and the virtual calibration rectangle

2.3 Transformation Mapping Function

The polar motion-webcam calibration can be represented by a transformation function T :

$$T(x, y) \rightarrow (\theta_x, \theta_y) \quad (1)$$

where:

x is the x-axis (horizontal) image coordinate of the sun image region.

y is the y-axis (vertical) image coordinate of the sun image region.

θ_x is the azimuth angle in degrees of the Polar-Motion Stage.

θ_y is the elevation angle in degrees of the Polar-Motion Stage.

The transformation T generate two different angular values one for each axis, then is preferable to define two transformation function one for each axis. Therefore, T_x is a mapping function to estimate the azimuth angle, and T_y is the mapping function to obtain the elevation angle

$$\begin{aligned} T_x: \mathbb{R}^2 &\rightarrow \mathbb{R}; & T_x(x, y) &= \Theta_x \\ T_y: \mathbb{R}^2 &\rightarrow \mathbb{R}; & T_y(x, y) &= \Theta_y \end{aligned} \quad (2)$$

If the solar image coordinates (x, y) would be restricted to a line, then the mapping function could be expressed by a linear transform. However, the relation between the solar image coordinates and the angular polar-motion stage is not linear due to camera lens distortion, image noise and mechanical instabilities.

Then, the transformation relation was modeled by a quadratic expression:

$$\begin{aligned} T_x(x, y) &= (a_1x + b_1y + c_1)(d_1x + e_1y + f_1) \\ T_y(x, y) &= (a_2x + b_2y + c_2)(d_2x + e_2y + f_2) \end{aligned} \quad (3)$$

From equations (2) and (3) the calibration transformations translating solar image coordinates to angular positions become:

$$\begin{aligned} \Theta_x &= h_1 + i_1y + j_1y^2 + k_1x + l_1xy + m_1xy^2 + n_1x^2 + o_1x^2y + p_1x^2y^2 \\ \Theta_y &= h_2 + i_2y + j_2y^2 + k_2x + l_2xy + m_2xy^2 + n_2x^2 + o_2x^2y + p_2x^2y^2 \end{aligned} \quad (4)$$

To get an estimate of the calibration parameters the constant coefficient must be calculated for expressions in (4). This can be done by solving a linear system of equations (5) where all the image coordinates x,y and their corresponding angular terms are known from table 1 which yield to the following equation:

$$\begin{bmatrix} 1 & y_1 & y_1^2 & x_1 & x_1y_1 & x_1y_1^2 & x_1^2 & x_1^2y_1 & x_1^2y_1^2 \\ 1 & y_2 & y_2^2 & x_2 & x_2y_2 & x_2y_2^2 & x_2^2 & x_2^2y_2 & \vdots \\ 1 & y_3 & y_3^2 & x_3 & x_3y_3 & x_3y_3^2 & \vdots & \vdots & \vdots \\ 1 & y_4 & y_4^2 & x_4 & x_4y_4 & \vdots & \vdots & \vdots & \vdots \\ 1 & y_5 & y_5^2 & x_5 & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 1 & y_n & y_n^2 & x_n & x_ny_n & x_ny_n^2 & x_n^2 & x_n^2y_n & x_n^2y_n^2 \end{bmatrix} \begin{bmatrix} h_k \\ i_k \\ j_k \\ k_k \\ l_k \\ m_k \\ n_k \\ o_k \\ p_k \end{bmatrix} = \begin{bmatrix} \Theta_{w1} \\ \Theta_{w2} \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \vdots \\ \Theta_{wn} \end{bmatrix} \quad (5)$$

Solving the linear system for the coefficients h_k, i_k, \dots, p_k for $k=1,2$ and $w=x,y$ the calibration mapping can be computed and used to obtain the polar coordinates (θ_x, θ_y) given the image coordinates (x, y) .

2.4 Calibration System Evaluation

To assess the performance of the calibration algorithm we carried out the following procedure:

We point the camera in such a way that sun becomes visible at different locations in the images. At this stage we ensure that sun's location in the images lay within the virtual calibration window and its image coordinates are different to those used for system calibration. After that, we run the tracking algorithm that gets the sun's central coordinates from subsequent images and estimate the transformation from pixels to degrees using the interpolation polynomial (4). Finally, since the system's objective is to keep the sun in the center of the image the tracking error is computed by comparing the final sun position against the image center after the positioning system reaches its target position. Thus, we observe that the average error is ± 4 pixels which are significantly lower than ± 10 pixels error using a linear algorithm. Figure () shows 9 of the 20 images used for tracking system evaluation.

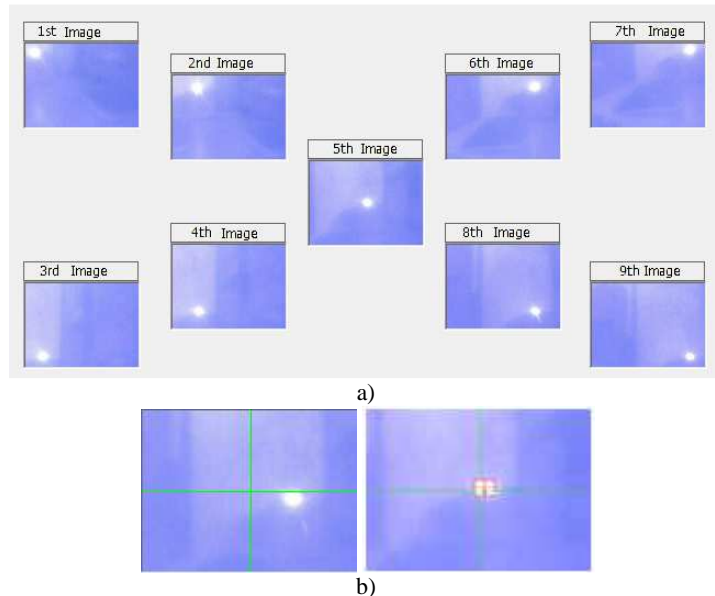


Figure 4. a) A subset of images used to assess the tracking system performance. b) A close up for the fifth image showing the initial and final sun location after tracking positioning using the proposed calibration algorithm.

3. CONCLUSIONS

In this work a calibration algorithm for solar tracking was presented. The calibration algorithm allows precise translation from pixel displacements to polar motion parameters. In the actual stage of the development a set of solar images are the inputs to the algorithm however, the calibration algorithm can be extended to auto-calibrate motion stages for a variety of applications which is part of our current work using salient features and robust descriptors.

ACKNOWLEDGEMENT

The authors thank CONACYT (CVU: 332238), PROMEP (BUAP-PTC-230), and VIEP (00189), for the financial support given to this research.

REFERENCES

- [1] Ethrog U., 2006. CCD Camera Calibration Based on the sun's images, *IAPRS*, vol. 36, pp. 25-27.
- [2] Roth P. *et al.*, 2004. Design and construction of a system for suntracking, *Renewable Energy*, vol. 29, pp. 393-402.
- [3] Lynch W.A. and Salameh Z. M., 1990. Simple electro-optically controlled dual-axis sun tracker, *Solar Energy*, vol. 45, pp. 65-69.
- [4] Kalogirou S.A., 1996. Design and construction of a one-axis sun-tracking system, *Solar Energy*, vol. 57, pp. 465-469.
- [5] H. Arbab, *et al.*, 2009. A computer tracking system of solar dish with two axis degree freedoms based on picture processing of bar shadow, *Renewable Energy*, vol. 34, pp. 1114-1118.
- [6] Clifford M.J. and Eastwood D., 1994. Design of a novel passive solar tracker, *Solar Energy*, vol. 77, pp. 269-280.
- [7] Ballesteros E. *et al.*, 1994. Electronic design of a solar correlation tracker based on a video motion estimation processor, *Microprocessors and Microsystems*, vol. 18, pp. 243-251.

UNSUPERVISED OPTIMAL DISCRIMINANT PLANE WITH UNCORRELATED FEATURES

Su-Qun Cao¹, Yun-Feng Bu¹, Xiao-Ming Zuo¹ and Quan-Yin Zhu²

¹*Faculty of Mechanical Engineering, Huaiyin Institute of Technology
No.1 MeiCheng Road, Huaian 223003, China*

²*Faculty of Computer Engineering, Huaiyin Institute of Technology
No.1 MeiCheng Road, Huaian 223003, China*

ABSTRACT

Optimal discriminant plane plays an important role in feature extraction and has been widely used in the pattern recognition field. On this basis, Jin et al. proposed uncorrelated optimal discriminant plane which is based on Fisher criterion function and the conjugated orthogonal constraint of the total-class scatter matrix. This method needs the class information to calculate the Fisher optimal discriminant vector. Thus, it can only be used in supervised pattern. This paper will extend it to the unsupervised case. The proposed method aims to optimize the defined fuzzy Fisher criterion function to figure out an optimal discriminant vector and fuzzy scatter matrices in an unsupervised way. With the conjugated orthogonal constraint of the fuzzy total-class scatter matrix, a novel feature extraction algorithm based on unsupervised optimal discriminant plane with uncorrelated features is proposed. Experimental results on UCI datasets show its validity.

KEYWORDS

Optimal discriminant plane; Uncorrelated feature; Feature extraction; Unsupervised pattern

1. INTRODUCTION

Linear discriminant analysis (LDA) [1] is the most successful supervised feature extraction method. It attempts to find out an optimal feature space using the Fisher criterion function and it can extract the discriminant features from the data sets which best separate two or more classes of data samples. Sammon[2] extended LDA by finding out a classical optimal discriminant plane (CODP) by computing orthogonal discriminant vectors from data. Jin et al. [3] proved that two discriminant vectors in CODP are essentially correlated and then proposed the uncorrelated optimal discriminant plane (UODP) method which are more powerful than CODP in such applications as face recognition.

But UODP are based on Fisher discriminant criterion. It can only be used in a supervised way because Fisher optimal discriminant vector should be calculated from labeled data. Such a limitation makes it very inappropriate for unlabeled data. This study here attempts to extend it to the unsupervised case.

2. FISHER CRITERION AND OPTIMAL DISCRIMINANT PLANE

Assume that x is a d -dimensional sample set with N elements belonging to c classes, S_b is the between-class scatter matrix, S_w is the within-class scatter matrix and S_t is the total-class scatter matrix. The Fisher criterion can be defined as follows:

$$J_{FC}(\omega) = \frac{\omega^T S_b \omega}{\omega^T S_w \omega} \quad (1)$$

Where ω is an arbitrary vector in d -dimensional space. The Fisher optimal discriminant vector ω_1 corresponding to maximum of $J_{FC}(\omega)$, can be derived as the eigenvalue of the following eigen-system equation:

$$S_b \omega_1 = \lambda S_w \omega_1 \quad (2)$$

Based on the Fisher' vector ω_1 , Jin et al. [3] obtained the second direction ω_2 which maximizes the Fisher criterion $J_{FC}(\omega)$ with the following conjugated orthogonal constraint:

$$\omega_2^T S_w \omega_1 = 0 \quad (3)$$

ω_1 and ω_2 make up an uncorrelated optimal discriminant plane(UODP).

3. FUZZY FISHER CRITERION AND UNSUPERVISED OPTIMAL DISCRIMINANT PLANE

Suppose that the membership function $u_{ij} \in [0,1]$ with $\sum_{i=1}^c u_{ij} = 1$ for all j and the fuzzy index $m > 1$ is a given real value, where u_{ij} denotes the degree of the j th d -dimensional sample belonging to the i th class. The fuzzy between-class scatter matrix S_{fb} , the fuzzy within-class scatter matrix S_{fw} and the fuzzy total scatter matrix S_{ft} [4] can be defined as follows:

$$S_{fb} = \sum_{i=1}^c \sum_{j=1}^N u_{ij}^m (m_i - \bar{x})(m_i - \bar{x})^T \quad (4)$$

$$S_{fw} = \sum_{i=1}^c \sum_{j=1}^N u_{ij}^m (x_j - m_i)(x_j - m_i)^T \quad (5)$$

$$S_{ft} = \sum_{i=1}^c \sum_{j=1}^N u_{ij}^m (x_j - \bar{x})(x_j - \bar{x})^T \quad (6)$$

Where m_i represents the mean of the i th class sample set and \bar{x} the mean of all samples.

Thus, a novel fuzzy Fisher criterion called fuzzy FLD can be derived as follows [5]:

$$J_{FFC}(\omega) = \frac{\omega^T S_{fb} \omega}{\omega^T S_{fw} \omega} \quad (7)$$

It is obvious that maximizing J_{FFC} directly in Eq. (7) is not a trivial mathematical derivation task due to the existence of its denominator. However, we can reasonably relax this problem by applying the following

Lagrange multipliers λ and $\beta_j (j=1,2,L,n)$ together with the constraint $\sum_{i=1}^c u_{ij} = 1$ to Eq. (7):

$$F = \omega^T S_{fb} \omega - \lambda \omega^T S_{fw} \omega + \sum_{j=1}^n \beta_j (\sum_{i=1}^c u_{ij} - 1) \quad (8)$$

Setting $\frac{\partial F}{\partial \omega}$, $\frac{\partial F}{\partial m_i}$, $\frac{\partial F}{\partial u_{ij}}$ to be zero, we respectively have

$$S_{fb} \omega = \lambda S_{fw} \omega \quad (9)$$

Where λ may be taken as the largest eigenvalue.

$$m_i = \frac{\sum_{j=1}^N u_{ij}^m (x_j - \frac{1}{\lambda} \bar{x})}{\sum_{j=1}^N u_{ij}^m (1 - \frac{1}{\lambda})} \quad (10)$$

$$u_{ij} = F_1 / F_2 \quad (11)$$

Where

$$F_1 = (\omega^T (x_j - m_i)(x_j - m_i)^T \omega - \frac{1}{\lambda} \omega^T (m_i - \bar{x})(m_i - \bar{x})^T \omega)^{-\frac{1}{m-1}} \quad (12)$$

$$F_2 = \sum_{k=1}^c (\omega^T (x_j - m_k)(x_j - m_k)^T \omega - \frac{1}{\lambda} \omega^T (m_k - \bar{x})(m_k - \bar{x})^T \omega)^{-\frac{1}{m-1}} \quad (13)$$

When Eq. (11) is used, as stated in the above, u_{ij} should satisfy $u_{ij} \in [0,1]$. Hence, in order to satisfy this constraint, let

$u_{ij} = 1$ and $u_{i'j} = 0$ for all $i' \neq i$, if

$$\omega^T (x_j - m_i)(x_j - m_i)^T \omega \leq \frac{1}{\lambda} \omega^T (m_i - \bar{x})(m_i - \bar{x})^T \omega \quad (14)$$

From the above analysis, the fuzzy scatter matrices and the fuzzy fisher optimal discriminant vector ω_1 based on fuzzy fisher criterion can be obtained in an unsupervised way. Based on $\omega_1, S_{fw}, S_{fb}, S_{fi}$, we can obtain the second direction ω_2 with the conjugated orthogonal constraint instead of the orthogonal constraint by Theorem 1 according to Jin et al.'s ideas [3]. Thus, ω_1 and ω_2 can make up an Unsupervised Uncorrelated Optimal Discriminant Plane (UUODP).

Theorem 1. The second discriminant vector ω_2 which maximizes the fuzzy Fisher criterion $J_{FFC}(\omega)$ with the following conjugated orthogonal constraint:

$$\omega_2^T S_{fi} \omega_1 = 0 \quad (15)$$

Is the eigenvector corresponding to maximal eigenvalue of the following eigen-system:

$$PS_{fb} \omega_2 = \lambda S_{fw} \omega_2 \quad (16)$$

Where

$$P = I - \frac{S_{fi} \omega_1 \omega_1^T S_{fi} S_{fw}^{-1}}{\omega_1^T S_{fi} S_{fw}^{-1} S_{fi} \omega_1} \quad (17)$$

Based on the fuzzy Fisher vectors ω_1, ω_2 , the following linear transformation from d -dimension to 2-dimension can be defined:

$$Y = \begin{bmatrix} \omega_1^T \\ \omega_2^T \end{bmatrix} X \quad (18)$$

In summary, a novel feature extraction algorithm based on UUODP can be described as follows:

Algorithm UUODP

1. Set the given threshold \mathcal{E} , initialize $U = [\mu_{ij}]_{c \times N}$ and $m = (m_1, m_2, \dots, m_c)$ using K-means;
2. Compute S_{fb}, S_{fw}, S_{fi} using Eq. (4), Eq. (5) and Eq. (6) respectively;
3. Compute the largest eigenvalue λ_1 and the corresponding ω_1 using Eq. (9);
4. Update m_i and μ_{ij} using Eq. (10) and Eq. (11) respectively;
5. Compute J_{FFC} using Eq. (7);

6. If the difference between J_{FFC} and its last value less than ϵ or the number of iterations no less than the given value, compute the second fuzzy Fisher vector ω_2 in term of theorem 1. Thus, the obtained ω_1 and ω_2 make up UUODP. Perform the linear transformation from the original data space to the 2-dimensional feature space using Eq. (18) and then terminate, otherwise go to Step 2.

4. EXPERIMENTS

In this experiment, three benchmarking UCI datasets Iris, Ecoli and Wdbc [6] were chosen to test the performance of feature extraction methods. Table 1 illustrates the basic information of the three datasets. Supervised feature extraction method UODP and unsupervised feature extraction methods including PCA and UUODP have been applied to reduce the dimensions of these datasets. Clustering experiments have been done on the reduced-dimension datasets.

Table 1. the basic information of three datasets

Datasets	Classes	Samples	Features
Iris	3	150	4
Ecoli	8	336	7
Wdbc	2	569	30

Fuzzy C-Means (FCM) and K-means clustering on these reduced-dimension datasets were performed in this experiment. The clustering performances are evaluated by the following Rand index [7]:

$$Rand(P_1, P_2) = \frac{a+b}{n \times (n-1) / 2} \quad (19)$$

Where P_1, P_2 denote the original datasets labels and the clustering results on the reduced-dimension datasets, a denotes the number of any two patterns in the original dataset belonging to the same cluster in P_1, P_2 , b denotes the number of any two patterns in the original dataset belonging to two different clusters in P_1, P_2 , and n is the number of all patterns in the original dataset. Obviously, $Rand(P_1, P_2) \in [0, 1]$. And $Rand(P_1, P_2) = 1$ when P_1 is the same as P_2 . The bigger $Rand(P_1, P_2)$ is, the smaller the difference between P_1 and P_2 is. For comparison, Table 2 lists the average values of Rand index after 10 executions of both clustering algorithms.

Table 2. the clustering performances of FCM, K-means for three reduced-dimension datasets extracted by UODP, PCA and UUODP

Datasets	No Feature Extraction		UODP		PCA		UUODP	
	FCM	K-Means	FCM	K-Means	FCM	K-Means	FCM	K-Means
Iris	0.950	0.858	0.946	0.910	0.942	0.915	0.946	0.891
Ecoli	0.788	0.804	0.792	0.793	0.770	0.774	0.766	0.767
Wdbc	0.750	0.750	0.939	0.942	0.750	0.750	0.863	0.863

Compared to supervised feature extraction methods, unsupervised feature extraction methods such as PCA and UUODP have a certain decline in performance because of the missing class information of samples in unsupervised pattern. But whether clustering with FCM or K-means, the clustering results of the reduced-dimension datasets extracted by UUODP have similar Rand index values to those extracted by PCA.

Especially in Wdbc dataset, compared to PCA, the clustering Rand Index values of the reduced-dimension dataset extracted by UUODP have improved more than 10% which means that the clustering results have better consistency to the original datasets.

5. CONCLUSIONS

This paper presents a method to extend uncorrelated optimal discriminant plane feature extraction technology for unsupervised pattern. The basic idea is to optimize the defined fuzzy Fisher criterion function to figure out an optimal discriminant vector and fuzzy scatter matrixes in unsupervised pattern. Based on these, a novel feature extraction method based on unsupervised optimal discriminant plane with uncorrelated features is proposed. It is changed that the traditional optimal discriminant plane can only be used in supervised pattern. Experimental results show that although this method is unable to surpass traditional supervised optimal discriminant plane, it has comparable performance compared to principal components analysis unsupervised feature extraction algorithm.

Additionally, the proposed method assumes that prior information about the number of classes is available. Thus, how to overcome multi-classes problems is worthy to be studied in future.

REFERENCES

- [1] R.A. Fisher, The use of multiple measurements in taxonomic problems. *Ann. Eugenics*, 1936, 7: 178-188.
- [2] J.W. Sammon, "An optimal discriminant plane", *IEEE Trans on Computers*, 1970, 19(9): 826-829.
- [3] Jin Zhong, Lou Zhen, Yang Jingyu, "An optimal discriminant plane with uncorrelated features," *PR & AI*, 1999, 12(3): 334-338.(In Chinese)
- [4] Kuo-Lung Wu, Jian Yu, Miin-Shen Yang. A novel fuzzy clustering algorithm based on a fuzzy scatter matrix with optimality tests. *Pattern Recognition Letters*, 2005, 26(4): 639-652.
- [5] Cao Suqun, Wang Shitong, Chen Xiaofeng et al., "Fuzzy Fisher criterion based semi-fuzzy clustering algorithm," *Journal of Electronics & Information Technology*, 2008, 30(9): 2162-2165.
- [6] C.L. Blake and C.J. Merz, *UCI repository of machine learning databases*, <http://www.ics.uci.edu/~mlearn/MLRepository.html>, 1998.
- [7] W. Rand, Objective Criteria for the Evaluation of Clustering Methods, *Journal of the American Statistical Association*, 1971, 66(336): 846-850.

AN OFF-LINE PERSIAN HANDWRITTEN FORGERY DETECTION METHOD

Behzad Helli and Mohsen Ebrahimi Moghaddam

Electrical and Computer Engineering Department, Shahid Beheshti University G.C, Tehran, Iran

ABSTRACT

The security of handwritten documents is very important in authentication systems. In this paper, a forgery detection method is proposed for Persian handwritten documents, which uses two types of novel features: Macro and Micro, Macro features show the structure of handwritten while micro ones show more details. Also, the micro features try to extract some online properties from offline data such as pen pressure and velocity. After extracting those features Weighted Euclidean Distance (WED) classifier is used to find forgery ones. It is very important that the weights of this classifier have been adjusted based on true data because forgery data are not identified in adjusting phase. To test the proposed method a Persian handwritten data set was prepared using three kinds of forgeries; Mimic, unskilled, and skilled. The method performance using different reference words showed the best result in correct rejection was 82% while the correct acceptance was 94%. Also, we believe that this method can be extended to other languages by adjusting some parameters.

KEYWORDS

Handwriting verification, image processing, forgery

1. INTRODUCTION

The need to verify the writer of a certain document, has led to the development of an automatic personal verification systems. These certain documents are mostly business documents, which might have been claimed to be forged. Signature verification and handwritten verification are the main branches of handwritten verification systems. On the other hand, these systems can be applied using on-line or off-line data. The on-line data is a data which have been gathered through electronic pens, touch-screen pads, or monitors while data is being written, but the off-line data is gathered using a scanner or a digital camera after the document has been written. The on-line systems can gather more information than the off-line ones. The on-line data is gathered while the elapsed time is taken into consideration. Also in some devices, the pressure can be measured in each position. A study found that, 59% of forgeries could be rejected if the writing duration difference with mean of writing duration is more than 20%[1].

In the literature, there are lots of on-line signature and handwritten verification, some off-line signature verification, and too few off-line handwritten verification researches. For example, Ibrahim Abuhaiba has designed an off-line signature verification based on graph matching in Ref.[2]. At first, the method normalizes the signature to well orient it as all the other signatures, then it is considered as a graph and algorithm tries to match the parts of the graph. The author asked five people to sign 15 times, and then each of them was asked to forge other ones signature after some practices. In total, 15 forged signatures were generated for each individual signature. Using this dataset, they got about 35% as equal error rate in signature forgery detection.

Schlapbach, Andreas, et. al. have trained an HMM (Hidden Markov Model) over the training data gathered from IAM database[3], and tested the system using forgeries gathered from 20 unskilled and 20 skilled forgers. The best ERR result in unskilled forgeries was reported to be about 2.0% where the False accept rate and False reject rate were 0.8% and 3.4% respectively.

Forgeries are usually categorized into two classes, *unskilled* and *skilled*. Unskilled forgeries are done by random people who have no experience doing forgery, but skilled ones are done by professionals. But there is another category of handwriting forgery that is called *mimic*. The method is very simple: mimic the

handwriting while an original one is under the forged one. In off-line systems, like a written and signed bank cheque, the mimic method is available for the forger.

In this paper, we have proposed a method for forgery detection that is based on Micro and Macro features. These features try to extract online data from offline handwritten, the proposed method converts the handwritten to a graph and uses the longest path in graph as a criteria. This method is able to detect most of forgeries such as mimic ones. To evaluate the proposed method, we gathered a dataset of Persian handwritings and the experimental results were satisfactory. We believe that the proposed method can be used in other languages by using some customization.

The remaining part of the paper is organized as follows: Section 2 explains the features that are used and the way of extracting. In section 3, the test bench and the classification method are described and the system's performance and conclusion is section 4.

2. FEATURE EXTRACTION

2.1 Macro Features

In feature extraction phase, the Gabor and XGabor filters are used with different parameter values for macro feature extraction.

Gabor and XGabor filters are deeply explained in [4-7], though a brief explanation follows.

Gabor filter is obtained by modulating a sinusoid function with a Gaussian one.

$$g(x, y, \theta, \phi) = \exp\left(-\frac{x^2 + y^2}{\sigma^2}\right) \times \exp(2\pi\phi i(x \cos \theta + y \sin \theta)) \quad (1)$$

An XGabor filter has a similar structure; despite that the sinusoid function is a centered one.

$$xg(x, y, \phi, r_x, r_y) = \exp\left(-\frac{x^2 + y^2}{\sigma^2}\right) \times \sin\left(\phi \frac{r_x x^2 + r_y y^2}{r_x + r_y}\right) \quad (2)$$

In this phase, 18 Gabor filters with different directions are used. The direction of Gabor filter increased by 10 degree. The XGabor filter was used in two situations: total XGabor and partial XGabor and each one in 3 basic forms: circular, horizontal and vertical. In total, 33 filters were convolved into each image segments, and the results are summed to make one of the features.

2.2 Micro Features

As noticed in [1], features generated from on-line data are almost necessary for a reliable verification. So we tried to extract features that are not comparable with on-line features (because obviously on-line data has more information), but they are pseudo online and can illustrate some information on the change of velocity and pressure of the pen. It is important to notice that on-line features have the exact velocity and pressure of the pen, but these features can only provide the change ratio of velocity and pressure of the pen.

The micro feature extraction is made up of three main steps. The first step is to find a route that is close to the route that pen has been moved. After finding such route, for each pixel of the route, a pressure related value is calculated. After calculating these values, their relation illustrates the change of pressure and velocity through that route.

Step 1 – Finding a possible route

There have been some methods that have tried to map the image to a graph that might be the graph which the writer have moved the pen across [2,8], but the proposed approach here is more accurate than others.

To do so, at first, the skeleton of the handwriting is generated. Using this skeleton, the aim is to find the movement of the pen as close as it has happened, so the generated skeleton should be in the middle of the thickness of the handwriting. So, at first in order to fill the gaps of the image, an opening operator is applied on the image, after that a simple skeletonizing algorithm is used to generate the skeleton. Figure 1 shows a sample of handwritten and results of applying opening operator and finding skeleton.

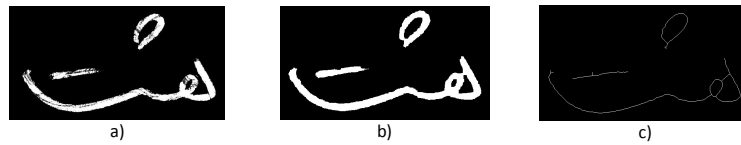


Figure 1. a) The original handwritten image b) The opened image (gaps are filled) c) The extracted skeleton

Any two-dimension array (e.g. image) can be seen as a planar graph, which each block (pixel) represents a node in the graph and each two neighbor nodes are connected using a simple edge. The extracted skeleton is considered as such graph and the longest path in the skeleton is selected. The problem of finding the longest path in a graph is NP-complete, so at first we simplified the explained graph and the simplified graph is used to decrease the run time of algorithm.

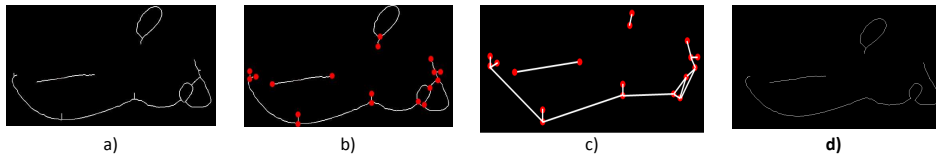


Figure 2. a) a sample of the extracted skeleton, b) each red ball is representing a node that doesn't have exactly 2 neighbors, c) each new edge represents a unique path between the two nodes, d) longest path extracted from sample

Figure 2 shows the steps of the graph complexity reduction method. Figure 2.a is the skeleton extracted from a written test word. All the pixels that have exactly two neighbor nodes are removed, so other nodes (having one neighbor or more than two neighbors) remain in the system.

Figure 2.b shows the nodes that remain in the image. In this phase, all the pixels between two red nodes are represented by one edge. So, the generated graph is like as Figure 2.c. So the resulted sample graph has 17 nodes and 24 edges, because the number of nodes and edges has reduced, the time needed to check for all possible routes became less than 1 second. After that, for each strongly connected component, the longest path is calculated. For example, Figure 2.d shows the result of the longest path.

Step 2 – Pressure based value

The thickness and density of a written character, shows the amount of pressure the writer used, but these attributes are not just dependent to pressure, but also type of pen, paper, mood of writer, and even the color of the pen affects these attributes. But in a written character, pen, paper, and mood are still attributes; so change of thickness and density of the writing, shows the change of pressure. In writer verification, this change ratio is very important.

For each pixel in the route, starting from one side, a virtual circle is placed where its center is fixed on the pixel. The value of the sum of the gray values inside the circle is calculated and the area of the circle divides it. We call this value D_{ij} .

$$D_{ij} = \frac{\sum_{k \in Circle_{ij}} (pixel_value(k))}{Area(Circle_{ij})} \tag{3}$$

$Circle_{ij}$ is the circle, centered on the i^{th} pixel of the route, with the radius of \sqrt{j} .

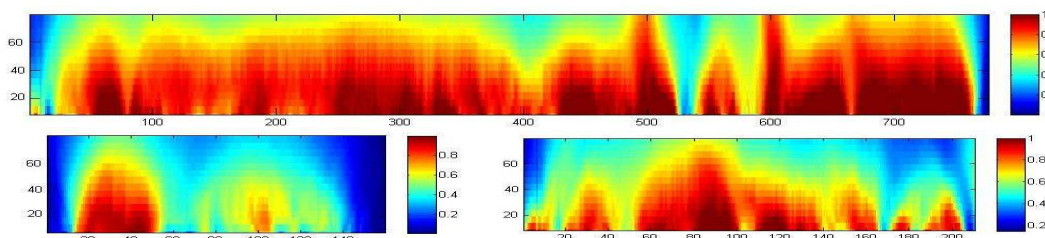


Figure 3. three density matrices of the three extracted sample routes, the longest route, the straight route and the circular route

Calculating this value over generated routes generates a 2D matrix of values. A sample of such matrix is shown in Figure 3.

In the above figure, the horizontal axis represents the length of the extracted route, the vertical axis represents the different “j” values considered in $Circle_{ij}$, and each point shows the D_{ij} value.

Step 3 – Feature value generation

Figure 4 shows a sample of a forgery and original form of a segment. As it can be seen, in the original document, the figure has more fade-in and fade-outs, in other words, it has more variations. But in the forged documents, such as the sample shown below, the variations (consecutive fade-in and outs) are less. That is because in a natural handwriting, the movement of pen is more rapid than the movement of a forger pen. That rapidness makes more changes in stroke width and that makes the represented feature to be more variant. As it can be seen in Figure 4, at the beginning of the signal and between 500 and 600, the changes are very rapid in the original data.

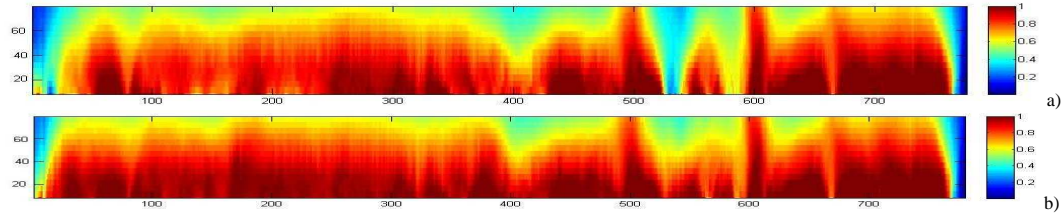


Figure 4. a sample of original handwriting extracted feature and a forged one. As it can be seen, the original one (a) is more variant than the forged one (b). (The first one changes rapidly)

We defined 7 feature values to extract numeral values from these figures.

Number of fade in/outs: this value is the number of consecutive fade-in and fade-outs along the path. For example in Figure 4, the value of this feature in the original handwriting is 25 while in the forged item is 19. To calculate this value, a measure of comparison was defined so that each pixel can be compared to its neighbors. Using this value, the number of peaks in the signals, represents the number of fade-in and outs.

$$v_i = \sum_j jD_{ij} \left(\sum_j D_{ij} \right)^{-1} \quad (4)$$

Average value of fade in/out lengths: each fade in/out phase, has a length. To calculate this value, first, the derivative of the generated signal is calculated, and then in the resulted signal, zero values represent a peak in original signal. So using this signal, the lengths are calculated and the average value is generated.

Standard deviation of fade in/out lengths: in an original document, there are different lengths of fade in/outs but in a forged one, they are not that different. That makes the standard deviation, a useful value to separate forged documents from the original one.

Average Pressure over the fragment:

Pressure: the pressure value for each point is calculated using below equations.

$$\frac{\partial D_{ij}}{\partial j} = D_{ij} - D_{i(j-1)} \quad (5)$$

$$Mean \frac{\partial D_{ij}}{\partial j} = \frac{1}{\max_j - 1} \sum_j \frac{\partial D_{ij}}{\partial j} \quad (6)$$

$$Pr \text{ essure }_j = \sigma \left(\frac{\partial D_{ij}}{\partial j} \right) = \frac{1}{\max_j - 1} \sum_j \left(\frac{\partial D_{ij}}{\partial j} - Mean \left(\frac{\partial D_{ij}}{\partial j} \right) \right) \quad (7)$$

Equation (4) shows how the derivative across “j” value is calculated. Equation (5) is the mean value on the derivative values, and equation (6) is the standard deviation on derivation values.

So, according to above equations, the pressure value is in relation with the standard deviation of the differentiation in the pressure-based mark of the pen.

For this feature, the mean value of the pressures over each fragment is calculated.

Standard deviation of pressures: as for fad in/out lengths, the standard deviation of the pressures is calculated.

Averages differences of the pressures: depending on the writer’s style, the ratio of pressure change differs. So this value is calculated, which is the average value of the difference of each two consecutive points.

Standard deviation of the pressures: the standard deviation also is calculated.

For example, for the shown sample in Figure 2, 21 micro features are calculated.

3. TEST BENCH GENERATION & CLASSIFICATION

As explained in section 3, handwriting forgeries are classified in three groups. Therefore, a test-bench has been designed to gather all these forms. On the other hand, in order to build a more useful image database, some words that are used more frequently in financial documents have been selected. An A4 sheet was designed to gather original and forged data. Having 21 lines for each word, 25 people were asked to write in their regular handwritings in odd lines, and different types of forgeries were done in the even lines. We had 11 lines of original handwriting and 10 lines of forgeries which are divided into 3 lines of unskilled, 3 lines of skilled and 4 lines of mimic forgeries.

While scanning the documents, it was important to scan them in the correct angle, so there would be no need to rotate the images, afterward. The fact is that, rotating the image in digital form changes its micro features. After scanning the data, the odd rows were localized using the black boxes at the right end of them.

On the other hand, the micro features explained in section 5, are influenced by the resolution of the image. Hence, the writing documents were scanned at 1200 dpi (dot per inch).

5 samples of each word original form were used in training phase and the other 6, plus all forgery data were used to test the system.

To classify these handwritings, unlike identification problems, we do not have both positive (original handwritings) and negative (forged handwritings) samples in training phase. In many classification methods, negative samples such as other people handwritings are given to the system, so it can find the best classification that the positive features are most likely inside and negative ones are outside. But in verification system, we cannot give the system any forgery sample to get trained with. Because different skilled forgers have different styles in forging a document, it is meaningless to train a system with someone forgery style while the system will be tested with others forgeries.

Hence, we need a classification method which only uses positive samples and tries to fit itself to the positive samples. In order to achieve such conditions, the WED (Weighted Euclidian Distance) classifier is used. To use the WED classifier, using the 5 samples of person words, the weights of the WED are determined as follows:

$$Weight_i = \frac{1}{\sqrt{N-1} \sqrt{\sum_j (F_{ij} - Mean(F, i))^2}} \quad (8)$$

Where in this equation, “N” is the number of train data (here 5), “j” is the index moving on training data, “i” is the index showing the sequence number in features, F is the feature values and “Mean(F,i)” is the average value of all i^{th} features in training data of same person. Then the maximum distance of these 5 samples is set to be the threshold of the WED decision, using this threshold value, the data is considered to be in or out of the class. The method of calculating distance is shown in equation 9.

$$Dist_{ab} = \sqrt{\sum_i [Weight_i (F_{ai} - F_{bi})]^2} \quad (9)$$

Where “a” is the sample data, “b” is the cluster center (average of features) of a known class, and “Weight” is the calculated weights. After calculating this distance value, if it is less than the clusters threshold, it belongs to that cluster.

4. EXPERIMENTAL RESULTS & CONCLUSION

It can be seen (Table 1) that the word “Three” is the most reliable word to verify the handwriting according to it and the words “Two”, “Seven”, “Nine” and “Billion” are the most unreliable words that can be used to verify handwriting. It can be concluded that if the word has longer strokes of pen, it is harder to forge it flawlessly. However, in the literature, there are not much verification experiments using forged data, and no standard data set is available. Also, to the best of our knowledge, mimic forgeries have not been considered, but we tried to compare the proposed method with some related ones. Table 2 shows the comparison.

There exist three kinds of handwriting forgeries, mimic forgeries, freehand unskilled forgeries and freehand skilled forgeries. The most troublesome of all is the mimic forgeries, because it is so similar to the original handwriting and unlike skilled forgeries, everybody can perform it.

Using the micro features, we were able to detect mimic forgeries from off-line data, and according to the experimental results, the words with longer strokes are more recognizable by the proposed system.

Table 1. Classification results of the system.

Word	Correctly Verified Ratio (TP)	Unskilled Forgeries Correctly Detected	Skilled Forgeries Correctly Detected	Mimic Forgeries Correctly Detected
One	80.67%	68.00%	0.00%	77.00%
Two	95.33%	72.00%	0.00%	22.00%
Three	94.67%	65.33%	73.33%	100.00%
Four	64.67%	100.00%	64.00%	49.00%
Five	68.67%	68.00%	13.33%	97.00%
Six	83.33%	73.33%	37.33%	95.00%
Seven	68.00%	69.33%	0.00%	28.00%
Eight	51.33%	62.67%	0.00%	54.00%
Nine	99.33%	29.33%	0.00%	53.00%
Thousand	80.00%	66.67%	45.33%	100.00%
Million	69.33%	100.00%	36.00%	79.00%
Billion	82.67%	2.67%	0.00%	62.00%

The best overall result of the proposed system was 94.67% in correct acceptance and 81.60% in correct rejection, which its partial rejection result was 65.33% in unskilled forgeries, 73.33% in skilled ones and 100% in mimic forgeries.

Table 2. comparison of different writer verification methods (H: Handwriting, S: Signature)

Ref.	Random		Unskilled		Skilled		Mimic		Remarks
	FAR	FRR	FAR	FRR	FAR	FRR	FAR	FRR	
[9]	-	-	18%	0%	-	-	-	-	On/H, 23 writers
[2]	8%	7%	27%	28%	-	-	-	-	Off/S, 5 writers
[10]	0.8%	3.4%	2%	8%	-	-	-	-	Off/H, 100 writers, 20 random, 20 unskilled
[11]	-	-	2%	2%	35%	2%	-	-	Off/S, 100 writers, Minimum FRR
[11]	-	-	0%	25%	20%	25%	-	-	Off/S, 100 writers, Minimum FAR
[this]	-	-	0%	30%	26.6%	5.3%	0%	5.3%	Off/H, using only one word (minimum FAR)
[this]	-	-	0%	.6%	26.6%	.6%	0%	.6%	Off/H, best case of having all word

In future, aim is refining the estimated stroke and trying to make the system not this much dependant to the text and its language.

REFERENCES

- [1] K.P. Zimmermann and M.J. Varady, "Handwriter identification from one-bit quantized pressure patterns," *Pattern Recognition*, vol. 18, 1985, pp. 63-72.
- [2] I.S.I. Abuhaiba, "Offline Signature Verification Using Graph Matching," *Turk J Elec Engin*, vol. 15, 2007, pp. 89-104.
- [3] V. Marti and H. Bunke, "The IAM-database: an English sentence database for offline handwriting recognition," *Journal on Document Analysis and Recognition*, vol. 5, 2002, pp. 39-46.
- [4] B. Helli and M.E. Moghaddam, "A Text-Independent Persian Writer Identification System Using LCS Based Classifier," *2008 IEEE International Symposium on Signal Processing and Information Technology*, IEEE, 2008.
- [5] B. Helli and M.E. Moghaddam, "Persian writer identification using extended Gabor filter," *Image Analysis and Recognition*, 2008, pp. 579-586.
- [6] B. Helli and M.E. Moghaddam, "A writer identification method based on XGabor and LCS," *IEICE Electronics Express*, vol. 6, 2009, pp. 623-629.
- [7] B. Helli and M.E. Moghaddam, "A text-independent Persian writer identification based on feature relation graph (FRG)," *Pattern Recognition*, vol. 43, Jun. 2010, pp. 2199-2209.
- [8] V. Pervouchine, G. Leedham, and K. Melikhov, "Handwritten character skeletonisation for forensic document analysis," *Proceedings of the 2005 ACM symposium on Applied computing*, ACM, 2005, p. 754-758.
- [9] A. Namboodiri and S. Gupta, "Repudiation Detection in Handwritten Documents," *Proceedings of the International Conference of Biometrics*, 2007, pp. 356-365.
- [10] A. Schlapbach and H. Bunke, "A writer identification and verification system using HMM based recognizers," *Pattern Analysis & Applications*, vol. 10, 2007, p. 33-43.
- [11] A. Piyush Shanker and A. Rajagopalan, "Off-line signature verification using DTW," *Pattern Recognition Letters*, vol. 28, 2007, p. 1407-1414.

PARTITION ITERATED FUNCTION RADIOGRAPHS IMAGE ENHANCEMENT FOR FISH BONE DETECTION

Noor Elaiza Abdul Khalid, Saadiah Yahya and Hidayah Shariff
University Technology MARA Malaysia - 40450 Shah Alam, Selangor

ABSTRACT

This paper investigate the effectiveness of Partition Iterated function(PIFs) in detecting suspected fish bone ingestion in the soft tissue lateral neck radiograph. Embedded fish bone lodge in the throat is not easily visible in plain radiograph. Serious complication may cause perforation of the lodged and inflammation that can progress to abscess. Forty X-ray images of 23 male and 17 female patients between the ages of 6 to 72 years are taken as the data sample. The quality of the images were assessed and evaluated during pre and post processing by the radiologist. The images are classifies into five; 1. fish bone(FB) detected in original and processes; 2. FB not detected in original but detected in processed images ; 3. FB detected in original but found as false-positive in PIFS; 4. Fish bone detected on PIFS but not detected on PAC and 5. FB not detected but abnormality detected. The result has proven PIFs as promising contrast enhancement method for detecting fish bone in lateral neck soft tissue radiographs.

KEYWORDS

Image Enhancement, Partition Iterated Function, Foreign body, Fish bone, Radiographs.

1. INTRODUCTION

Fish bone (FB) accounts to sixty percent of foreign body accidentally ingested. (Goh et al.,2006: 710-714) Usually the FB passes through the intestinal tract uneventfully but in some cases it lodges in the esophagus and poses variety complications. (Chiu, Li and Chen, 2006: 298-301) Patients may or may not be aware of this ingestion and does not present specific symptoms. The frontline protocol investigation for patients suspected of FB accidental ingestion is the evaluation of lateral neck radiographs. (A.Chawla, Eng and Peh ,2004:397-403). Akazawa et al (2004: 413-416) in their studies indicate a lower fish bone ingestion detection rate in radiograph images as compared to computed tomography (CT). Nevertheless CT is significantly more expensive and less available compared to radiograph machines. Radiographs images, however are noisy, low in contrast and the anatomy of the neck makes detecting the tiny FB very difficult to detect (Bathla et al, 2011: 63-71; Wei et al, 2007: 702 - 705). Consequently, many researchers have accepted the challenge of enhancing radiographs images to improve diagnostic interpretability by deploying noise reduction filter and enhancing contrast (Ahmad et al,2011:528-533; Wei et al, 2007: 702 – 705; Juan and Antonio, 2009:7786–7797; Guodong et al, 2009:140–147)

Median filter due to its smoothing and edge preserving nature is one of the most popularly used noise reduction filter (Lin , 2007: 1073–1087 ; Juan and Antonio, 2009:7786–7797; Martin et al. 2007:291–308). Its popularity has sparked other researches to improve its performance, thus generates the modified median filters (Juan and Antonio, 2009:7786–7797; Wei et al, 2007: 702 – 705). Many successful noise reduction filters such as homomorphic filtering methods (Salem et al, 2010: 139-143), Non-linear anisotropic (Phan et al, 2009: 101–112), Gaussian filter (Guodong et al, 2009:140–147) are ascertained. Although, the deployment of these filters have considerably help reduced noise, radiograph images are low in contrast and are mostly limited as the first line of diagnosis. Small or low radiolucence abnormalities are often missed diagnosed. Therefore, this inspires the deployment of contrast enhancement.

Contrast is important in determining the quality of radiographs images (Bushberg et al, 2001:255). Contrast enhancement filters can be performed as global and local methods (Kim, Kim, and Hwang, 2001:475-484). The global methods considers the statistics of the whole image which may not work well in large ranged gray level images . The local method on the other hand, improve contrast based on local statistic

whose gray level range are usually low (Khalid et al, 2007:702-708) Over the years many contrast enhancement filters have been introduced. Histogram equalization a global method is the most popular contrast enhancement algorithm due to its effectiveness and simplicity (Salem et al, 2010: 139-143) which performs well in low gray level ranged images. Modified local method of histogram equalization includes Contrast Limited Adaptive Histogram Equalization (CLAHE) and adaptive histogram equalization performs well in applications such as dental radiographs (Ahmad et al, 2011:528-533), and neck histogram (Khalid et al, 2010:163 – 167) respectively. These attempt maybe successful but there are still rooms for improvement. Economopoulos et al. (2010:45–54) have proposed Partitioned Iterated Function System (PIFS) method which uses the local processing method which incorporates both contrast enhancement and noise reduction. This paper proposes to investigate the performance of PIFS in enhancing and detecting fish bone PIFS from lateral neck radiographs.

2. RESEARCH DESIGN AND DEVELOPMENT

The study initially involves acquiring 40 radiographic images which are taken in the area of esophagus. The study is divided into two phases. The first phase consists of designing and implementation of the PIF system and the second phase involves in classifying the images into three different classes. The following sections discuss PIFS algorithm and the classification processes in detail.

2.1 Partition Iterated Function System

PIFS consists of five steps namely: Partitioning the images into range and domain blocks; shearing the domain block into sub-samples; calculating the Euclidean distances; searching for the fittest domain blocks from the domain pool by matching the range blocks with the subsample domain blocks; and transforming the blocks into global contractive. Each of these steps is explained in the subsequent section. PIFS algorithm begins with partitioning of the image into two basic block unit which are range block and domain block. The range blocks consists of a set of non-overlapping block with a fixed size of $w_x \times w_y$ where as the domain blocks are a set of overlapping image block with $2w_x \times 2w_y$. This study uses non-overlapping range blocks of size 4×4 which are expressed as in Equation $R_{ij} = [x_i, x_i+w_x] \times [y_j, y_j+w_y]$, where i = number of range block in vertical line, j = number of range block in horizontal line, x_i = first position value of x of every range block and y_j = first position value of y of every range block as equation $D_{kl} = [u_k, u_k+2w_x] \times [v_l, v_l+2w_y]$, where k = number of domain block in vertical line, l = number of domain block in horizontal line, u_k = first position value of x of every domain block and v_l = first position value of y of every domain block. The second phase is shearing the domain block where the size of domain block images are reduced from 8×8 to 4×4 sub-samples. Equation $I_d(x, y) = \frac{1}{4} [I_d(x, y) + I_d(x + 1, y) + I_d(x, y + 1) + I_d(x + 1, y + 1)]$ represents shearing process. The next step is finding the Euclidean distance between sub-sampled domain blocks with range block as shown in Equation $E(k, l; i, j) = \|\gamma (d_{R,l} - \mu_{d(k,l)}) - (\eta_j - \mu_{r(i,j)})\|^2$, where $\mu_{r(i)} = \frac{\sum_{j=1}^4 r_{ij}}{16}$ and $\mu_{d(k,l)} = \frac{\sum_{i=1}^4 d_{k,l}}{16}$. This Euclidean distance is used as a matching parameter for matching the sub-sampled domain blocks with the range blocks. The matching process involves a computationally intensive exhaustive search algorithm. Parameters such as location and mean of the sub-sampled blocks are successively used in the global contractive transformation (W) process as Equation $W(I(x, y)) = \sum_{i,j} [\gamma (I((2(x - x_i) + u_{k(i,j)}), (2(y - y_j) + v_{l(i,j)})) - \mu_{D(k,l; i(j))}) + \mu_{R(i,j)}] M_{ij}(x, y)$, where $\mu_{D(k,l; i(j))}$ represents mean domain block corresponding the best matched with range block, $\mu_{R(i,j)}$ represents mean range block, $u_{k(i,j)}$ represents the first position x of domain block best matched with range block, $v_{l(i,j)}$ represents First position y of domain block best matched with range block, $M_{ij}(x, y) = 1$, $(x, y) \in R_{ij}$, $M_{ij}(x, y) = 0$, $(x, y) \notin R_{ij}$ and $(k(i, j), l(i, j)) = \arg \min\{E(k, l; i, j)\}$. Global contractive transformation consists of *coding* and *decoding* processes. Firstly, the image is coded using equation (5) with relative high value for the parameter γ . The *coding* process is done by iterating the set of transformations on an arbitrary initial image and the image partition is used to determine the range in the image. If the distance is less than a given tolerance, then we save the number of the domain and the data set for the contractively transformation. Next, the decoded image is created by substituting equation (5) with lower value for γ . The *decoding* codes

are used to calculate the interpolation functions from the information of the encoded image and to compute the transformations out of them. After encoding the images, there are some senses of correctness and it lies on an underlying but unknown curve, decoded algorithm to be able to estimate the values of the curve at any position between the known points within the range of a discrete set of known data points. The resulting image is the low pass version, I_{LP} of the original image. The enhanced image, I_{enh} , is finally obtained using the following equation $I_{enh}(x, y) = I(x, y) + \lambda I_{HP}(x, y)$, where $I_{HP}(x, y) = I(x, y) - I_{LP}(x, y)$

2.2 Qualitative Study

In the qualitative study, the Picture Archiving Communication system (PACS) and PIFS-enhanced images are assessed by two radiologists and classified according to Table 1. The two sets of images were shown separately to avoid recall bias. In each session, only one set of image either PACS or PIFS-enhanced image was shown to the radiologist in random order. During the classification, radiologist was allowed to manipulate the original images using window-level settings in the PACS with no time constraint. Fish bone detected on both sets of images, and show enhancement in PIFS. Fish bone detected on PIFS-enhanced image, but not detected on PACS. Fish bone detected on PACS, but confirmed in AHE-enhanced image as false-positive finding. No fish bone detected, but abnormality detected, and show enhancement in PIFS. No fish bone and abnormality detected on both sets of images.

Table 1. Classification of Soft Tissue Lateral Neck Radiograph

Category	Descriptions
1	Fish bone detected on both sets of images, and show enhancement in PIFS.
2	Fish bone detected on PIFS-enhanced image, but not detected on PACS.
3	Fish bone detected on PACS, but confirmed in AHE-enhanced image as false-positive finding.
4	No fish bone detected, but abnormality detected, and show enhancement in PIFS.
5	No fish bone and abnormality detected on both sets of images.

3. RESULTS

The radiologist found discrepancies in the diagnosis of eight patients out of the forty patients lateral neck images. From the forty images processed, two are classified as class 1, one as class 2 and two as class 5. The rest are found to be normal. Figure 1 and 2 were classified by the radiologists as Class 1. The fish bones were barely visible originally but with PIFS, the fish bones were much clearer. The PIFS confirmed the site and size of the fish bone which hardly seen on the PACS images.



Figure 1a. Fish bone detected - 1 cm,



1b. Fish bone detected and enhanced -1 cm

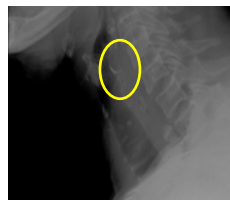


Figure 2a. Fish bone detected -2 cm



2b. Fish bone detected and enhanced -2 cm

Figure 3 was classified by the radiologists as Class 2. Although the fish bone is visible in PACS image, the PIFS has successfully enhanced the fish bone location and geometry.

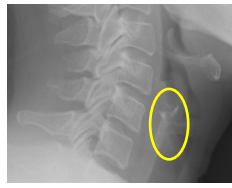


Figure 3a. Fish bone not detected,

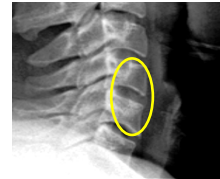


3b. Fish bone detected and enhanced -1 cm

Figure 4 and Figure 5 was classified by the radiologists as Class 5. Although in the original image, it seemed to be a fish bone but the PIFS has caused the suspected fish bone is false positive.



Figure 4a. Misdiagnose Fish bone detected,



4b. confirmed as the continuity of cartilage calcification



Figure 5a. Abnormality detected (air seen).



5b. Abnormality detected and enhanced.

All figures (fig 1-2) show the fish bone is clearly seen in the PIFS-enhanced image. The PIFS-enhanced images show the surface of the fish bone the location and the geometry of the fish bone. PIFS not only enhance the visualization of fish bone but also confirm the negative finding in the PACS image as in figure 4. Figures 1-5 has proven that PIFS performs well in detecting and enhancing fish bone images. The PIFS manage to enhanced images to show the location and the geometry of the fish bone. PIFS not only enhance the visualization of fish bone but also confirms the negative findings in the PACS image as in fig 4. Whether the fish is easily visualized or barely visible, the PIFS has proven that the fish bone can be enhanced and can be clearly identified. The range of fish bone size detected was from 0.5 cm to 2.0 cm. This size of fish bone is easily overlooked during eating which increases the likelihood of ingesting the fish bone.

4. CONCLUSION

In this research we have successfully proven that the enhancement of radiographs images can provide a non-invasive and efficient tool in diagnosing the presence of fish bone in the upper neck areas. PIFS have proven to be useful in the enhancement of low contrast soft tissue lateral neck radiographic images. The findings in this research can be divided into three cases. The first case shows a similar result observed for both the PACS and the enhance images. The second cases shows PIFS enhanced image can provide better diagnosis and finally the third case will ensure the detection of false positive cases that may lead to misdiagnosis. This enhancement method is quite sufficient in diagnosing the fish bone quickly. These preliminary results will be continue to specify various types of abnormalities that can be detected as a first line of diagnosis.

ACKNOWLEDGEMENT

We would like to thank the Hospital Sungai Buloh and Pusat Perubatan Universiti Kebangsaan Malaysia for their generosity in providing the images. We also wished to thank all the doctors and radiologist from Pusat

Perubatan Universiti Kebangsaan Malaysia involved in the evaluation of the PACS and PIFS-enhanced images.

REFERENCES

- Abdul Khalid N. E., Manaf M., Aziz M. E., Ali M. H., ‘CR images of Metacarpel Cortical Edge Detection-Bone Profile Histogram Approximation Method”, *Intelligent and Advanced Systems*, 2007, 702-708
- Ahmad S.A., Taib M.N., Khalid N.E.A, Ahmad R.and Taib H., 2011. “Performance of Compound Enhancement Algorithms on Dental Radiograph Images”, *World Academy of Science Engineering and Technology*, Issue 74,pp: 528:533
- Akazawa Y., Watanabe S., Nobukiyo S., Iwatake H., Seki Y., Umehara T., Tsutsumi K., Koizuka I. 2004. “ The management of possible fishbone ingestion”, *Auris Nasus Larynx*, Vol 31, pp 413-416.
- Bathla G, Teo LL, Dhanda S., 2011. Pictorial essay: Complications of a swallowed fish bone. *Indian J Radiol Imaging*. No. 21, pp 63-71
- Bushberg J. T., Seibert J. A., Leidholdt E. M. Jr., and Boone J.M., *The essential physics of medical imaging*. Second edition. 2002, pp 255
- Chawla, Eng S.P., Peh W.C.G. 2004. Clinics in diagnostic imaging. *Journal of Singapore Medical*, Vol 45, pp 397-403.
- Chiu Hsin-Hui, Joe Hergens Li, and Jin-Sheng Chen, 2006. *Penetration of Esophageal Wall by A Fish Bone*. Division of Gastroenterology, Department of Internal Medicine, Kuo General Hospital, Tainan, Taiwan, ROC
- Economopoulos. T.L, Asvestas. P.A., Matsopoulos. G.K. 2010. Contrast enhancement of images using Partitioned Iterated Function Systems. *Image and Vision Computing* 28, pp 45–54.
- Goh Yu-Meng Tan Brian K. P., Shueh-En Lin, Pierce K. H. Chow, Foong-Koon Cheah, London L. P. J. Ooi, Wai-Keong Wong. 2006. CT in the Preoperative Diagnosis of Fish Bone Perforation of the Gastrointestinal Tract. Gastrointestinal Imaging and Clinical Observations. *American Roentgen Ray Society*. No. 187. pp 710-714
- Guodong Wang, Nong Sang, Luxin Yan, Xubang Shen (2009). X-ray angiogram images enhancement by facet-based adaptive anisotropic diffusion. *Computerized Medical Imaging and Graphics* 33 (2009) 140–147
- Jain A. K., 1989. *Fundamentals of Digital Image Processing*. Prentice-Hall.
- Juan L. Mateo, Antonio Fernández-Caballero. 2009. Finding out general tendencies in speckle noise reduction in ultrasound images. *Expert Systems with Applications* 36 (2009) 7786–7797
- Kim Joung-Youn, Kim Lee-Sup, and Hwang Seung-Ho. 2001. An Advanced Contrast Enhancement Using Partially Overlapped Sub-Block Histogram Equalization, *IEEE Transactions On Circuits And Systems For Video Technology*, Vol. 11, No. 4, April,475-484
- Lin Tzu-Chao, 2007. A New Adaptive Center Weighted Median Filter For Suppressing Impulsive Noise In Images. *Information Sciences*. No.177, pp 1073–1087
- Martin Welka, Joachim Weickerta, Florian Beckerb, Christoph Schno, Christian Fedderna, Bernhard Burgeth. 2007. Median and related local filters for tensor-valued images. *Signal Processing*. No. 87, 291–308
- Ngan J.H., Fok P.J., Lai E.C., Branicki F.J., Wong J. A., 1989. Prospective study on fish bone ingestion. Experience of 358 patients. *Ann Surg*. No.211, pp. 459-62.
- Noor N.M., Khalid N.E.A., Ali M.H., Numpang A.D.A. 2010. Fish Bone Impaction Using Adaptive Histogram Equalization (PIFS), *International Conference On Computer Research And Development*. pp.163 – 167
- Phan T.H. Truc, Md. A.U. Khan, Young-Koo Lee, Sungyoung Lee , Tae-Seong Kim. 2009. Vessel enhancement filter using directional filter bank. *Computer Vision and Image Understanding*. No.113. pp 101–112
- Salem Saleh Al-amri1 Mr., Kalyankar N.V., Khamitkar S.D. 2010. Linear and Non-linear Contrast Enhancement Image. *IJCSNS International Journal of Computer Science and Network Security*, Vol.10. No.2, pp 139-143
- Wei Zheng, Yang Hua, Sun Hui-sheng, Fan Hong-qi (2007). *X-ray Image Enhancement Based on Multiscale Morphology*. Science and Technology Research and Development Program of Hebei Province 702 - 705

EVALUATION OF TWO RECONSTRUCTION SYSTEMS BASED ON STEREOVISION

Amira Soudani and Ezzeddine Zagrouba
*Laboratoire RIADI, Université de La Manouba,
Equipe de recherche SIIVA, Institut Supérieur d'Informatique de Tunis, Tunisie*

ABSTRACT

This paper presents an evaluation and comparison of two 3D reconstruction approaches. The first one is an existing sparse reconstruction which is based on the triangulation of extracted points of interest using calibration parameters. The results obtained by this approach present the outline of the object. The second one is improved dense reconstruction approach based on a generated disparity map. For this approach, we improve the stereo matching algorithm. In fact, the initial disparity map is computed using a weighted dissimilarity measure with application of adaptive correlation windows. Then, planes are modeled by a set of planar surface patches. First, an initial set of planes is estimated using only reliable pixels. Then, those planes will be merged to obtain a final representation of the scene. Thus, each pixel and region will be assigned to a disparity layer according to a cost function which is subject of minimization. The experimentations and evaluations show that the second approach gives efficient results concerning the recovered informations.

KEYWORDS

Computer vision, Stereovision, Calibration, Matching, Rectification, 3D Reconstruction.

1. INTRODUCTION

The classical 3D reconstruction process is generally composed by four major steps which are camera calibration, image rectification followed by primitive matching and finally 3D reconstruction.

The camera calibration is an offline procedure that aims to find intrinsic parameters as well as the extrinsic parameters. Camera calibration approaches can be divided into three major categories. The first one includes the classical methods (Zhang 1998), the photogrammetric methods (Lavest 1999) and self-calibration methods (Sturm 2001).

The epipolar rectification is the process of warping the stereo images in order to align horizontally the epipolar lines. Rectification methods can be classified into methods based on the calibration parameters (Fussiello et al. 2000) and methods based on the fundamental matrix (Fussiello et al. 2008).

The third step is the primitive matching. Once primitives (points, segments or regions) are extracted from the pair of images, we'll fetch those that match to each other. Matching methods can be classified into local methods that perform minimization of local costs (Aschwanden 1992, Zagrouba et al. 1992) and global methods which perform a minimization of a global cost (Sadeghi et al. 2008, Bleyer et al. 2007).

The last step of the stereo vision process is the 3D reconstruction (Zagrouba 1997). It is defined as is the process of finding the 3D coordinates of a point in world reference. Depending on the availability and accuracy of the calibration data, there are three possible degrees of 3D reconstruction: Euclidean reconstruction, affine reconstruction (Trucco 1998) and projective reconstruction (Sturm 1996).

This paper introduces two 3D reconstruction prototypes. The first one is an existing Euclidean reconstruction prototype which provides us a set of sparse 3D points that present the outlines of the reconstructed object. The second proposed prototype is a dense reconstruction prototype based on a generated disparity map improved using a weighted dissimilarity measure and adaptive correlation windows. Thereby, planes are modeled by a set of planar surface patches.

The remainder of this paper is organized as follows: section 2 presents the first reconstruction prototype while section 3 explains the proposed dense reconstruction. Then, the experimental results are illustrated in

section 4. Finally, conclusion and perspectives are drawn in section 5.

2. FIRST PROTOTYPE: 3D SPARSE RECONSTRUCTION

The classical 3D reconstruction system is composed of several successive steps.

2.1 Stereo Head Calibration and Rectification

We calibrate each camera independently, then the transformation between the two cameras will be determined using a set of captured pattern images. The used calibration method is based on Zhang algorithm (Zhang 1998). Thus, we obtain intrinsic and extrinsic parameters associated to the left and right cameras. In order to simplify the procedure of stereo matching, stereo images are first rectified. The method used is based on the projection matrix issued from the calibration step (Fusiello et al. 2000).

2.2 Primitive Matching and 3D Reconstruction

Once left and right images are rectified, primitives shall be extracted from them. The SIFT detector (Lowe 2004) is used. Then, the detected points of interest are matched based on the ZNCC score (Zero mean Normalized Cross Correlation) and we'll proceed to a 3D triangulation. It is based on the linear relation that exists between the homogenous coordinates of a point X^w in the world reference and its projections X^{P1} , X^{P2} in the two camera planes expressed as below:

$$X^{P1} = PX^w ; X^{P2} = QX^w \quad (1)$$

Where P and Q are the projection matrix associated to the left and right cameras.

Once the coordinates of X are deduced, we obtain set of sparse 3D points representing the outlines of the reconstructed object. This does not provide us enough information about the scene so the 3D structure should be improved by a dense estimation of the depth.

3. SECOND PROTOTYPE: DENSE RECONSTRUCTION

In this section we'll present the dense reconstruction process. This process is composed by three steps. The calibration and rectification steps have been already presented in section 2.

3.1 Dense Stereo Matching and 3D Modeling

In this step, the rectified stereo images will be used to achieve a dense stereo matching.

3.1.1 Initial Disparity Map

An initial disparity map is determined using a local matching approach based on correlation. The used dissimilarity measure combines the Sum of Absolute Intensity Differences (SAD) and a gradient based measure (GRAD). Thus, the dissimilarity associated to a pixel $p(x, y)$ at the disparity d is defined as below:

$$Diss_{tot}(x, y, d) = w Diss_{SAD}(x, y, d) + (1-w) Diss_{GRAD}(x, y, d) \quad (2)$$

Where $Diss_{SAD}(x, y, d) = \sum_{(i,j) \in N(x,y)} |I_1(i, j) - I_2(i + d, j)|$ and

$Diss_{GRAD}(x, y, d) = \sum_{(i,j) \in N_k(x,y)} |\nabla_k I_1(i, j) - \nabla_k I_2(i + d, j)|$ with w is a user-defined penalty, $N(x, y)$ is a surrounding window at the point $p(x, y)$ and ∇_k is the gradient vector with k expresses the horizontal and vertical component.

This dissimilarity is estimated using a winner-take-all optimization in conjunction with a cross-checking test. To further improve the resulted disparity map, adaptive windows were used.

3.1.2 Disparity Planes Extraction and Grouping

We first decompose the reference image into regions of homogeneous. The algorithm assumes that disparity

values vary smoothly in homogenous regions so an over-segmentation of the image is preferred. Then, we try to determine an initial disparity map in order to extract an initial representation of the scene which is modeled by a set of planar disparity planes specified by a planar equation as below:

$$d(x, y) = ax + by + c \quad (3)$$

Where $d(x, y)$ is a disparity associated to pixel of coordinates (x, y) and a, b, c are the plane's parameters.

A robust version of the method of least squares was applied on all valid points within a region issued from initial disparity map in order to deduce those parameters. Then, extracted planes will be grouped in order to have a more accurate representation of the scene using the "Mean-shift" clustering algorithm.

3.1.3 Pixels and Regions Assignment

The stereo matching problem is formulated in terms of energy minimization. Then our goal is to find the labeling function denoted " D_{layer} " that assigns regions and pixels to a corresponding disparity layer and which minimize the energy expressed in Equation 4.

$$E_{tot} = E_{int} + E_{ext} \quad (4)$$

Internal energy E_{int} combines two terms. The first is E_{reg} that enforces the assumption of smoothly varying disparity inside a region (Equation 5) where $Reg(p)$ is a function that determines the region to which a pixel belongs and α_{MAX} is a positive infinite number. While the second term is E_{smooth} which penalizes neighboring regions assigned to different disparity layers (Equation 6).

$$E_{reg} = \alpha_{MAX} \cdot Cardinal \{p \in I_{left} / Label(p) \notin \{Label(Reg(p)), 0\}\} \quad (5)$$

$$E_{smooth} = \alpha_{disc} \cdot \sum_{(R_i, R_j) \in NB \text{ and } Label(R_i) \neq D_{layer}(R_j)} CF(R_i, R_j) \cdot S(R_i, R_j) \quad (6)$$

Where α_{disc} a user-defined penalty, NB is the set of all couples of neighboring regions, $CF(R_i, R_j)$ determines the length of the common frontier and $S(R_i, R_j)$ expresses the similarity between two regions R_i and R_j .

The term of external energy E_{ext} combines E_{diss} that expresses the radiometric constraint which supposes that the intensities of a pixel and its homologous according to a given disparity layer assignment are the same. This term includes an occlusion penalty that prevents the maximizing of the number of pixels assigned to the occluded disparity layer (Equation 7). The second term is E_{mis} , it enforces the hypothesis of belonging of each pixel and its homologous to the same disparity layer (Equation 8).

$$E_{diss} = \sum_{p \in I} MAX(Diss(p, Match[Label(p)](p)), \alpha_{occ}) \quad (7)$$

$$E_{mis} = \alpha_{mis} \cdot Cardinal \{p \in I_{left} / Label(p) \notin \{Label(Match[Label(p)]), 0\}\} \quad (8)$$

Where $Diss(p_i, p_j)$ is a function that computes the color dissimilarity between two pixels p_i, p_j and $Match[Label(p)]$ is a function that determines the homologous pixel of a pixel p by the labeling $Label(p)$ while α_{occ} and α_{mis} are user-defined penalties.

To minimize the cost function, we first determine the initial labeling functions associated to regions and pixels. We merge regions that belong to the same disparity layer. Thus, the initial labeling associated to those regions is deduced from extracted planes. Then, we assign the disparity value of a region to pixels that belong to it. We use this assignment as an initial combination of disparity for pixels and regions level. Later, we proceed to a minimization of the cost function. We generate a set of combinations using the initial combinations and we choose the one that minimize more the cost function.

3.2 3D Model Rendering

The depth map is computed by triangulation. The depth map is then converted in a triangular mesh using Delaunay triangulation. The 3D representation is then visualized with a more realistic appearance by applying a texture on the obtained structure.

4. EXPERIMENTAL RESULTS

We evaluate the two reconstruction prototypes cited above on different pairs of images. First, we compute calibration parameters which are shown Table1. To evaluate the results obtained by this step, we compute the reprojection error associated to some pairs of pattern images. Table2 shows this error which is relatively low (between 0.2 and 0.5 pixel).

Table 1. Calibration parameters

	Intrinsic parameters		Extrinsic parameters	
	Focal length	Principal Point	Translation	Rotation
Left camera	[540.626 542.391]	[187.118 142.491]	[-221.634 -32.688 22.067]	[0.015 -0.029 0.063]
Right camera	[541.087 548.074]	[185.281 146.082]		

Table 2. Reprojection errors

	5th left Image	9th left image	11th left Image
Error (pixel)	[0.51280 0.47880]	[0.415321 0.427020]	[0.28180 0.20410]

We first rectify the self-captured pair of images (Figure 1.b and Figure 1.c). Then, SIFT primitives are extracted and matched. Thereby, those pairs are triangulated which provide us a set of 3D points (Figure 1.d) representing the outlines of the reconstructed object.

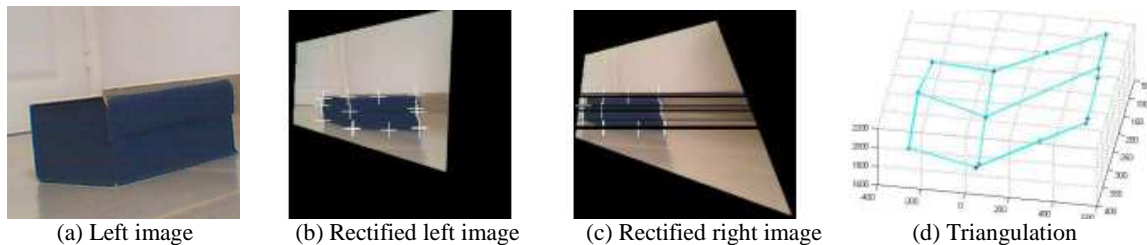


Figure 1. Sparse reconstruction

We present in Figure 2 and Figure 3 the results provided by applying this process on two pairs of images. Figure 2 shows results for a pair of images taken by our stereo head. Since we do not have the ground truth for this image pair, we show the 3D reconstruction of the scene to demonstrate the good quality of the obtained result. For quantitative evaluation of the stereo matching algorithm, we compute the percentage of unconcluded pixels whose disparity's error is greater than 2 for the second pair of images that we have the ground truth. For the computed disparity map in Figure 3.b, we obtained 11,41% of wrong pixels.

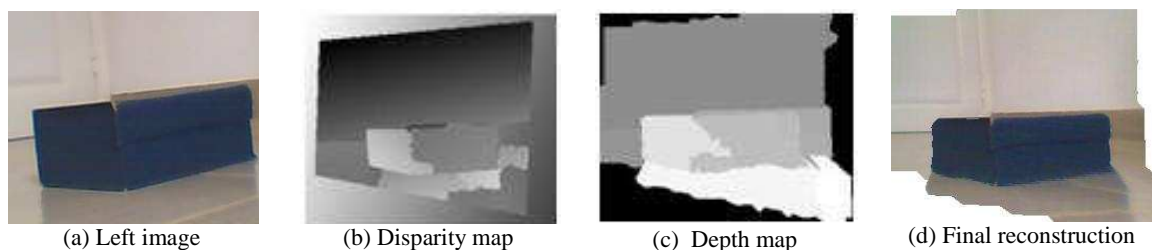


Figure 2. Dense reconstruction of a self-captured pair of images

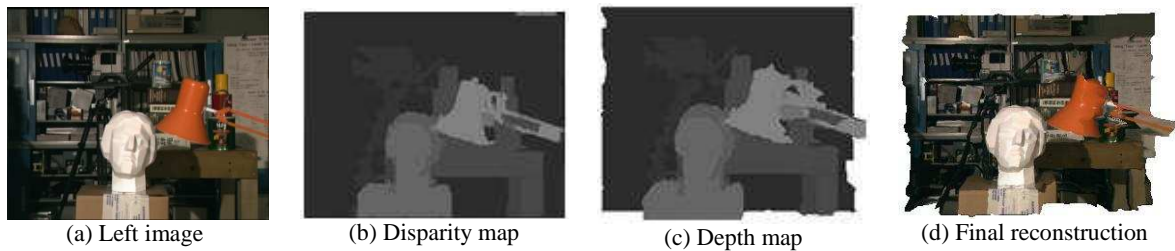


Figure 3. Dense reconstruction of a given pair of images (Tsukuba)

5. CONCLUSION

Two prototypes of 3D reconstruction have been presented in this paper. The first one is a sparse reconstruction which provides us the outlines of the reconstructed object. This does not provide us enough information about the scene. Thus, the 3D structure was improved by a dense estimation of depth using the dense reconstruction prototype. An initial disparity map is computed using a weighted dissimilarity measure. Then, planes are modeled by a set of planar surface patches that will be merged to obtain a final representation of the scene. Thereby, each pixel and region will be assigned to disparity layer by minimizing a cost function. A Delaunay triangulation followed by a texture mapping step are then applied. The experiments show the results obtained and the improvements provided by the dense reconstruction prototype. As perspective to this work, we will try to generate 3D models using surface modeling that takes into account discontinuities and occlusions.

REFERENCES

- Aschwandten, P. and Guggenbul, W. (1992). Experimental results from a comparative study on correlation type registration algorithms. *Robust computer vision: Quality of Vision Algorithms*, pages 268-282.
- Bleyer, M. and Gelautz, M. (2007). Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions. *Signal Processing: Image Communication* 22, pages 127-143.
- Fusiello, A. and Irsara, L. (2008). Quasi-euclidean uncalibrated epipolar rectification. *19th International Conference on Pattern Recognition (ICPR2008)*.
- Fusiello, A., Trucco, E., and Verri, A. (2000). A compact algorithm for rectification of stereo pairs. *Machine Vision and Applications*, 12(1) : 16-22.
- Lavest, J., Viala, M., and Dhome, M. (1999). Quelle précision pour une mire d'étalonnage. *Traitement du Signal*.
- Lowe, D. (2004). Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60: 91-110.
- Sadeghi, H., Moallem, P., and Monadjemi, S. (2008). Feature based dense stereo matching using dynamic programming and color. *International Journal of Computational Intelligence*, 4(3) :179-186.
- Sturm, P. (2001). On focal length calibration from two views. *IEEE Conference on Computer Vision and Pattern Recognition*, pages 145-150.
- Sturm, P. and Triggs, B. (1996). A factorization based algorithm for multi-image projective structure and motion. *Proceedings of the 4th European Conference on Computer Vision*, 1065 :709-720.
- Triggs, B., Lauchlan, P. M., Hartley, R., and Fitzgibbon, A. (1999). Bundle adjustment-a modern synthesis. *Proceedings of the International Workshop on Vision Algorithms (ICCV 99)*, pages 298-372.
- Zagrouba, E. (1997). 3-D Facets Constructor for Stereovision. *Advances in Artificial Intelligence, AI*IA 97, Springer*, pages 116-126.
- Zagrouba, E. and Krey, C. (1992). Region matching by adjacency propagation in stereovision. *Proceedings 2nd ICARCV'92*.
- Zhang, Z. (1998). Understanding the relationship between the optimization criteria in two-view motion analysis. *IEEE International Conference on Computer Vision (ICCV98)*, pages 231-236.

EYE CENTRE LOCATION WITH HOUGH TRANSFORM*

Ivan A. Matveev

*Dorodnitsyn Computing Centre of Russian Academy of Sciences
Vavilov str., 40, Moscow, 119333, Russia*

ABSTRACT

An algorithm of locating approximate eye center in image is presented. It is based on Hough approach and exploits the fact that brightness anti-gradient vectors at the circular borders of pupil and iris are all targeted to the circle's centers. Pixels with high brightness gradient value are selected in image as possible border points. In each of these pixels voting to Hough accumulator is performed with a segment in the direction of brightness anti-gradient. Position of maxima of the accumulator is considered as an eye center. Tests were performed for more than 95000 images of public domain iris databases.

KEYWORDS

Iris segmentation, image processing, Hough transform.

1. INTRODUCTION

One of main biometric identification technologies is iris image recognition. As in any recognition task significant problem here is selecting region of interest (i.e. iris area) in the image. Substantial optimization of this process can be achieved by using a method for approximate location of eye center in the image under condition that such method takes low amount of calculations and introduces few additional errors. Minimizing calculations is particularly important in case when iris segmentation methods are applied to process video stream input from camera in real time in order to fetch iris images appropriate for further processing and generating templates for recognition.

One of the requirements to iris images used for recognition is that iris is registered in frontal view (eye is looking directly into the camera). Under this condition both pupil and iris have shape close to circular, unless occluded by eyelashes and eyelids. These circles are approximately concentric, pupil is darker than surrounding iris, and iris, in its turn, is darker than surrounding sclera. A simple algorithm detecting eye center as a center of an object selected by thresholding function is proposed in [7]. With an additional condition that there is no specular reflection in pupil, detection of center as a point having maximum distance from substantial brightness variations was proposed in [6]. However in the majority of images partial occlusions of the iris by eyelashes and eyelids is present, and in many images pupil is also occluded. Apart, in most iris registration systems specular reflection from illuminator is present inside the pupil. Image interchange standard [3] recommends not to use images with occluded pupil for recognition. However, during the work of iris registration system (at the stage of positioning and waiting for good eye image to appear in video stream) it is necessary to detect eyes with partially occluded pupil and with iris occlusions more than half of perimeter. Thus the task of center detection is the locating of the common center for a set of several disjointed arcs, which are parts of pupil and iris borders broken (interrupted) by eyelids, eyelashes, specular reflections, boundaries of image. Hereinafter we speak about pupil and iris border circles having in mind that they may be seen only partially in image. Figure 1 gives two images with occlusions from ND-IRIS database [9]. Left image is not suitable for recognition since only small (less than 20%) of iris surface is visible. However, such occluded images are quite common in the input of real iris recognition systems, and iris location should be determined in order to perform eye tracking and positioning tasks of the system. Right image of Figure 1 illustrates yet other difficulty of iris location problem. In this image both eyebrow and

* Supported by RFFI grants 11-01-00781 and 11-01-90435

eyelid are darker and bigger than pupil. Hence it is not sufficient to apply segmentation by brightness and look for "biggest dark object" (as it was proposed by earlier methods). Involving circularity properties of iris and pupil is necessary. Daugman [2] proposed a stable method for detecting eye center as a position of maximum response of integro-differential circular-symmetric operator. However, this method is calculation-costly and cannot be applied to real-time processing. Methods based on color segmentation like [5,11] can be used only with color images, whereas iris recognition images are usually monochrome and registered in infrared range.



Figure 1. Samples of occluded iris images

Here a Hough method is exploited. It is stable against noise and breaking figure into fragments. The voting to accumulator is performed by a segments originating from points with high brightness gradient and spanning in the direction of anti-gradient. The length of the segment is selected so as to cover all possible pupil and iris radii. Places with higher density of these segments correspond to center of circles in images. Blurring of accumulator is added to decrease noise and enhance stability. Direction of voting segments is adjusted to compensate natural slightly elliptic shape of iris. Darker points vote with higher values to favor pupil and iris borders, which are usually darker than the majority of edge points of the image.

2. METHOD

The method was developed so as to be applicable to any images, which can be used for iris recognition. Thus the set of image conditions is an "intersection" of characteristics of all known systems and is worse than any one of them. Input data are raster monochrome images with signal-to-noise ratio 30dB (5 binary digits). Typical image size is $X * Y = 640 * 480$ pixels (at least, all databases used for tests originally have this image size). Iris diameter in images of the analyzed databases ranges from 150 to 400 pixels. Pupil spans from 20 to 75% of iris radius (i.e. from 20 to 300 pixels), and nearly same range of pupil to iris ratio is given in [4]. Presence of specular reflection in any place of image as well as absence of the reflection is possible.

Algorithm is executed as a sequence of steps.

Step 1: Gradient calculation

Denote image brightness as $b(x, y)$. Its gradient: $\vec{\nabla}b(x, y) = \vec{g}(x, y) = (g_x(x, y), g_y(x, y))^T$ is calculated as a convolution $g_x(x, y) = S_x * H_\sigma * b(x, y)$, $g_y(x, y) = S_y * H_\sigma * b(x, y)$, where H_σ is a Gaussian with spread parameter $\sigma = 4.0$, $S_x = \begin{pmatrix} -1 & 0 & 1 \end{pmatrix}$ and $S_y = \begin{pmatrix} -1 & 0 & 1 \end{pmatrix}^T$ are simplest masks for taking derivatives in horizontal and vertical direction. Gradient vector lengths $G^2(x, y) = g_x^2(x, y) + g_y^2(x, y)$ are also estimated. Distribution of these values is collected to a histogram $H_g(l) = \left| \left\{ (x, y) : G^2(x, y) = l \right\} \right|$ that is used in further steps. The algorithmic complexity of this step is linear relevant to the number of image pixels, i.e. proportional to $X * Y$, and processing of each pixel requires approximately two dozen CPU operations.

Step 2: Building the "voting set"

It is proposed to search circle of arbitrary size by drawing a straight lines along the brightness gradient at selected points in Hough accumulator space. In our case sizes of pupil and iris circles are limited by image size. For the purpose of recognition iris should not span outside of image. Hence iris diameter should not exceed image size and iris radius cannot exceed $R_M = 0.5 \min\{X; Y\}$ pixels. In databases used for tests it is not above 200 pixels. Thus a segment of limited length is drawn in the accumulator. Call it "voting set" or "voting segment". Points with big gradient value may belong both to pupil and iris border. Thus two voting sets should be drawn to vote for both hypotheses simultaneously. To vote for the hypothesis that the point

belong to iris border it is necessary to draw voting segment in a direction opposite to brightness gradient and the length of the segment should cover all possible iris sizes, so the segment should span the range $[R_m; R_M]$, where $R_M = 50$ (as required by [3], or some other expected minimum value) and R_M are minimum and maximum possible iris radii respectively. Analogously, the voting for pupil hypothesis is the segment going also in the opposite direction to brightness gradient and spanning the range $[r_m; r_M]$, where $r_m = 0.2R_m = 10$ and $r_M = 0.75R_M$ are minimum and maximum possible pupil radii. Joining tests of two hypothesis the "signal set" is obtained as:

$$v(r) = \begin{cases} 1, & \text{if } r \in [r_m; r_M) \\ 0, & \text{otherwise} \end{cases} + \begin{cases} 1, & \text{if } r \in [R_m; R_M) \\ 0, & \text{otherwise} \end{cases}$$

where positive values of r go in anti-gradient direction.

Step 3: Voting to accumulator

The voting is performed in all points that are considered as possibly belonging to pupil or iris border. These are the points with high value of brightness gradient. The threshold for selection high gradient is calculated as $T_1 = \max\{6\sqrt{2} \max\{\sigma, 2\}, T_{75}\}$, where σ is mean square deviation of brightness value caused by noise, T_{75} is a value of gradient such that 75% of image points have gradient below it, i.e. is

calculated from a condition $\sum_{l=0}^{T_{75}} H_g(l) = 0.75 \sum_l H_g(l)$. Figure 2a depicts such points obtained from iris

image shown in Figure 1b. Drawing the voting set (i.e. segment in the discrete raster) is done using Bresenham's algorithm. Sample of parameter space (accumulator $A(x, y)$, which is geometrically congruent to the source image) obtained at this step is given in Figure 2b. The algorithmic complexity of the voting procedure is XYR_M , so it is above linear and is the most complex one among all steps, especially for images of bigger size. However, each step of Bresenham segment drawing is performed in less than dozen CPU operations.

Step 4: Blurring the accumulator

Parameter space is subjected to a low-pass filtration to outline the positions of maximum accumulated parameter density: $A_b(x, y) = H_\sigma * A(x, y)$, where $A(x, y)$ is a Hough transform accumulator obtained on the previous step. Gaussian kernel same as in Step 1 is used.

The algorithmic complexity of the blurring procedure is XY

Step 5: Maxima location

Position of maximum in blurred accumulator is the most probable center of an eye $(x_c; y_c)$. If maximum value is below threshold T_2 , it is stated that no eye is present in image:

$$(x_c; y_c) = \begin{cases} \arg \max_{(x,y)} A_b(x, y), & \text{if } \max_{(x,y)} A_b(x, y) > T_2 \\ \text{none,} & \text{otherwise} \end{cases}$$

The value of T_2 is determined from the following. Consider image with random brightness. Gradients of brightness will also have random direction distributed evenly. Rejecting low level gradients with T_{75} will yield one fourth share of points also randomly scattered in image. Drawing segments of length R_M will make an average density of $0.25R_M$ in accumulator. Taking twice this value seems a good threshold to reject spurious center detections due to image noise. Hence, the threshold is set as $T_2 = 0.5R_M$

Two heuristic modifications taking into account the nature of human iris are introduced in the algorithm.

Adjusting gradient. Both iris and pupil contours are used for eye detection. Iris has a shape of ellipse that is flattened in vertical direction. Average size of iris is 12.5 mm horizontally and 11.8 mm vertically [4]. Thus iris is an ellipse with horizontal mean axis and eccentricity $\varepsilon \approx 0.06$. Search of ellipses with known eccentricity and orientation is easily reduced to search of circles by affine image transform. For described

method even simpler solution is possible that is adjusting gradient direction. In order to be directed to a center of ellipse with horizontal big axis and eccentricity \mathcal{E} horizontal component of (anti-)gradient calculated at its border should be multiplied by $(1 + \mathcal{E})^2$. As for the pupil, it is a circle in average by population, its expected shape is circular, hence no gradient adjustment is needed for pupil detection. In order to search both pupil and iris by a single process, it is advisable to take an average of unit and $(1 + \mathcal{E})^2$.

Since \mathcal{E} is small, this is simply $(1 + \mathcal{E})$ or the ratio of big to small axis of iris ellipse. Thus, horizontal gradient component at the step 1 is additionally multiplied by 1.06.

Favoring darker points. Iris and pupil border points have high brightness gradient values, but high gradient occurs in many other points, like eyelids and specular reflections. At that eyelid border can produce false position of eye center since it can be almost ideal circular arc. However, iris and pupil borders are usually darker than other edge points. To emphasize the influence of dark points voting to accumulator the value of $v(r)$ is multiplied by $255 - b(x, y)$ in Step 3.

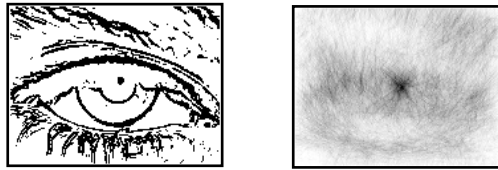


Figure 2. (a) points with gradient value above T_{75} , (b) accumulator

3. EXPERIMENTS

The following publicly available databases were used for method verification:

- UBIRIS.v1 database [10], 1204 images;
- BATH database [8], 13177 images (not all were downloaded).
- CASIA-IrisV3-Lamp database [1], 16180 images;
- ND-IRIS database [9], 64980 images.

Testing method. Eye images were reviewed by human expert who indicated pupil border as a circle in each of them. These data were then considered as "ground-truth" and were used for method verification. Then images were processed automatically. Below is the table representing numbers of images from these four databases where eye center was detected with specified error. The error is calculated as a ratio of displacement of detected center from true center to a true radius.

Table 1a. Distribution of center detection error.

Range of detection error	UBI	BATH	CASIA	ND-IRIS	Total
$E < 0.1$	416	11744	11143	59472	82775
$0.1 \leq E < 0.2$	42	1374	4497	4568	10781
$0.2 \leq E < 0.3$	180	54	444	711	1389
$0.3 \leq E < 0.4$	128	4	58	157	347
$0.4 \leq E < 0.5$	69	1	25	19	114
$0.5 \leq E < 0.6$	32	0	6	13	51
$0.6 \leq E < 0.7$	20	0	6	3	29
$0.7 \leq E < 0.8$	10	0	0	0	10
$0.8 \leq E < 0.9$	2	0	0	0	2
$0.9 \leq E < 1.0$	0	0	0	0	0
$1.0 \leq E$	5	0	1	37	43
Total	1204	13177	16180	64980	95541

Table 1b summarizes some more characteristics of the algorithm execution on the databases. These are: average error (average displacement of detected center from true center) expressed in pixels and in percent of true pupil radius, median value of displacement (i.e. 50% quantile of error), and 0.1% quantile of error (such value that the algorithm gives bigger displacement in 1 of 1000 images of the database). Execution times measured for PC with PIV-3GHz processor are included.

Table 1b. Average characteristics and error quantiles.

Characteristic	UBI	BATH	CASIA	ND-IRIS	Total
Average error, in %	19.6	5.07	7.91	4.33	5.23
Average error, in pixels	7.62	3.50	3.23	2.43	2.78
Median error, in %	14	4	7	3	4
Median error, in pixels	6	3	3	2	2
0.1% error quantile, in %	100	24	45	38	40
0.1% error quantile, in pixels	80	21	16	21	20
Average calculation time, msecs	256	271	249	282	275
Maximum calculation time, msecs	318	378	291	425	425

In order to reduce time consumption the algorithm was executed with images scaled by factors 2, 4 and 8. Values of X , Y and derived ones r_M , T_2 et.c. were changed appropriately.

Table 2. Results of center detection in original images and scaled with factors 2,4 and 8.

Characteristic	Original (scale 1)	Scale 2	Scale 4	Scale 8
$E < 0.1$	82775	82165	77726	56935
$0.1 \leq E < 0.2$	10781	10979	15161	30604
$0.2 \leq E < 0.3$	1389	1707	1887	5976
$0.3 \leq E < 0.4$	347	401	450	1277
$0.4 \leq E < 0.5$	114	127	159	362
$0.5 \leq E < 0.6$	51	62	62	142
$0.6 \leq E < 0.7$	29	33	24	50
$0.7 \leq E < 0.8$	10	10	12	31
$0.8 \leq E < 0.9$	2	5	2	13
$0.9 \leq E < 1.0$	0	1	0	12
$1.0 \leq E$	43	49	59	139
Average error, in %	5.23	5.57	6.55	10.13
Average error, in pixels	2.78	2.92	3.42	5.08
Median error, in %	4	4	6	9
Median error, in pixels	2	3	4	5
0.1% error quantile, in %	40	56	61	100
0.1% error quantile, in pixels	20	24	27	70
Average calculation time, msecs	275	37.1	5.14	1.38
Maximum calculation time, msecs	425	60.4	7.97	2.02

4. CONCLUSION

It can be seen that even scaling by the factor 4 gives satisfactory results, with the throughput of more than 200 frames per second, that is adequate to process typical video stream for iris recognition (640*480*30 fps). Typical error of the method (almost all cases where detected center jumps out of true pupil) is locating the center in the middle of eyeslot for nearly closed eye or in the corner of an eye. Such images are usually strongly occluded. This typically happens in low contrast or extremely occluded images. The algorithm is being used for detection of eye center in the iris image capture device.

REFERENCES

1. Chinese Academy of Sciences Institute of Automation., 2010. CASIA Iris image database. Available from <<http://www.cbsr.ia.ac.cn/IrisDatabase.htm>>.
2. Daugman, J. 2004. How iris recognition works *In Proc. IEEE Trans. Circ. Syst. Video Technol.* Vol. 14, No. 1, pp.21-30.
3. ISO/IEC 19794-6:2005 Information technology - Biometric data interchange formats - Part 6: Iris image data.
4. Jack J. Kansky, 2003 *Clinical Ophthalmology: a Systematic Approach*. Elsevier, London (5th ed.)
5. Kashima, H. et al., 2001. A Robust Iris Detection Method of Facial and Eye Movement. *Proceedings of Vision Interface Annual Conference*. Ottawa, Canada
6. Lipinski, B., 2004. Iris Recognition: Detecting the pupil Available from <<http://cnx.org/content/m12487/latest/>>
7. Maenpaa, T., 2005. An Iterative Algorithm for Fast Iris Detection *Proceedings of IWBRIS* Beijing, China, pp. 127-134.
8. Monro, D., 2005. University of Bath Iris Image Database Available from <<http://www.bath.ac.uk/elec-eng/research/sipg/irisweb/>>
9. Phillips, P., Scruggs, W., O'Toole, A., Flynn, P., Bowyer, K., Schott, C., and Sharpe, M., 2010. Frvt2006 and ice2006 large-scale experimental results. *In IEEE PAMI*, Vol. 32, No. 5, pp. 831-846.
10. Proenca, H., Alexandre, L., 2005. UBIRIS: A noisy iris image database. *Proceedings of 13th International Conference on Image Analysis and Processing*. Volume LNCS 3617, pp. 970-977, Cagliari, Italy.
11. Samet, R., et al., 2010. An implementation of automatic contour line extraction from scanned digital topographic maps *In Appl. Comput. Math.*, Vol. 9, No. 1, pp.116-127.

APPROXIMATION OF 3D DATA WITH PIECEWISE SERIES EXPANSIONS FOR SURFACE INSPECTION

Sebastian von Enzberg and Bernd Michaelis
Otto-von-Guericke University Magdeburg
Universitätsplatz 2, 39106 Magdeburg

ABSTRACT

Optical 3D measurement with active photogrammetric systems allows for precise measurement with high lateral and depth accuracy. Measured 3D data can thus be used for surface quality inspection. The detection of surface defects is a non-trivial task especially for deformable plastics or sheet metal parts and can be solved by 3D surface approximation with series expansion. In this paper we show that splitting of complex parts into smaller patches with piecewise series expansions yields better part approximation. A method is proposed which retains smooth transitions between these patches, thus allowing continuous measurement along patch boundaries.

KEYWORDS

Optical 3d measurement, surface quality inspection, 3d data approximation

1. INTRODUCTION

Surface quality inspection is the task of finding defects like dents, bumps, waviness or sink marks on free-form surfaces. It is an important tool for quality control in industrial manufacturing of premium goods like car body and interior parts, where a perfect, defect-free surface is crucial for customer acceptance [10]. Requirements in the automobile industry reach down to 10 μm at a single measurement of a car body part, which is close to noise level of common measurement methods [5]. This is an especially hard task for the measurement of sheet metal or plastics parts. Here a systematic error is introduced by deformation resulting from non-ideal placement on the measuring support. The surface defects to be detected may be inside of a given geometric tolerance band.

Optical measurement methods combined with 2D and 3D Computer Vision algorithms are well suited for inspection of surfaces, because of fast, contact-free, automatable and scalable measurement. 3D Vision is commonly used for *geometric* inspection [4], however *surface quality* inspection is still mainly accomplished with 2D methods [3][14][7][9]. This implies a homogeneously textured surface or a complex lighting setup that make defects visible, which increases setup time and involves the need to adapt algorithms for specific surface shapes. Also surfaces with areas of high curvature or discontinuities like edges, creases or moldings are hard to inspect with 2D algorithms. With the growing resolution of image sensors [8] and the advent of high precision 3D measurement methods based on active photogrammetry with fringe projection like phase shifting [6], the high lateral and depth accuracy of 3D measurement required for surface quality inspection can be achieved even for large measurement volumes [1]. While this allows a higher degree of automation since a more general sensor setup can be used for most surfaces, new methods and algorithms for high accuracy processing of 3D data have to be developed and evaluated.

Normally, surface quality inspection is realized by comparing a measurement to a reference piece and evaluating their difference. For deformable parts, a direct comparison to CAD is not possible. One method to obtain an adapted reference piece to a measurement is by a finite series expansion. After an overview of series expansions for surface quality inspection in section 2, we show that splitting of a measurement into patches yields even better approximation results (section 3). To make a continuous inspection over patch boundaries possible, we derive conditions for retaining smooth transitions between several surface patches and their series expansion (section 4). Experimental results of our proposed method and conclusions are given in section 5 and 6, respectively.

2. SERIES EXPANSIONS FOR SURFACE APPROXIMATION

In general, a series expansion is a weighted sum of base functions w_i . To allow simple transform of a function into a new base, it is reasonable to require the w_i to form an orthogonal base of the Hilbert space. Approximation of a function $z(x,y)$ is achieved by truncating the series to a sum of n base functions $w_1 \dots w_n$.

$$z \approx f(\vec{\alpha}) = \sum_{i=1}^n \alpha_i w_i \quad (1)$$

with coefficients α_i . Base functions used for surface approximation are e.g. polynomials [2], wavelets [11] or sinusoids [12]. Usually the omission of higher base functions leads to a smoothed surface or in terms of signal processing a low-pass filtered surface. Using this approach, we assume that the defects are short-wave compared to the overall surface curvature. If the surface contains short-wave features by design, they will be mistakenly detected as defects. While this may be manually corrected by masking these areas, it is impossible to find defects on highly curved surfaces areas. Also defects with a large lateral spread pose a problem to low-pass methods. In [5] adapted base functions are computed by statistical analysis (discrete Karhunen-Loève-transform) of several faultless parts, which resolves the flaws of simple low-pass filtering.

For digitized surfaces, it is common for the depth values to be given on a discrete x - y -grid. The measurement values at these positions may then be vectorized as \vec{z} , leading to discrete base vectors $\vec{w}_{*,i}$. Eq. 1 may then be written in matrix form

$$\vec{z} = \sum_{i=1}^n \alpha_i \vec{w}_{*,i} + \vec{\varepsilon} = W\vec{\alpha} + \vec{\varepsilon} \quad (2)$$

with the matrix of base vectors $W=[w_{j,i}]=[w_{*,1} \dots w_{*,n}]$, coefficient vector $\vec{\alpha}$, and approximation errors $\vec{\varepsilon}$. The coefficients α_i can be calculated by minimizing the residual error $\|\vec{\varepsilon}\|$, leading to

$$\vec{\alpha} = W^{-1} \cdot \vec{z} \quad (3)$$

3. IMPROVED APPROXIMATION BY SPLITTING INTO PATCHES

For the following experiments, a detail of a car door is used which has been measured using a multi-camera system with fringe projection and the phase shift method, similar to the setup given in [6]. A car door is made up of large smooth areas as well as complex geometric features (door handle cup), which makes surface inspection difficult and thus a good benchmark of surface inspection methods for industrial applications. We used the method described in [5] to compute adapted base vectors from 80 samples of different good parts and measurements; the considered surface area is approx. 30 cm × 30 cm. The parts were systematically cut in half by alternating horizontal and vertical splits, resulting in an increasing number of patches of decreasing size (fig. 1).

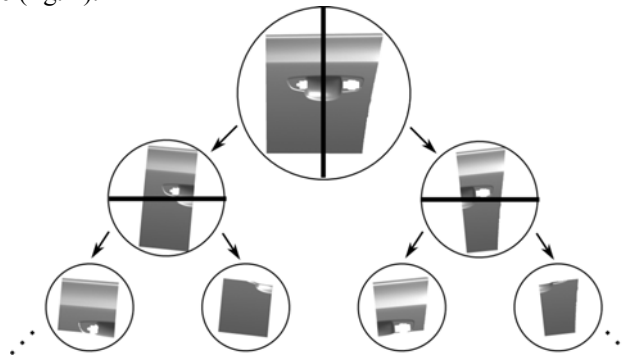


Figure 1. Alternating horizontal and vertical splitting of a part (car door) into patches

As a measure of approximation quality, the standard deviation $\text{std}(\vec{\varepsilon})$ and maximum error $\max(\vec{\varepsilon})$ of the residual vector $\vec{\varepsilon}$ (cf. eq. 2) is used. Fig. 2 shows the averaged results of 10 different measurements as a function of the number of patches. The approximation error of a part can be decreased to approx. 30% of the original error by splitting into 16 patches. In fig. 3, the lateral distribution of the approximation error can be seen for the examined part. The overall approximation error has decreased, but the patch boundaries are clearly visible, thus making automated evaluation of defects difficult.

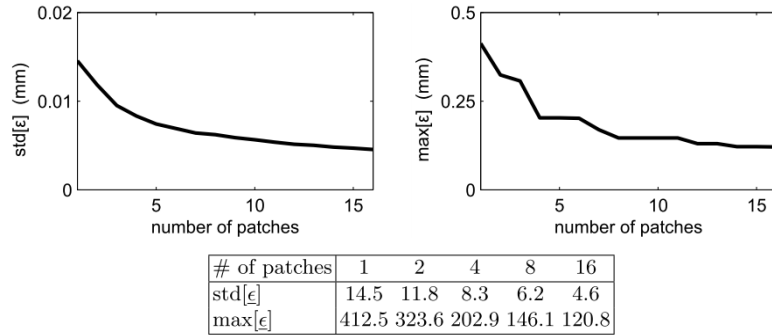


Figure 2. Standard deviation (top left) and maximum error (top right) of the residual error against number of patches. Approximation with $n = 10$ base vectors

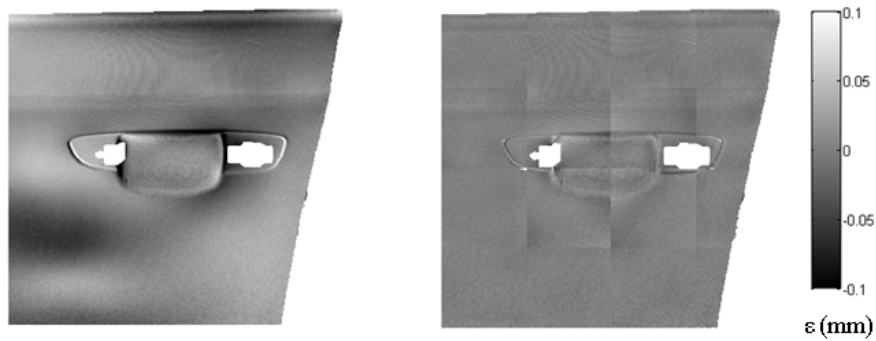


Figure 3. Approximation error map of a car door for a single series expansion (left) and 16 separate surface patches (right). The approximation error in z-direction is coded into gray scale values and given on an x - y -grid

4. RETAINING SMOOTH PATCH TRANSITIONS

In the following section constraints for the general series expansions in eq. 2 are derived that assure smooth transitions between two patches, thus making automated defect detection possible. For C^1 -continuity, the measurement value as well as the derivative have to be equal for both patches at their transition points. For points with indices s and t on patches $P1$ and $P2$, this means

$$f_s^{P1} = f_t^{P2} \tag{4}$$

$$\frac{f_s^{P1} - f_{s-1}^{P1}}{\Delta s} = \frac{f_t^{P2} - f_{t-1}^{P2}}{\Delta t} \tag{5}$$

Presuming equidistant points ($\Delta s = \Delta t$), we get constraints

$$Q_0 = \|f_s^{P1} - f_t^{P2}\| \xrightarrow{!} \min \tag{6}$$

$$Q_1 = \|(f_s^{P1} - f_t^{P2}) - (f_{s-1}^{P1} - f_{t-1}^{P2})\| \xrightarrow{!} \min \tag{7}$$

which can be solved by least squares minimization. For (6), this leads to the homogenous linear system of equations

$$\begin{bmatrix} T_0^{11} & T_0^{12} \\ T_0^{21} & T_0^{22} \end{bmatrix} \cdot \begin{pmatrix} \vec{\alpha}^{P1} \\ \vec{\alpha}^{P2} \end{pmatrix} = T_0 \cdot \vec{\alpha} = 0 \quad (8)$$

with sub-matrices

$$T_0^{11} = (\vec{w}_{s,*}^{P1})^T \vec{w}_{s,*}^{P1} \quad (9)$$

$$T_0^{22} = (\vec{w}_{t,*}^{P2})^T \vec{w}_{t,*}^{P2} \quad (10)$$

$$T_0^{12} = -(\vec{w}_{s,*}^{P1})^T \vec{w}_{t,*}^{P2} \quad (11)$$

$$T_0^{21} = -(\vec{w}_{t,*}^{P2})^T \vec{w}_{s,*}^{P1} \quad (12)$$

In eq. 8 we introduced the *Transition Matrix* T_0 which leads to C^0 -continuous patch transitions and thus is the core of our proposed method for retaining smooth patch transitions. It holds the conditions which can be used for correction of the coefficient vectors $\vec{\alpha}^{P1}$ and $\vec{\alpha}^{P2}$ of the series expansion. The transition matrix T_0 is constructed by using row vectors of the matrices of base vectors W^{P1} and W^{P2} used for series expansion of patches $P1$ and $P2$. Its size depends on the number of base functions chosen for expansion, which usually is much smaller than the dimensionality of the measurement vectors themselves and thus is invertible with little computational effort. Similarly, the Transition Matrix T_1 for C^1 -continuous patch transitions can be derived from eq. 7. Both Transition Matrices can be combined with the computation of series coefficients given in (3), leading to

$$(\rho_0 T_0 + \rho_1 T_1 + I_{n_1+n_2}) \cdot \vec{\alpha} = \begin{pmatrix} (W^{P1})^{-1} \vec{z}^{P1} \\ (W^{P2})^{-1} \vec{z}^{P2} \end{pmatrix} \quad (13)$$

where $I_{n_1+n_2}$ is the unit matrix of size $(n_1+n_2) \times (n_1+n_2)$ and $\rho_0, \rho_1 \geq 0$ are regularization coefficients for controlling the amount of adjustment, with $\rho_0 = \rho_1 = 0$ yielding the standard transform (3) to the new base vectors without consideration of the constraints (4), (5). For surfaces, several points along the patch boundary have to be considered, resulting in single Transition Matrices $T_{0,i}$ and $T_{1,i}$ for each pair of boundary points $f_{s,i}^{P1}, f_{t,i}^{P2}$. It can be shown that these matrices can be combined into total Transition Matrices by summation.

5. EXPERIMENTAL RESULTS

In fig. 4 we see the approximation error along the transition line of two patches. Due to different base vectors, a significant discontinuity appears between the two patches (fig. 4, left). With the proposed method (fig. 4, right), the difference of approximation errors along the patch border is decreased by 60%, leading to smoother patch transitions. Remaining deviations are of much smaller degree and can be smoothed by ordinary low pass filtering (e.g. Gaussian filter).

The result can be seen in fig. 5, where the approximation error values are visualized in gray scales on the x-y-grid. Without correction, the vertical split is clearly visible even after applying a Gaussian filter mask. Again, the standard deviation of the error $\text{std}(\vec{\varepsilon})$ is used as a measure of overall approximation quality. As shown in section 3, approximation quality is better for evaluation in smaller patches. Applying the Transition Matrices, the approximation quality slightly decreases, but still remains better than without patching (fig. 5).

The same experiments have been conducted for measurement data of the door handle cup only, where the surface shows a higher degree of detail and curvature; also a small defect (a pit in the middle of the door handle) was included in this measurement. Here the results for our proposed correction method (fig. 6) are even better with a remaining transition error $\|\vec{\varepsilon}_{P1} - \vec{\varepsilon}_{P2}\|$ along the patch boundary of only 0.05 mm compared to 0.34 mm without correction, making the automatic detection of the defect easier.

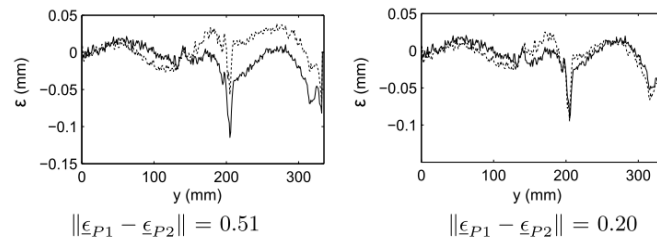


Figure 4. Left: approximation error along the transition line between patch 1 (solid line) and patch 2 (dotted line). Right: approximation error with the proposed correction method

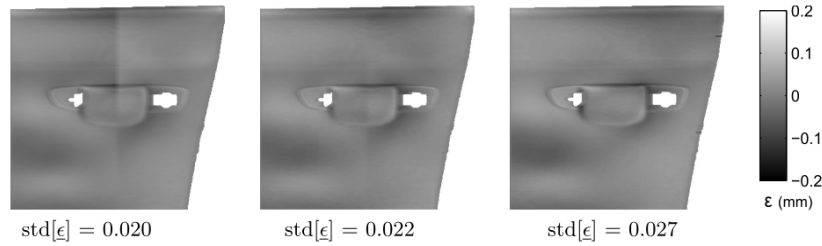


Figure 5. Approximation error maps of a car door, post processed with additional smoothing. Left: with 2 vertically split patches, middle: with 2 patches and proposed correction method, right: whole part without patches

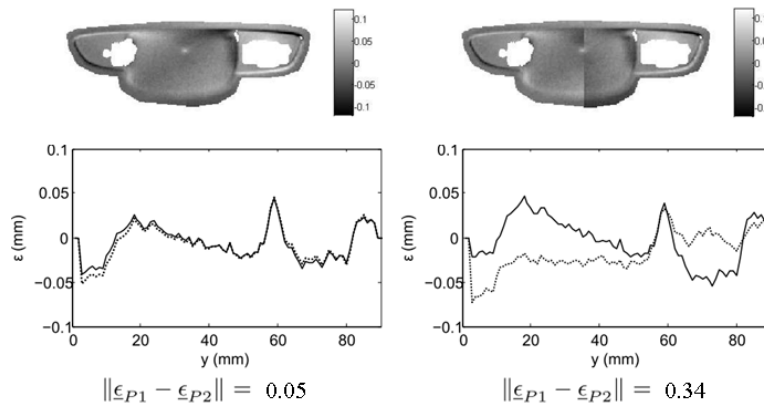


Figure 6. Approximation error maps (top) and errors along the transition line for patch 1 and patch 2 (bottom) for a door handle cup without additional smoothing. Left: results with proposed correction method, right: without correction

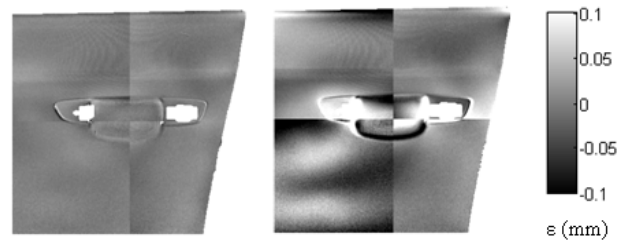


Figure 7. Approximation error maps of a car door, split into 4 patches without further post processing. Left: without correction of patch boundaries, right: with the proposed correction method

Fig. 7 shows the resulting approximation errors for 4 patches, which significantly increases due to the additional constraints introduced by the new patches. For the 8 base vectors used in this example, it is not possible to properly approximate the surface and hold smooth patch transitions at the same time.

6. CONCLUSIONS

High precision surface approximation from 3D point clouds as needed for surface quality inspection of deformable parts can be achieved by finite series expansion. For the example of a car door we have shown that splitting of the measured point cloud into smaller patches and processing patches with separate series expansion can significantly increase approximation quality while using the same amount of series base functions per patch. For large patches this may also decrease computational complexity, especially when using adapted base functions. A systematical measurement error is introduced along patch boundaries.

We have further derived constraints for the computation of coefficients for the series, that allow smoothing of the transitions between patch boundaries. As the degree of freedom for the computation of coefficients is lowered, the approximation quality decreases. For the chosen example, the total approximation quality with two patches is still better than for single surface approximation. However when using more than two patches, the approximation quality may be degraded significantly. While the derived constraints for smooth patch boundaries are mathematically general and rigorous, further research focus has to be put in the choice and design of base functions to get applicable results for high precision surface approximation.

A suitable solution may be B-spline base functions [13] which allow a higher local control of the series coefficients, meaning lateral surface approximation and approximation along patch boundaries have fewer influence on each other. Our further research towards new algorithms for surface quality control will include approximation with B-spline base functions under high precision considerations, which are needed to match current and future industrial requirements and which have been neglected in the context of surface approximation till now.

REFERENCES

- [1] Cuypers, W. et al, 2009. Optical measurement techniques for mobile and large-scale dimensional metrology. *Optics and Lasers in Engineering*, Vol. 47(3-4), pp. 292-300.
- [2] Helzer, A., Barzhoar, M. And Malah, D., 2004. Stable fitting of 2D curves and 3D surfaces by implicit polynomials. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 26(10), pp. 1283-1294.
- [3] Jähne, B. et al, 1996. *Technische Bildverarbeitung - Maschinelles Sehen*. Springer, Berlin, Germany.
- [4] Li, Y. and Gu, P., 2004. Free-form surface inspection techniques state of the art review. *Computer-Aided Design*, Vol. 36(13), pp. 1395-1417.
- [5] Lilienblum, T. et al, 2000. Dent Detection in car bodies. *Proceedings of the 15th International Conference on Pattern Recognition*, Vol. 4, pp. 775-778.
- [6] Lilienblum, E. and Michaelis, B., 2007. Optical 3D surface reconstruction by a multi-period phase shift method. *Journal of Computers*, Vol. 2(2), pp. 73-83.
- [7] Louban, R., 2009. *Image Processing of Edge and Surface Defects: Theoretical Basis of Adaptive Algorithms with Numerous Practical Applications*. Springer, Berlin, Germany.
- [8] Luhmann, T., 2010. Close range photogrammetry for industrial applications. *ISPRS Journal of Photogrammetry and Remote Sensing*, Vol. 65(6), pp. 558-569.
- [9] Ohl, A., 2005. *Modellierung von Gestaltmerkmalen für die Oberflächeninspektion*. Shaker Verlag, Aachen.
- [10] Özkul, M., 2009. *Qualitätsansprüche bezüglich des äußeren Erscheinungsbildes von Automobilen der Premiumklasse*. Hieronymus, Munich, Germany.
- [11] Recknagel, R.-J., 2000. *Defekterkennung an Oberflächen mittels Waveletmethoden*. PhD Thesis, University Jena.
- [12] Taubin, G., 1995. A signal processing approach to fair surface design. *Proceedings of the 22nd annual conference on Computer graphics and interactive techniques*, pp. 351-358, ACM, New York, USA.
- [13] Unser, M., 1993. B-Spline Signal Processing: Part I - Theory. *IEEE trans. on sig. proc*, Vol. 41(2), pp. 821-832.
- [14] Xie, X., 2008. A review of recent advances in surface defect detection using texture analysis techniques. *Electronic Letters on Computer Vision and Image Analysis*, Vol. 7, No. 3, pp. 1-22.

Reflection Papers

IMPLEMENTING COMPOSITING MANAGERS IN EMBEDDED SYSTEMS

Mirto Musci

*University of Pavia – Dpt. of Informatics and System Science
Via Ferrata 1, 27100 Pavia (Italy)*

ABSTRACT

This paper discusses the issues arising from the implementation of a graphical stack in embedded systems, to support the advanced features of GUI based on the *compositing* paradigm. Compositing managers allow the set up of flexible, fully-fledged GUIs that rely on 2D and 3D graphical effects. The underlining software architecture has specific characteristics that require careful implementation if the resources of the system are limited, as in the embedded case.

The discussion is general, but it is focused on open platforms. The case of the soon-to-be-released MeeGo OS is used as a case instance, along with a specific device from ST-Ericsson.

KEYWORDS

GUI, compositing managers, embedded systems

1. INTRODUCTION

Compositing managers (CMs in the following) are components embedded in a graphical system (GS) or simply part of it. Their function is to enrich user experience by modifying the GUI paradigm, so providing various graphical effects to make interfaces more accessible or aesthetically pleasing. The most common effects obtainable are exposé-like task switching, window transparency as well as flickering elimination through double-buffering. Other effects include screen magnification, desktop switching and quick widget laying. Nowadays, CMs are part of most desktop OS. Microsoft Windows (since Vista), Mac OS X (since the very first X release) and many, if not all, GNU/Linux distributions. In literature, the discussion is focused mainly on Unix-X Server releases of compositing managers due to availability of sources and information. This article follows this approach.

To introduce composition in the Linux platform, the X Server had to be redesigned [Gettys 2004]. The very first CM, Metacity 2.8.4, was released in 2004 [Paul 2005]. Compiz was released in 2006, and was the first fully-accelerated 3D CM on a Linux platform. Compiz was based on the Glitz library, originally developed by Nilsson and Reveman [Nilsson & Reveman 2004]

In the last few years, demand for composition implementation in embedded platforms has grown. Due to the very nature of these systems, porting a CM from desktop environment has a lot of caveats, which this article will try to illustrate and clarify. A lot of different approaches are available, and still there is a lot of platform-dependent work to be done on each separate machine. The article is structured as follows: at first a short introduction to a general CM architecture will be given (section 2). Then the embedded system approach to CM implementation will be discussed (section 3). Finally a real case analysis of embedded system composition will be briefly presented (section 4).

2. COMPOSITING MANAGER ARCHITECTURE

The most important concept in CM architecture is *redirection*. Usually, every window managed by the graphic system is directly painted on the display, while the Window Manager (WM) merely communicates with the applications to set size, position and other attributes of the windows. When a CM is running, each

applications output, instead, is redirected to a separate off-screen buffer that is explicitly ignored by the GS. Thus the CM has the responsibility to paint applications content. CM and WM can cooperate as separate components or their functions can be performed by a single object, called Window Compositing Manager.

In the GNU/Linux/Unix world, the GS is structured as a client-server architecture, so the CM (as well as the WM) act as a client, interacting with its X Server counterparts: the three fundamental extensions “Composite”, “Damage” and “X Fixes” [Gettys 2004]. Basically, these extensions provide redirection to the GS coupled with the ability to monitor and report windows content change (damage) events to the CM client.

The CM manipulates the buffers, applies any number of graphical effects and builds a representation of the screen. To perform this task, any different graphic library can be used: Qt, Gtk+, etc. Once the representation is created, the CM has to paint it on the display. In doing this, CM can use 2D or 3D painting functions, provided by the X Server or external libraries, such as OpenGL [Leech 2005]. Server side support is granted by “Render” extension [Packard 2009].

This final step is commonly referred to as *rendering*, and it is of course the most critical part in a CM implementation. Both 2D and 3D functions can be software only or hardware accelerated and can possibly interact with the X Server to get their job done, or simply bypass it.

Either way, the objects being manipulated are pixel buffers of different sizes that represent application window contents and other supporting images. The main tool employed is alpha-compositing, which relies on transparency channels and image blending. These techniques were first described by Porter and Duff in 1984 and are well known in the computer graphics world [Porter & Duff 1984]

3. COMPOSITION IN EMBEDDED SYSTEM

At the time of writing, commercial embedded CMs are relatively few. Among these is Matchbox, an open source window manager designed for embedded systems with X Window System support. The main difference between Matchbox (and most embedded CMs) and its desktop counterparts, is the self-imposed limitation of being able to show and manage only one window at a time. Matchbox has been used in previous versions of Maemo [Matchbox Project 2011]. Other two noteworthy X embedded CMs are SurfaceFlinger (SF), included in the Android OS, and MeeGo Composer, which will be discussed in detail later. According to the source code documentation, SF relies on a “surface” abstraction to manage window content. SF depends on a WM that controls position, size, and Z-ordering of Android windows [Wu 2009].

3.1 Implementation Issues

Implementing a compositing manager in an embedded system is not an easy task. For instance, these systems are generally short on computing power, memory and graphic capability. Under these conditions using a pure software rendering is not feasible, so most if not all embedded GSs have to use some form of graphics acceleration. This means that underlying hardware should be equipped properly. At the same time this also means that compositing managers have to be scaled down to meet strict requirements. As a consequence, a lot of graphical effects and functionality could be lost. For example, 3D effects could be dropped entirely. In some cases, implementing a compositing manager is simply not possible.

Even when using hardware acceleration, another problem arises due to the 3D library being used, which is normally OpenGL ES. The initial specification of this library, still supported by several machines on the market, has a number of limitations that are particularly relevant to the rendering of a GUI. This feature is not surprising, since OpenGL ES was created primarily to support video gaming.

The biggest problem is the size of the texture, limited to 64x64 texels, which makes it very difficult to manage texture buffers obtained from redirection. OpenGL ES supports texture compression, which would allow simple management of the texture size required. This system can certainly be appropriate for video games, in which textures are loaded only once and basically do not change throughout program execution. On the contrary, compression cannot be used implementing an interface, because the textures associated with the windows are updated very frequently [Larsson & Klinstrom, 2006].

These problems, fortunately, have been fixed in version 2.0 of OpenGL ES, a much more complete specification, but at the same time much more demanding on hardware resources. Currently there are few machines that fully support it (e.g. Nokia N900 or STU8500).

3.2 Optimizing Composition with Hardware Accelerators

It is very unlikely for an embedded system to host a full-fledged 3D graphics card. Compositing operations could be very costly, even if scaled down. At the same time, users demand a minimum set of graphics effects. The solution is to enrich the architecture with a set of very specific hardware accelerators, expressly tailored to fast pixel buffers manipulation. In fact this is the most important and therefore most critical composition phase. Of course different multimedia applications could benefit from such hardware.

Silicon vendors adopt this strategy, as implementing custom accelerators is quite common nowadays for audio and video decoding. Likewise, raw image manipulators are included in most recent multimedia-oriented embedded platforms. For instance, the STU8500 board includes a B2R2 component (section 4.2).

Introducing this kind of hardware means that compositing architecture should be modified in some way, in order to use and to exploit acceleration to the maximum. One should note that a “vanilla” X server release cannot work with these devices.

At the least two different approaches are possible to optimize a CM and port it on such machines:

- Creating a custom implementation of the chosen composer, which can program the hardware, thus entirely suppressing interaction with the X server.
- Changing the X server implementation, creating a custom X server able to acknowledge accelerator presence.

The first approach, unfortunately, has a major, critical drawback. Programming the hardware directly means that support from the server would be dropped entirely, so rendering operations have to be carried out directly by the composer and manually coded by the programmer. Apart from software complexity issues, this approach would imply the necessity to copy all the pixmap, stored on server side, in client memory space due to lacks of server side manipulation. Thus, a huge drop in performance is expected.

It is therefore necessary to adopt the second approach, which would leave the CM unchanged. There are different ways to modify an X server implementation to suit embedded compositing needs. One could, for example, change the implementation of the “Render” extension, ensuring that composition calls are translated into calls of an appropriate interfacing library to the accelerator. Another possibility is to introduce new X server device plugins. A more detailed discussion on this topic is beyond the scope of this article.

4. REAL CASE ANALYSIS: MEEGO COMPOSER ON STU8500

This section gives a brief overview of a typical configuration in which an embedded CM (MeeGo Composer in this case) is executed on an embedded system. STU8500 from ST-Ericsson is chosen due to its multimedia capability and the presence of a specific hardware accelerator.

4.1 MeeGo Composer Architecture

MeeGo [Haddad 2010] is a soon-to-be-released open source OS developed jointly by Nokia and Intel and based on Linux kernel. The most important feature in MeeGo is that it is designed to be available on several different platforms. There will be at least three different releases: mobile, netbook and automotive. Each one of these versions retains the same core code (including the kernel), while the user interface (User eXperience) differs. MeeGo general architecture includes a CM which is part of every UX released so far. In what follows, only the mobile UX and its MeeGo Composer (MC) [Gitorious 2011] will be considered.

MC is based on a software architecture which is quite interesting and innovative, because of its use of the Qt “scene-item-view” abstraction to manage and render composited window content. According to this model, data and its representation have to be strictly separated. In this case data is the window content, while the representation is what the user views on the screen. Implementing the “scene-item-view” paradigm means that programmers job is easier, and window content management is faster [Trolltech 2010].

MC uses Qt libraries and more specifically MeeGo Touch libraries. MC presents common embedded managers features: one window at the time only, scaled-down graphical effects, hardware accelerated rendering only. MC requires some OpenGL ES 2 functionalities.

4.2 STU8500

The STU8500, a product by ST-Ericsson, is a “high performance, cost optimized smartphone platform”, expressly tweaked to improve multimedia performance [Gallo 2010]. The board is equipped with a Mali accelerator that fully supports OpenGL ES 2. Also, STU8500 mounts a series of specific accelerators. Among these is included a B2R2 (blit, blend, resize, rotate) component which is meant to provide fast pixel matrix manipulation and storage, in order to accelerate the compositing process. ST-Ericsson OS implementation adopts the second approach described in section 3.2, so a modified X.org release is included.

Of course, this is an ad-hoc solution, targeted for one specific machine, but preliminary testing confirms great compositing performance improvement when compared to graphics card acceleration only. B2R2 programming has some problems though, as the accelerator can only access physical memory and its driver interface it's quite low-level. MeeGo UX integrates smoothly with St-provided core OS, and MeeGo composer code doesn't need to be modified thanks to the chosen approach.

5. CONCLUSION

Compositing managers are quickly becoming standard in embedded operating systems. However, obtaining reasonable performances in general configurations is still a problem, as a lot of work is still needed on ad-hoc basis. To this regard, there are some open problems that have still to be resolved, among which:

- It is not currently feasible to build a software architecture capable of exploiting a general hardware accelerator meant to optimize composition, and not only a specific one at the time;
- In relation to the above problem, it is not clear whether it would be profitable to use different accelerators in parallel: the amount of parallelism available in graphics to be exploited by the compositing architecture has to be assessed more clearly.

REFERENCES

- Gallo A., 2010, *3G and multimedia platform - Linux activities on U8500*, St-Ericsson Technical Report, viewed 12/12/2010, <<http://www.stericsson.com/platforms/U8500.jsp>>
- Gettys, J., 2004. The Re-Architecture of the X Window System. *Proceedings of the Linux Symposium*, pp. 227-238.
- Haddad I., 2010, *Introduction to the MeeGo project*. The Linux Journal, viewed 28/12/2010, <<http://www.linuxjournal.com/article/10797>>
- Larsson A. and Klintstrom A., 2006, *Visual Effect Managment in a Mobile User Interface*. Master's Thesis, Umea University, viewed 12/12/2010, <<http://www8.cs.umu.se/education/examina/Rapporter/LarssonKlintstrom.pdf>>
- Leech J., 2005, *OpenGL graphics with the X window system (version 1.4)*, OpenGL.org, viewed on 12/12/2010, <<http://www.opengl.org/documentation/specs/glx/glx1.4.pdf>>.
- Matchbox Project, 2011, *X not on the Desktop!*, Matchbox-Project.org, viewed 20/01/2011, <<http://matchbox-project.org>>
- Gitorious 2011, *MeeGo Touch Framework source code*, Gitorious.org, viewed 05/10/2010, <<http://www.meego.gitorious.org/meegotouch>>
- Nilsson J., Reveman D., 2004, Glitz: Hardware Accelerated Image Compositing using OpenGL, *Proceedings of Usenix 04 Technical Conference*, pp. 29-40
- Packard K, 2009. The X Rendering Extension protocol definition. FreeDesktop.org, viewed on 12 December 2010, <<http://webcvs.freedesktop.org/xlibs/Render/protocol?revision=1.9>>.
- Paul, R., 2005. *Innovations in window management*, Linux.com archive, viewed 20/01/2011, <<http://www.linux.com/archive/feed/41869>>.
- Porter T. and Duff T., 1984, Compositing Digital Images. *Computer Graphics*, 18(3), pp. 253-259.
- Trolltech 2010, *The Graphics View Framework*, Qt Documentation, viewed 10/01/2011, <<http://doc.trolltech.com/4.6/graphicsview.html>>
- Wu C., 2009, *Introduction to Android Window System*, Debian.org, viewed 28/12/2010, <<http://people.debian.org.tw/~olv/surfaceflinger/surfaceflinger.pdf>>

A COMPARISON OF OPTICAL FLOW TECHNIQUES FOR A BLIND SPOT CAMERA

Lorenz Van den Heuvel, Kristof Van Beeck and Toon Goedemé
Lessius Mechelen, Association K.U. Leuven
Jan De Nayerlaan 5, 2860 Sint-Katelijne-Waver, Belgium

ABSTRACT

In this paper we present an overview and comparison of the most important optical flow calculation techniques. Our comparison is based upon three aspects: accuracy, speed and robustness against noise. Based on the results of our comparison, we determine which of the algorithms is best suited for real-time implementation on dedicated hardware. This decision is based on the trade-off that we found between our evaluation aspects. In the future we will use this result to achieve the automatic detection of vulnerable road users using a blind spot camera.

KEYWORDS

Computer vision, object recognition, optical flow, quantitative comparison.

1. INTRODUCTION

The work in this paper is part of an ongoing research project that aims on reducing accidents with trucks caused by the so-called blind spot area (figure 1(a)). Using a blind spot camera, represented in figure 1(b), our goal is the automatic detection of vulnerable road users. Therefore, this research focuses on object recognition under harsh conditions, meaning that the objects of interest, the pedestrians and other road users, as well as the camera are moving and that the time for recognition of these objects is extremely short. The problem that emerges is that simple object recognition techniques, such as background subtraction, need a constant background. We need a more advanced approach for object recognition. The chosen approach is independent motion segmentation based on optical flow. Motion segmentation is the partitioning of an image into regions of the same pixel motion. The goal of this paper is to get an in-depth look at optical flow techniques and to choose the best suited technique for real-time implementation on dedicated hardware such as a Field Programmable Gate Array. We take a close look on which techniques exist, and what their advantages and disadvantages are. Based on these results, we select the optimal optical flow algorithm that suits our application. This decision is based upon the techniques ability to generate optical flow in *real-time* and its capability for *independent motion segmentation*. The final goal of this work is to build a real-time demonstrator that can detect pedestrians and other vulnerable road-users using optical flow and independent motion segmentation. In section 2 we give an introduction of optical flow, followed by the comparison of optical flow techniques in section 3. In section 4 we state our conclusion from this study.



Figure 1. (a) Blind spot of a truck. (b) Blind spot camera mounted on truck. (c) And (d) are test frames of the blind spot camera.

2. OPTICAL FLOW

The goal of this study is to look at the different techniques that exist to calculate the optical flow. We look for the best technique that is most suitable for real-time implementation in dedicated hardware with respect to the accuracy and speed, and that can be adequately adapted to perform independent motion segmentation.

Optical flow is a vector representation of the movement of each pixel in two successive frames. Figure 2(a) and 2(b) are two frames of the Yosemite sequence. Figure 2(c) represents the result of an optical flow algorithm on this sequence.

Optical flow techniques can be categorized into 4 groups (Barron et al. (1994)):

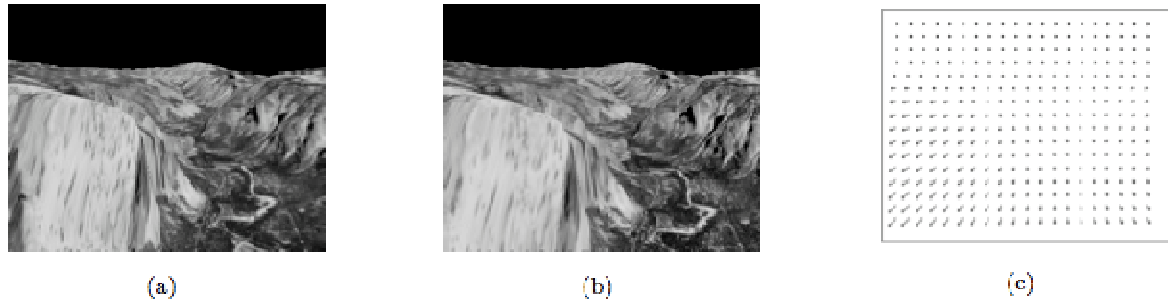


Figure 2. (a) And (b) are 2 frames from the Yosemite sequence. (c) Is the optical flow field for (a) and (b).

Differential techniques are based upon partial differentials of the intensity in the spatial and time domain. Differential techniques themselves are then again divided into three groups: local, global and combined techniques. Local techniques, such as Lucas & Kanade (1981), try to maximize a local energy expression while global techniques, like Horn and Schunk (1981), try to minimize a global energy expression. The key assumption that is made when using differential techniques is *brightness constancy*. This means that the intensity of the pixels from 2 successive frames may not change, represented by equation 1. Where $I(x,y,t)$ is the intensity of a pixel on x,y in the spatial domain and t in the time domain.

$$I(x, y, t) = I(x + dx, y + dy, t + dt) \quad (1)$$

Correlation techniques compare a *window* from frame 1 with a *window* from frame 2. A window is a neighborhood of pixels. They search for the best match in correlation between two windows, using a correlation criterion. Examples of such correlation criteria are SSD (Sum of Squared Differences) and SAD (Sum of Absolute Differences). There are two parameters that are important when using correlation techniques: firstly the size of the window and secondly the size of the displacement of the window. Some well-known correlation techniques are introduced by Anadan (1987) and Camus (1997).

Energy techniques use velocity-tuned filters. These filters are adjustable for specific frequency ranges. This means that filters respond with a high (energy) output when they perceive frequencies for which they are sensitive. This output is used to calculate the optical flow field. An example of an energy algorithm is Heeger (1988).

Phase techniques use band-pass velocity-tuned filters to calculate optical flow. Band-pass filters are used to separate the input image accordingly to scale, speed and orientation. Waxman, Fleet and Jepson (1990) are examples of phase-based techniques.

3. COMPARISON OF OPTICAL FLOW TECHNIQUES

The comparison of the optical flow techniques in this section is separated into three parts. Each part presents the performance of the different optical flow techniques based on a specific characteristic: accuracy, speed and robustness against noise. These results are obtained by an in-depth literature study that we performed, and thus summarizes results from several authors (Barron et al (1994), H.Top (2007), Lui et al (1996)).

Table 1. Summary of results from the Yosemite sequence

Technique	Average Error	Standard deviation	Density
Differential			
Horn & Schunck	32.43°	30.28°	100%
Lucas & Kanade ($\lambda_2 \geq 1.0$)	4.10°	9.58°	35.1%
Uras et al. ($\det(H) > 1.0$)	6.73°	16.01°	14.7%
Nagel	11.71°	10.59°	100%
Correlation			
Anadan	15.84°	13.46°	100%
Singh (Step 2, n = 2, w = 2)	13.16°	12.07°	100%
Energy			
Heegher (combined)	11.74°	19.04°	44.8%
Phase			
Waxman et al. ($\sigma_f = 2.0$)	20.32°	20.60°	7.4%
Fleet and Jepson ($\tau = 1.25$)	4.95°	12.39°	30.6%

3.1 Accuracy

The accuracy of an optical flow algorithm is measured by the average angular error and the density it produces. The angular error is calculated as stated in equation 2. In table 1 the results are shown of tests conducted by Barron et al. (1994).

$$= \dots \dots \dots (2)$$

From these results we can conclude some interesting facts. Firstly there is a noticeable difference between global and local differential techniques. Global techniques deliver good density but a large average error, the opposite is true for local techniques. Secondly, correlation-based techniques have an average error that is reasonable while having dense fields. Thirdly, the energy-based techniques deliver poor results both on average error and density. Finally we can determine that the phase-based techniques deliver the best results with respect to the average error and density, therefore these phased-based techniques are the most interesting for accurate optical flow.

3.2 Speed

Lui et al. (1996) did research into the speed aspect of optical flow techniques. They displayed the techniques in Accuracy-Efficiency graphs (AE graphs), where they plotted the error in degrees on the x-axis and the efficiency in seconds/frame on the y-axis. Figure 3 represents the different types of optical flow techniques in an AE graph. We see that the phase-based techniques, which delivered outstanding results in the tests conducted by Barron et al. (1994), here also deliver good accuracy but have a very bad time-complexity. This means that the time needed to calculate the optical flow is very large. Differential techniques deliver better time results but also have a larger error rate. Correlation techniques deliver the best time-complexity of them all, with accuracy varying accordingly to the size of the window used. From these results we can conclude that the optical flow techniques with the best time-complexity are the correlation-based techniques followed by the differential-based techniques.

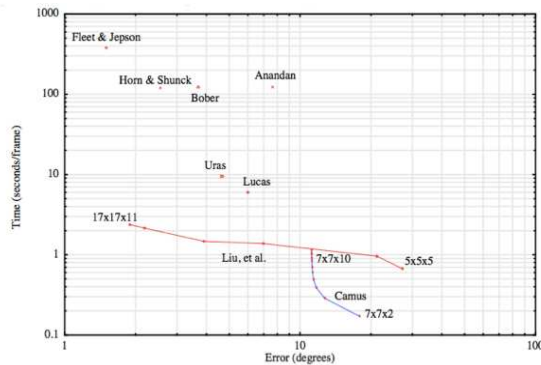


Figure 3. 2-D performance diagram for Diverging Trees sequence.

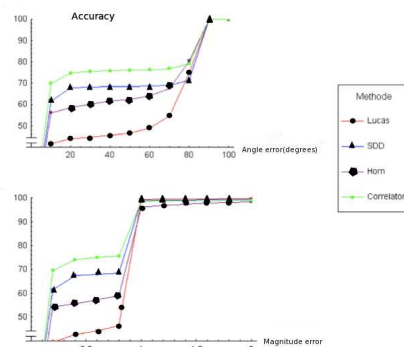


Figure 4. Influence of noise on the performance.

3.3 Robustness against Noise

A very important factor at which we also looked at is the robustness against noise. In many real-life situations variations in light conditions may occur, which result in a change of brightness. Figure 4, from H.Top (2007), represents the performance of the different techniques after introduction of noise. The results reveal that the differential-based techniques are the worst of them all. This results in the usage of brightness constancy as mentioned in section 2. The correlation and energy-based techniques are not affected by noise or the change in brightness because they don't rely on brightness constancy.

4. CONCLUSIONS

After reviewing the result of section 3, we can come to some very interesting conclusions. There is an obvious difference between techniques that deliver good accuracy and those that deliver good time-complexity. The accuracy declines with increasing speed. It has also the ability of being very robust under different circumstances. Energy and phase-based techniques have very good accuracy, but very bad time complexity. As we are looking for optical flow techniques that could be implemented in real-time, energy and phase-based technique are not suited for our goal. The choice remains between differential and correlation-based techniques. Differential based techniques are less accurate than energy/phase-based techniques but are more accurate than correlation-based techniques. A major drawback is that they are sensitive to noise and brightness variation. They are usually implemented with a pre-smoothing step to resolve the noise factor. Correlation techniques have the advantage that they are more robust in real-life situations, only use integer computation and are not affected by the brightness constancy factor. This makes correlation-based techniques the ideal choice to use for dedicated hardware implementation. We chose specifically for the Camus correlation technique because it has a very good time-complexity, as we have seen in section 3, and acceptable accuracy.

In our future work we plan to implement the Camus (1997) correlation technique on an FPGA and build a real-time demonstrator that will detect vulnerable road-users using independent motion segmentation.

We built a test set-up with a blind spot camera on a Volvo truck and recorded 8000 frames. Figure 1(c) and (d) are examples of such test frames.

REFERENCES

- Barron et al., 1994. *Performance of optical flow techniques*. Int. Jour. of computer vision, Vol. 2, No. 1, pp 43-77, Springer.
- Fleet, D.J. and Jepson, A.D., 1990. *Computation of component image velocity from local phase information*, International Journal of Computer Vision, Vol. 5, No.1, pp 77-104
- Heeger, D.J., 1988. *Optical flow using spatiotemporal filters*. Int. Jour. of Computer Vision, Vol. 17, No.4, pp 279-302
- Horn, B.K.P. and Schunck, B.G., 1981. *Determining optical flow*, Artificial intelligence, Vol. 17, No. 1-3, pp 185-203.
- H.Top., 2007. *Optical flow en bewegingillusies*. University of Groningen. Faculty of Mathematics & Natural Sciences
- Lui et al., 1996. *Accuracy vs. efficiency trade-offs in optical flow algorithms*. Computer Vision-ECCV'96 ,pp 174-183.
- Lucas, B.D. and Kanade, T., 1981. *An iterative image registration technique with an application to stereo vision*, International Joint Conference on Artificial Intelligence, Vol. 3, 674-679, Citeseer
- P.Anadan, 1987. *Measuring Visual Motion From Image Sequences*, PhD dissertation and COINS Technical Report 87-21, University of Massachusetts
- T. Camus, 1997. *Real-time quantized optical flow*, J. Real-Time Imaging 3, 71-86.

Posters

A FLEXIBLE FRAMEWORK FOR MULTI-VOLUME DATA VISUALIZATION

Nikolay Gavrilov, Alexandra Belokamenskaya, and Vadim Turlapov

*Lobachevsky state university of Nizhny Novgorod
603950, N. Novgorod, pr. Gagarina, 23*

ABSTRACT

In this paper we present a framework for visualization of three-dimensional multiple volumetric datasets. We use such optimization strategies for GPU-based raycasting as the early ray termination and empty space skipping. An SMV (Stereo Multi-volume Viewer) provides such visualization techniques as Direct Volume Rendering via 1D- or 2D- transfer functions, multiple semi-transparent discrete isosurfaces, MIP, MIDA, etc. We also use random ray start position generation and further frames accumulation in order to reduce visual artifacts. The quality can be also improved by GPU-based volumetric tri-cubic up-sampling of the source datasets or by on-the-fly tri-cubic filtering during the rendering process. In addition to the clipping bounding box, user can use custom bounding mesh for more accurate region-of-interest selection. The SMV also supports 4 different stereoscopic visualization modes. We outlined the visualization performance in terms of frame rates for different visualization techniques on several graphic cards.

KEYWORDS

GPGPU, raycasting, multi-volume rendering, medical visualization.

1. INTRODUCTION

In scientific visualization it is often necessary to deal with volumetric scalar datasets. These data may be obtained by some numerical simulation or via scanning equipment such as a tomographic scanner. The output data of the CT scanning is a set of slices, i.e. two-dimensional scalar arrays. The stack of such slices can be interpreted as a volumetric dataset which can be visualized as a 3D object.

Since the 90s, the Direct Volume Rendering shows itself as an efficient tool for visual analysis of volumetric data (Lundström C., 2007). Different established approaches (Klaus E. et al, 2004) make possible the implementation of real-time volume rendering via the parallel and high-performance computations on a GPU. The recent progress in GPGPU computations makes the real-time multi-volume rendering possible (Kainz B. et al, 2009). In this paper we present a new framework SMV (Stereo Multi-volume Viewer) for visual exploration of the multi-volume data.

2. METHODS AND ALGORITHMS

2.1 Rendering Methods

Due to the high flexibility of the raycasting volume rendering method there is a huge amount of different possible data visualization techniques. The Maximum Intensity Projection (MIP) is one of the most popular volume visualization techniques in medical imaging (Geoffrey D., 2000). It is easy for clinicians to interpret an MIP image of blood vessels. By analogy the Minimum Intensity Projection (MinIP) may be used for bronchial tube exploration. The Direct Volume Rendering (DVR) technique is also used in medical survey, but its usage is more limited (Lundström C., 2007). However the using multi-dimensional Transfer Function (TFs) may help for exploration of unknown to science phenomena (Gordon L., 1999).

The SMV supports multiple outputs for several rendering techniques, i.e. a user can see the visualization via several rendering techniques simultaneously. Each output viewport contains its own rendering settings, like the TF, isovalues, colors for isosurfaces, window center and width, widgets for the two-dimensional TF, etc.

2.2 Optimization Strategies

The early ray termination is a common optimization strategy for a raycasting technique. Because of the growing flexibility of the GPU shading programs it is possible to terminate the raycasting algorithm for each individual ray if the accumulated opaqueness is close to 1. However, in rendering techniques like the MIP it is necessary to browse the whole ray path until leaving the bounding box.

The empty space skipping approaches are used extensively in CPU-based DVR (Grimm S. et al, 2004) however some techniques can be extended to GPU-implementation. There are two levels of volumetric data hierarchy in the SMV: the source dataset, e.g. of size 512x512x512, and the acceleration structure, i.e. a small two-channel volumetric dataset of size 16x16x16. So each 32x32x32 block in the source data corresponds to one cell from the acceleration structure. Each of these cells contains minimum and maximum from the corresponding block in the source data. So for each block it is known, whether the corresponding space is transparent and may be skipped by a ray. This approach accelerates visualization up to 1.5-2 times.

Custom bounding polygonal mesh can be used either for rendering acceleration or for data clipping, which is an alternative to the segmentation. We use frame buffers to store distances from the viewpoint to the mesh front and back faces. Buffers may contain 4 ray path segments so the mesh is not to be convex.

2.3 Rendering Artifacts Reduction

Because of the finite ray step count, the rays may skip meaningful features in the dataset even if the ray step is much less than the voxel size. As a result 'wood-like' image artifacts may appear. This wood-likeness appears because the rays start from the same plane (i.e. the bounding box face). This artifacts' regularity can be removed by randomization of the ray start positions. The final image will contain 'noisy' artifacts instead of 'regular' ones. If a user doesn't change viewpoint and other visualization settings, these frames can be accumulated in such a way that the user will see an average image that contain no noise.

The visualization quality can be improved via the tri-cubic filtering instead of the common tri-linear filtering. In order to make a single tri-cubic sampling it is necessary to make 8 tri-linear samplings from the same dataset (Daniel R. et al, 2008), so it is costly to use this on-the-fly filtering in the raycasting algorithm. However, if the dataset size is not huge (e.g. 256³) or if a rendering technique is simple enough (e.g. Direct Volume Rendering, opaque iso-surface) the visualization will be in real-time, i.e. over 20-30 fps.

2.4. Data Processing Algorithms

In order to improve visualization quality we have implemented a data resampling tool. A user can select any area from the data via the bounding box and resample this selection in another resolution. By default the resulting data is a resampled copy of the selected area in the source data. This processing is also implemented via the GLSL shaders, so the fragment shader defines the volumetric data processing algorithm. In the SMV there is an option to make additional operations with the data, like the Gaussian smoothing or median filtering. There are also algorithms for generating other fields derived from the source data (e.g., the gradient magnitude field).

3. VISUALIZATION PERFORMANCE RESULTS

In table 1 we outlined the SMV volume rendering performance obtained on different graphic cards. We used a full screen resolution (1280x1024), 16-bit dataset of size 512x512x512, the ray step 0.0004 (i.e. 0.2 of the voxel size), and the viewpoint shown in Figure 1a.

Table 1. Visualization performance for different rendering techniques in terms of frame rates (fps). *Isos* means visualization of three semitransparent isosurfaces

GPU	DVR + Isos	Isos	MIP	DVR	Opaque iso-surface
NVIDIA GeForce GTS 250	9	10	10	12	30
NVIDIA GeForce 9500 GT	4	4	4	5	13
NVIDIA Quadro FX 5600	14	16	19	23	63
ATI RADEON HD 4870	18	21	35	38	81
ATI RADEON HD 4890	21	25	39	44	108

In general ATI cards overcome NVIDIA cards for the raycasting task. This can be explained by the fact, that the ATI cards have vector architecture, while raycasting algorithms contain many vector operations.

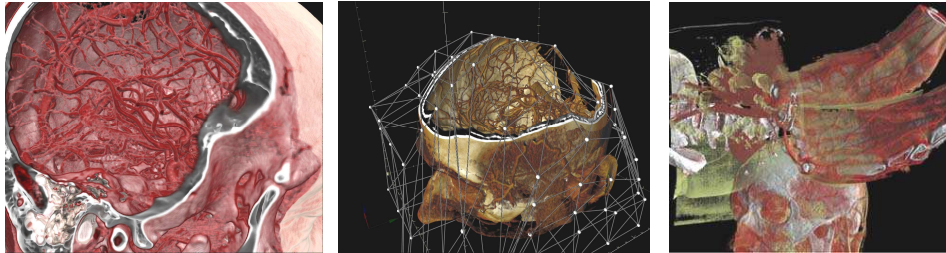


Figure 1. Left to right: a) Brain blood vessels. Test viewpoint. b) Volumetric data clipping via the custom polygonal mesh. c) Multi-volume rendering of 3 datasets of sizes 512^3 each.

4. CONCLUSION

The proposed framework can be used for multi-modal medical data exploration. E.g., a user can load datasets of the CT and MRI modalities as DICOM slices and properly place them together in the space via the control points, so that the user sees the comprehensive picture of the medical research. The SMV is able to visualize up to 4 datasets together. The framework is good for stereo demonstrations via the stereo-projector, anaglyph, interlaced rendering, or with shutter glasses via the 3D Vision technology. Owing to the framework's flexibility it is easy to add new rendering techniques and data processing algorithms by creating new fragment shader source files, following defined interfaces (i.e. uniform variables). Proposed framework is good either for the volumetric data visualization or for new raycasting and data processing algorithms investigation.

This work was supported by the Federal Program "Research and Research and teaching staff of Innovative Russia", State Contract No. 02.740.11.0839.

REFERENCES

- Klaus E. et al, 2004; *Real-Time Volume Graphics*, A.K. Peters, New York, USA.
- Kainz B. et al, 2009. Ray Casting of Multiple Volumetric Datasets with Polyhedral Boundaries on Manycore GPUs. *Proceedings of ACM SIGGRAPH Asia 2009*, Volume 28, No 152.
- Daniel R. et al, 2008. Efficient GPU-Based Texture Interpolation using Uniform B-Splines. *In IEEE Transactions on Journal of Graphics, GPU, & Game Tools*, Vol. 13, No. 4, pp 61-69.
- Lundström C., 2007. *Efficient Medical Volume Visualization: An Approach Based on Domain Knowledge*. Linköping Studies in Science and Technology. Dissertations, 0345-7524 ; No. 1125.
- Geoffrey D., 2000. Data explosion: the challenge of multidetector-row CT. *In IEEE Transactions on European Journal of Radiology*. Vol. 36, Issue 2, pp 74-80.
- Gordon L., 1999. *Semi-automatic generation of transfer functions for direct volume rendering*. A Thesis Presented to the Faculty of the Graduate School of Cornell University in Partial Fulfillment of the Requirements for the Degree of Master of Science.
- Grimm S. et al, 2004. Memory Efficient Acceleration Structures and Techniques for CPU-based Volume Raycasting of Large Data. *Proceedings of the IEEE Symposium on Volume Visualization and Graphics 2004*. pp 1 – 8.

EDGE-DIRECTED IMAGE UPSAMPLING METHOD BASED ON DIRECTIONAL KERNEL INTERPOLATION

Jungho Yoon¹ and Yeon Ju Lee²

¹*Department of Mathematics, Ewha W. University, Seoul, S.Korea*

²*Department of Mathematical Sciences, KAIST, Daejeon, S.Korea*

ABSTRACT

This paper summarizes the edge-directed image upsampling method based on radial basis function (RBF) interpolation [5] and discusses a new kernel based interpolation method. The resampling evaluation is determined according to the edge orientation. The proposed scheme is as simple to implement as linear methods but it demonstrates improved visual quality by preserving the edge features better than the classical linear interpolation methods. The algorithm is compared with some well-known linear schemes as well as recently developed nonlinear schemes.

KEYWORDS

Radial basis function, Kernel-based scheme, edge-directed interpolation, image upsampling

1. INTRODUCTION

Image upsampling to a higher resolution image is a prime technique in image processing. It has applications in medical imaging, remote sensing and digital photographs. The standard procedure for image upsampling can be regarded as a two-step operation. First, the discrete data are interpolated into a continuous function, and then resampled at a finer sampling rate as desired.

Traditionally, image interpolation has been accomplished through convolution of the image samples with a single kernel typically, pixel replication, bilinear, bicubic and splines [1]. Usually these methods either tend to smooth edges or produce blocky interpolated images with staircase edges, thus providing blurred images. The various nonlinear interpolation methods proposed to improve the quality of upsampled images while preserving the edges include the explicit edge-directed interpolation methods. Instead of explicit edge-directed methods, implicit edge-directed image interpolation methods have been studied [2,3,4]. In this paper, we first revisit the implicit edge-directed upsampling method based on radial basis function (RBF) interpolation. The RBF-based scheme produces high quality images, it does have limitations in angle resolution. Thus, we suggest a kernel based interpolation method.

2. NEW DIRECTIONAL KERNEL-BASED INTERPOLATION

Let the function $f: D \rightarrow \mathbb{R}$ be an ideal continuous image, and assume that our observed image is a discrete sampling of f at equally spaced points in a rectangle $D \subset \mathbb{R}^2$. The primary goal of this section is to introduce edge-preserving nonlinear image upsampling algorithm based on the interpolation method by using the Gaussian function $\varphi(\mathbf{x}) := \varphi_\lambda(\mathbf{x}) := \exp(-|\mathbf{x}|^2 / \lambda^2)$ [5].

2.1 Directional Gaussian RBF Interpolation

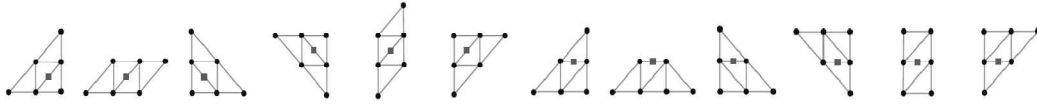


Figure 1. (a) Stencils for the face points

(b) Stencils for the align points

Let \mathbf{p} be a resampling point and $S_j := S_j(\mathbf{p}) := \{\mathbf{x}_n : n = 1, \dots, 6\}, j = 1, \dots, 6$ be a stencil around \mathbf{p} of different direction. The RBF interpolation $G_{S_j} f (j = 1, \dots, 6)$ in to the discrete image data $f(\mathbf{x}_n), \mathbf{x}_n \in S_j$, can be written in the form of Lagrange-type:

$$G_{S_j} f(\mathbf{x}) := \sum_{n=1}^6 u_n(\mathbf{x}) f(\mathbf{x}_n)$$

The interpolation kernel $\mathbf{u}_j(\mathbf{x}) := (u_n(\mathbf{x}) : n = 1, \dots, 6)$ associated with S_j can be obtained by solving the following linear system

$$\mathbf{A}_j \cdot \mathbf{u}_j(\mathbf{x}) = \mathbf{R}_j(\mathbf{x}),$$

with

$$\mathbf{R}_j(\mathbf{x}) := (\varphi(\mathbf{x} - \mathbf{x}_j) : \mathbf{x}_j \in S_j), \mathbf{A}_j := (\varphi(\mathbf{x}_i - \mathbf{x}_j) : \mathbf{x}_i, \mathbf{x}_j \in S_j).$$

For the construction of interpolation kernels, polynomials up to degree 2 can be used instead of the RBFs. However, the matrices generated by the polynomial interpolation are not solvable on the stencils on some rectangular stencil. Here, we employ the notion of the (double) curvature of a surface $g(\mathbf{x})$, defined by

$$\kappa_g(\mathbf{x}) := (g_{xx}(\mathbf{x})g_{yy}(\mathbf{x}) - g_{xy}(\mathbf{x})^2) / (g_x(\mathbf{x})^2 + g_y(\mathbf{x})^2 + 1)^2$$

For each Gaussian interpolant $G_{S_j} f (j = 1, \dots, 6)$, the corresponding local curvatures $\kappa_{G_{S_j} f}(\mathbf{p})$ are calculated at the point \mathbf{p} . Then, we choose an interpolant $G_{S_j} f$ whose curvature is smaller than the other ones.

2.2 Directional Kernel Interpolation

We now introduce kernel based interpolation method which can be expressed in the form:

$$Lf(\mathbf{x}) := \sum_{\mathbf{x}_n \in S} K(\mathbf{x}, \mathbf{x}_n) f(\mathbf{x}_n)$$

where S the local stencil around \mathbf{x} and $K(\mathbf{x}, \mathbf{x}_n)$ is anisotropic kernel considering the edge-direction. The kernel $K(\mathbf{x}, \mathbf{x}_n)$ is supposed to reproduce a certain set of basis functions. It guarantees the accuracy of the scheme. Good examples are the kernels of subdivision scheme, edge-directed Lagrange polynomials and generalized moving least squares method. In order to implement the interpolation Lf , we first approximate the gradient vector of the image at the upsampling position \mathbf{x} . According to this direction, the kernel $K(\mathbf{x}, \mathbf{x}_n)$ is implemented in a way of considering edge-direction.

3. EXPERIMENTAL RESULTS

The results of the proposed upsampling algorithm are compared with those from the traditional C^2 -bicubic interpolation method [1] and a recently developed nonlinear schemes, the new edge-directed interpolation (NEDI) [2]. Figs.2 include a comparison of the portions of the magnified images by different algorithms. The C^2 -bicubic interpolation method is inferior to the other (edge-directed) methods. The proposed method is using the kernel generated by an adapted moving least squares method. We see that it provides higher quality images than the other non-linear methods.

4. CONCLUSION

Although the Gaussian-based scheme produces high quality images, it does have limitations in angle resolution. To overcome this limitation, we propose a kernel-based scheme. In averaging process, the kernels are implemented in a way of considering edge-direction. Subdivision schemes or by adapted local least squares method with 'locally-adapted' penalty function would be good tools to construct the interpolation kernels. In the near future, a detailed algorithm will be introduced along with some numerical examples.

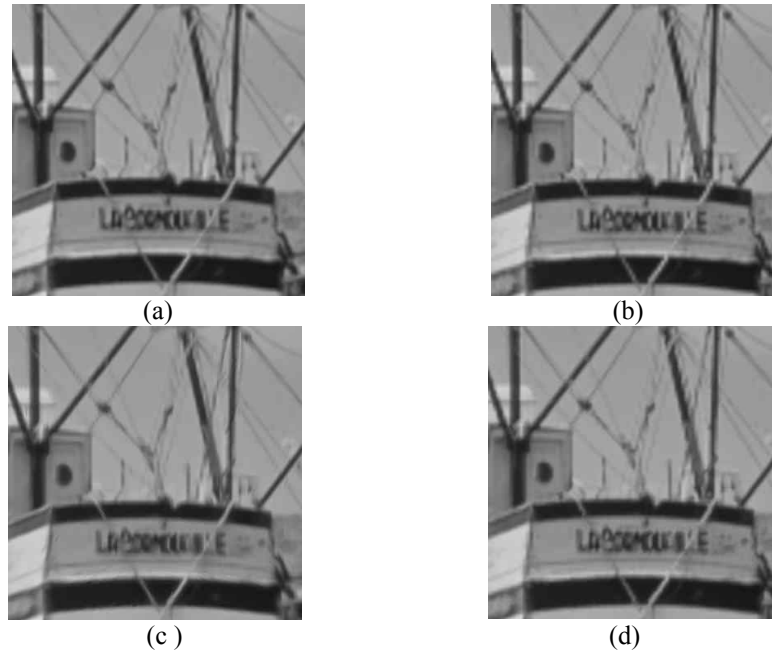


Figure 2. (a) the proposed, (b) Bicubic interpolation, (c) NEDI method in [2], and (d) Gaussian RBF method.

REFERENCES

- [1] R. G. KEYS, 1981, Cubic convolution interpolation for digital image processing, *IEEE Trans. Acoust. Speech Signal Processing*, vol. 29, no. 6, 1153-1160.
- [2] X. LI AND T. ORCHARD, 2001, New edge directed interpolation, *IEEE Tran. Image. Proc.*, vol. 10, no. 10, 1521-1527.
- [3] Y. CHA, AND S. KIM, 2007, The error-amended sharp edge (EASE) scheme for image zooming, *IEEE Tran. Image. Proc.*, vol. 16, no. 6, 1496-1505.
- [4] X. ZHANG AND X. WU, 2008, Image interpolation by adaptive 2-D autoregressive modeling and soft-decision estimation, *IEEE Tran. Image. Proc.*, vol. 17, no. 6, 887-896.
- [5] Y. LEE AND J. YOON, 2010, Non-linear image zooming upsampling method based on radial basis function interpolation, *IEEE Tran. Image. Proc.*, vol. 19, no 10, 2682 – 2692.
- [6] X. PANG AND M. PANG, 2009, Adaptively up-sampling point-sampled models, *NIMES'09 International conference*, vol.1, 314-318.
- [7] M. ALEXA, et al, 2003, Computing and rendering point set surfaces, *IEEE TVCG.*, vol. 9, no 1, 3-15

HOMOGRAPHY USING TOPOLOGICAL INFORMATION FROM SCATTERED FEATURE POINTS

Rishu Gupta and Lee Byung Gook
*Dept. of Visual Contents, Dongseo University,
San69-1, Jurye2-Dong, Sasang-Gu, Busan, 617-716, Korea*

ABSTRACT

This paper proposes a method for finding homography between the left and the right images using correspondence points or the matching points calculated by topological information of feature points, which are extracted with the help of Harris corner detection algorithm. Delaunay triangulation is used to get information about scattered feature points and the simplex method is introduced to calculate neighborhood points of competent edges extracted from Delaunay triangulation. The competent edges are extracted using edge length as a parameter. In the final step, normalized correlation is performed by taking gray values of the image into account. The gray values of vertices that are selected using simplex method are taken and the normalized correlation is performed of vertices from the first image to the vertices of second image. The most suitable points are taken as correspondence points. And in the last step homography matrix is computed using correspondence points calculated.

KEYWORDS

Homography, Correspondence Search, Delaunay triangulation, Simplex method, Normalized Correlation.

1. INTRODUCTION

Computational stereo vision for the extraction of three dimensional scene structure from multiple images of the same physical world has been an intense area of research for decades. The initial basis for stereo is that the same physical location projects to unique point in pair of images when observed from two different cameras. The biggest problem in multiple views is calculation of feature point and finding homologous or correspondence points. The process of finding location of same physical world point from the stereo image is termed as correspondence search. The block matching method [1] uses Normalized cross correlation (NCC), Sum of squared differences (SSD), Sum of absolute differences (SAD). Feature matching method is proposed in which search is performed using reliable features in the images like edges [2]. In the proposed paper topological information from scattered feature point is used to compute the correspondence points. The topological information is extracted using one of historical concept Delaunay triangulation which is unique for every scattered feature point. Edge length parameter is used to select some of the competent edges for computing correspondence points. The contents of the paper are as follows. In section 2, some of Correspondence search algorithms have been introduced. In section 3, the proposed algorithm is explained. Experimental results and conclusion are discussed in section 4.

2. CORRESPONDENCE SEARCH ALGORITHM

The correspondence search can be local or global depending upon region and the space used for calculating identical pair. When we refer to only certain pixels around pixel in consideration the constraints are local. Similarly, when we consider constraint on scan line or the entire image then it is termed as global constraint. The local correspondence methods fall into three main categories: Block Matching [1], Gradient methods [9], and feature matching [2]. Both block matching and the gradient method are sensitive towards depth discontinuities and uniform texture region in images. Feature matching overcomes these problems to extract the reliable features by limiting area of consideration for computing the features. Global correspondence

method reduces sensitivity towards the local region that does not match by local correspondence method. Such constraint increases the computational complexity to a considerable extent. There are various categories for global correspondence method: Dynamic Programming [6], Graph Cuts [8], Nonlinear Diffusion, Belief Propagation, Correspondenceless method, Intrinsic Curves [4] etc. Many correspondence search methods are discussed in detail in [3].

3. PROPOSED METHOD

In proposed method for homography estimation the correspondence points are computed using topological information from scattered feature points. Further, Delaunay Triangulation of points provide information regarding the topology. Simplex method [7] helps in selection of points which are used for correlation to get the correspondence point and later homography.

3.1 Feature Point and Delaunay Triangulation

Feature point from two stereo images is computed using Harris corner detection algorithm [10] for corner feature extraction which can be seen in Figure 1. The Delaunay triangulation containing topological information of scattered feature points which will be exploited for computation of correspondence point is shown in Figure 2.

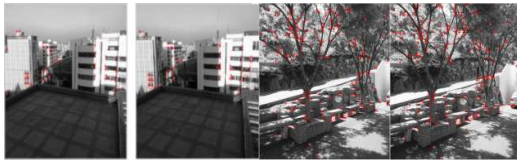


Figure 1. Feature points Extracted.

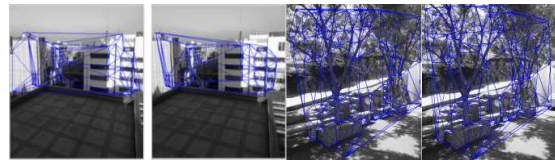


Figure 2. Delaunay Triangulation of the Feature points.

3.2 Competent Edge Extraction and Normalized Correlation

After Delaunay triangulation, length of randomly selected edge from first image Delaunay Triangulation and all the edges of second image Delaunay Triangulation is computed and stored. After extraction of one edge from the first image, difference of the edge length is performed with the second images calculated edge lengths, if difference between edge lengths is less than threshold value, edge is considered as the competent edge. Selecting boundary edges from both first and second image Delaunay Triangulation should be avoided.

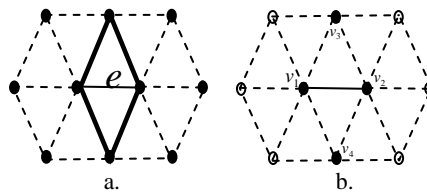


Figure 3. a).Up Simplex of the Selected Edge. b) Two Down-Simplex of the triangles on Selected Edge.

In next step, vertices of triangles on selected edge are computed with the help of simplex method [9]. The four vertices (v_1, v_2, v_3, v_4) of two triangles connected by an edge can be extracted using Up-Simplex one time followed by two Down-Simplex. v_1, v_2 are common vertices of two triangles. The Figure 3 shows two extracted triangles from the considered edge and four vertices from two linked triangles. In last step, the normalized correlation between gray values of four vertices obtained, using topology information from Delaunay Triangulation, is computed by taking a window of size $N \times N$ across the pixel, if pixel distance between matched edge from two images is not too large. The correlation matrix is created for every competent edge and the points having maximum correlation along both rows and columns are selected as

correspondence points. Normalized correlation of four points from the first image is calculated with every four selected point from edge of second image using (1).

$$cormat = \frac{\sum_{x=-N}^N \sum_{y=-N}^N I_1(i+x, j+y) * I_2(i+x, j+y)}{\sqrt{\sum_{x=-N}^N \sum_{y=-N}^N I_2(i+x, j+y) * I_2(i+x, j+y)}} \quad (1)$$

The points $I_1(i, j)$ and $I_2(i, j)$ are gray intensity values of vertices (v_1, v_2, v_3, v_4) from first and second image respectively, N is size of the window.

4. EXPERIMENTAL RESULT AND CONCLUSION

The algorithm was implemented using Matlab(R2010b), on the 32-bit operating system and Pentium(R) Dual Core processor with 2.60 GHz. The image taken was 500*373 pixels. The number of feature points and edges extracted in the left image and right image are 72, 83 points and 205, 237 edges in first pair and 171,185 points and 502,542 edges in second pair which can be seen in Figure 1 and Figure 2 respectively. The time taken to compute homography in first and second pair is 0.37sec and 0.53sec The threshold value to find the competent edge is taken as $Th = 0.03$. In Figure 4 result of calculated homography matrix [1] using correspondence points calculated by the proposed algorithm have been shown. In this paper images taken have gone translation and rotation but no scaling; scaling can be considered by taking scaling invariant triangle properties like ratio of edges and angles from triangulation. The algorithm can perform better with more stable feature points. In future, our efforts will be to improve accuracy and robustness of algorithm.

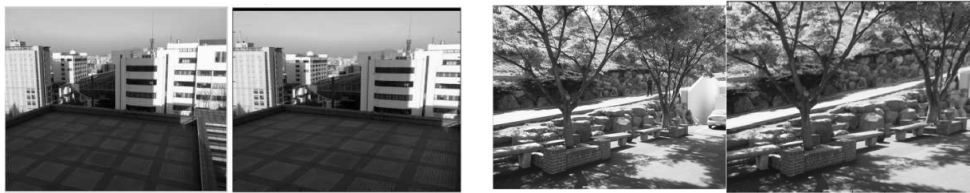


Figure 4. a) Right Image b) H^* (left image). H: homography matrix calculated using the points from above algorithm.

REFERENCES

1. R. Hartley and A. Zisserman, 2004. "Multiple View Geometry in Computer Vision". Cambridge University Press, ISBN: 0521540518, second edition.
2. V. Venkateswar and R. Chellappa, 1995. "Hierarchical Stereo and Motion Correspondence Using Feature Groupings," *Int'l J. Computer Vision*, vol. 15, pp. 245-269.
3. M.Z. Brown, D. Burschka, and G.D. Hager, 2003. "Advances in Computational Stereo," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, Vol. 25, No. 8, pp. 993-1008,.
4. C. Tomasi and R. Manduchi, 1998. "Stereo Matching as a Nearest-Neighbor Problem," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 20, pp. 333-340.
5. P. Aschwanden and W. Guggenbuhl, 1993. "Experimental Results from a Comparative Study on Correlation-Type Registration Algorithms," *Robust Computer Vision, Forstner and Ruwiedel*, eds., pp. 268-289, Wickmann,.
6. H.H. Baker, 1982. "Depth from Edge and Intensity Based Stereo," *Technical Report AIM-347, Artificial Intelligence Laboratory, Stanford Univ.*
7. Peter Lindstrom, Greg Turk, 1998. "Fast and Memory Efficient Polygonal Simplification," *IEEE_vis*, pp.279, *Ninth IEEE Visualization 1998 (VIS '98)*.
8. V. Kolmogorov and R. Zabih, 2001. "Computing Visual Correspondence with Occlusions Using Graph Cuts," *Proc. Int'l Conf. Computer Vision*.
9. B.D. Lucas and T. Kanade, 1981. "An Iterative Image Registration Technique with an Application to Stereo Vision," *Proc. Int'l Joint Conf. Artificial Intelligence*, pp. 674-679.
10. C. Harris and M. Stephens (1988). "A combined corner and edge detector". *Proceedings of the 4th Alvey Vision Conference: pages 147–151*.

ADEQUATE 3D VISUALIZATION OF REMOTELY SENSED ICE SHELF AND SEA SURFACE TEMPERATURE DATA

Guido Staub

*University of Concepción – Campus Los Ángeles
Departamento de Ciencias Geodésicas y Geomática
J.A. Coloma 0201, Los Ángeles, Chile*

ABSTRACT

This article describes a novel approach, in which Long Wavelength Infrared and Synthetic Aperture Radar (SAR) observations are combined in one single and comprehensive 3D representation of the study area that comprises the northern George VI ice shelf front and Marguerite Bay, on the west of the Antarctic Peninsula. Such a 3D model is generated without the use of any Digital Terrain Model. Instead of height information, data, Sea Surface Temperature (SST) and Radar backscattering coefficient respectively, derived from the raw observations are assigned to the z-axis. By such a three dimensional, multi-sensor model it is possible to make a statement about the actual situation of the ice shelf and the environmental conditions of the adjoining ocean. Furthermore it can be considered as the primary terrain representation that incorporates environmental information. Therefore it allows embedding observations made by other sensors in order to take further environmental parameters into account, and to generate time series to study spatio-temporal variations as well as possible interaction processes between the ice shelf and its surroundings.

KEYWORDS

Remote Sensing, 3D Visualization, Environmental study, Antarctica

1. INTRODUCTION

In particular ice shelf variations allow estimating the impact of almost any kind of environmental alterations as they are exposed to e.g. atmospheric and ocean temperature changes, precipitation anomalies, ocean currents and so on. As an example, recent studies (Humbert and Braun 2008) have shown that new rifts and break-up events could have been observed at the Wilkins Ice Shelf in 2008. During April 2009, the ice bridge connecting Wilkins Ice Shelf to Charcot Island collapsed. Further observations of the ice shelf that have been published later on (European Space Agency 2009) show that icebergs started to break away within three weeks of this event. The recurrence of such an event at a different ice shelf is conceivable because similar events have already been reported (e.g. Larsen B (2002) and the Jones Ice shelf (2003)). However the reason and the driving forces are still subject of investigation.

The study area comprises the northern front of George VI Ice shelf (GVIIS), embedded in George VI sound, and Marguerite Bay, see figure 1. With respect to GVIIS a similar situation as for the Wilkins Ice Shelf can be constituted there unless no significant changes have occurred until now. Vaughan and Doake (1996) already reported that the George VI ice shelf has almost reached its thermal limit of stability. Therefore thorough studies have to be initiated to understand the past and present changes of the ice shelf and the causes inducing them. Furthermore there are many other studies that have shown that it is important to continue monitoring to get a better understanding of George VI ice shelf dynamics (Smith et al. 2007, British Antarctic Survey 2008).

Besides the surrounding ocean and especially the situation in Marguerite Bay play a significant role that has to be considered as well. Ocean Currents

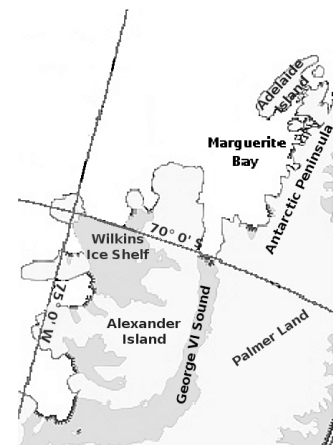


Figure 1. Map of the study area

(Beardsley et al. 2002), fresh water balance (Meredith et al. 2008) and sea ice properties (Perovich et al. 2004) are only a few examples of parameters that has to be taken into account when investigation aims to contribute to the understanding of ocean events in Marguerite Bay and their possible impact on GVIIS variations. The major problem is that all of these studies make use of in-situ and therefore local observations, what delimitates the possibility to draw conclusions for the whole area of interest.

2. DATA SETS

In order to analyse SST variations in Marguerite Bay 8-day composite Aqua MODIS (Moderate Resolution Imaging Spectroradiometer) SST 11 μ m daytime observations with a spatial resolution of 4 km are used. These are global data sets that are available free of cost on the Internet (<http://oceancolor.gsfc.nasa.gov>). In consequence, subsets that cover only the study area, Marguerite Bay, have to be generated first.

For monitoring purposes of the northern GVIIS front WSM (Wide Swath Mode) observations of the ASAR (Advanced Synthetic Aperture Radar) instrument onboard of the ENVISAT satellite are used. They have a spatial resolution of 150 m. Observations made by SAR typically are being processed with the aim that the resulting image contains the backscattering coefficient σ^0 , the normalized measure of the reflected radar signal. To get a better picture of the study area, the subsets not only contain the northern front of GVIIS, but also George VI Sound, the northeast coast of Alexander Island and the north-western part of Palmer.

The observations made by the two remote sensing sensors are already pre-processed by the responsible authorities, NASA in the case of MODIS and ESA in the case of ASAR. Nevertheless the acquired data sets are post-processed in order to reduce scene extension, the already mentioned subsets, and in consequence its file size, to eliminate Speckle noise within the ASAR scenes and to overcome missing data within the MODIS observations by spatial interpolation (Staub 2010). Finally the generated subsets are geocoded so that they are available in the WGS84 reference system (UTM zone 19 South).

3. VISUALIZATION APPROACH

Neither ASAR WSM observations nor MODIS SST data have implicitly defined the third dimension that is necessary to generate 3D models. Therefore within this approach a novel solution is applied that assigns SST and σ^0 to the z-axis respectively. This relies on the following assumptions:

- SST data refers in general to subsurface (≤ 0.01 mm) temperatures
- σ^0 is correlated to the moisture content of the target area. σ^0 of snow, ice and calm water are quite different (σ^0 (snow) $>$ σ^0 (ice) $>$ σ^0 (calm water)) so these three types can be arranged according to their vertical distribution (snow accumulates over ice that is floating on (calm) water)

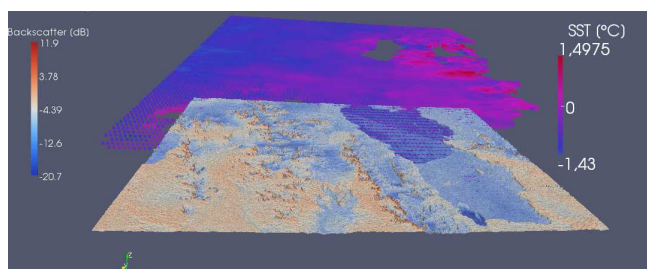


Figure 2. 3D model composed by σ^0 and SST data of the study area

Information about the spatial position is implicitly defined in the SST and σ^0 data sets and can therefore be assigned to the x and y-axis respectively. A pseudo colour model is derived from this triple by assigning a colour map scaled to the z data range. Each voxel receives its corresponding colour from the colour map according to the z-value. That way the visual interpretation of the models gets facilitated.

4. RESULTS

As an example one 3D model is shown in figure 2. The ASAR WSM sub-scene has been acquired on 03 December 2009 showing principally George VI Sound, parts of Alexander Island, Palmer Land and Marguerite Bay. This model is an approximate representation of the topography obtained from the σ^0 values. The SST data derived from observations made during 03 December 2009 and 10 December 2009 of Marguerite Bay is visualized as a plane point cloud reduced to some sub- millimetres below sea level.

Interpretation of the SST point cloud or surface, based on a Delaunay triangulation, lead to the conclusion that elevated SST can be observed in Marguerite Bay and close to the northern GVIIS front. So there might be evidence that SST variations have a direct influence on the ice shelf front formation and as a result can lead to a break off event in the near future.

5. CONCLUSIONS

- The 3D model allows to identify and to describe in a rather easy way the actual situation observable in the study area.
- A digital terrain model is not necessary in the vicinity of the northern George VI Ice shelf front because approximate height information can be derived from the considered data itself.
- Incorporation of SST permits to generate a better understanding of the whole study area and ongoing processes.

REFERENCES

- Beardsley, R., Limeburner, R. and Owens, W. 2002. Drifter measurements of surface currents near marguerite bay on the western antarctic peninsula shelf during austral summer and fall, 2001 and 2002. *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 51, no. 17-19, pp. 1947–1964.
- British Antarctic Survey 2008. The antarctic peninsula's retreating ice shelves. Science Briefing.
- European Space Agency 2009, *Satellite imagery shows fragile wilkins ice shelf destabilised*. Available from: <http://www.esa.int/esaEO/SEMRAVANJTF_index_0.html>. [16 March 2011]
- Humbert, A. and Braun, M., 2008. The wilkins ice shelf, antarctica: break-up along failure zones. *Journal of Glaciology*, vol.54, no. 188, pp. 943 – 944.
- Meredith, M., Brandon, M., Wallace, M., Clarke, A., Leng, M., Renfrew, I., van Lipzig, N. and King, J. 2008. Variability in the freshwater balance of northern marguerite bay, antarctic peninsula: Results from $\square 180$. *Deep-Sea Research Part II: Topical Studies in Oceanography*, vol. 55, no. 3-4, pp. 309–322.
- Perovich, D., Elder, B., Claffey, K., Stammerjohn, S., Smith, R., Ackley, S., Krouse, H. and Gow, A. 2004. Winter sea-ice properties in marguerite bay, antarctica. *Deep Sea Research Part II: Topical Studies in Oceanography*, vol. 51, no. 17-19, pp. 2023–2039.
- Smith, J., Bentley, M., Hodgson, D. and Cook, A., 2007. George vi ice shelf: past history, present behaviour and potential mechanism for future collapse. *Antarctic Science*, vol. 19, no. 1, pp. 131 – 142.
- Staub, G. 2010. 'A geostatistical approach for spatio temporal interpolation of sea surface temperature - marguerite bay (Antarctica)'. Proceedings of the Latin American Remote Sensing Week. Available from: <http://www.lars.cl/>. [16 March 2011].
- Vaughan, D. G. and Doake, C. S. M., 1996. Recent atmospheric warming and retreat of ice shelves on the antarctic peninsula. *Nature*, vol. 379, pp. 328–331.

A VISUAL ANALYSIS TOOL FOR AMPLIFYING STORY GENERATION CYCLE

Mina Akaishi, Makoto Sato and Koichi Hori
University of Tokyo

ABSTRACT

A vast amount of human experience records is stored in many fields. For understanding and using such chronicles effectively, it is useful to find stories embedded in records. In this paper, we regard a story as a sequence of relations among target objects and propose Chronicle Analysis Tool (CAT) for amplifying story generation cycle. It shows various patterns of chronicle data based on different viewpoints. For such purpose, we introduce the notion of chronicle data and a visual analysis framework for chronicles. Based on these notions, we describe Chronicle Analysis Tools and its implementation examples.

KEYWORDS

Story generation, chronicle analysis, frequency pattern.

1. INTRODUCTION

Currently, a vast amount of information is being accumulated at an accelerating pace in many fields. As a result, piles of records are stored. However, it is difficult to understand contents of them within the limit of time. Moreover it is harder to create new idea or to discover new knowledge from the chaos of information.

A novel, a story, a myth or a tale spares us from the complexity of the world. A simple story can communicate a complex multi-dimensioned idea, not by simply transmitting information as a message, but by actively involving the listeners in co-creating that idea (Brown 2005). In order to identify the topic of a story, techniques for topic detection and tracking, text mining and visualization are well explored (Swan 2000)(Wong 2000)(Kleinberg 2003). Besides, there are many studies for automatic story generation based on artificial intelligence techniques (Meehan 1977)(Pemberton 1989)(Turner 1994)(Bringsjord 1999)(Perez 2001). Also, there are studies for story generation support with interaction between systems and the users (Cavazza 2002)(Riedl 2006)(Young 1999).

Information visualization systems enable users to find patterns, relationships, and structures in data and help users verify knowledge or hypotheses.

In this paper, we regard a story as a sequence of relations among target objects and propose Chronicle Analysis Tool (CAT) for amplifying story generation cycle. It shows various patterns of chronicle data based on different viewpoints.

The remainder of the paper is organized as follows. In Section 2, we introduce the notion of chronicle data and a chronicle analysis framework. In Section 3, we present Chronicle Analysis Tools and its implementation examples. In Section 4, we make some concluding remarks.

2. CHRONICLE ANALYSIS FOR STORY GENERATION

In this section, we describe the chronicle data model and its example. Then we explain a chronicle analysis framework and how it helps story generation cycle.

2.1 Chronicle Data

In this study, we regard chronicle data as text with some attributes which include date information. For example, each record of *Dai-Nihon Shiryo* database stores historical events with some attributes, such as the date when the event happened, the place where the event occurred, and the names and official positions of people who were involved in the event.

Dai-Nihon Shiryo, which is the still-to-be completed, will be the most comprehensive collection of pre-modern Japanese historical materials. It consists of primary source material from the period between 887 to 1867, and is organized chronologically by major historical events. A part of *Dai-Nihon Shiryo* is provided as *Dai-Nihon Shiryo* database.

The attributes of *Dai-Nihon Shiryo* are given by historiographers. It means that sophisticated sets of attribute values are formed by themselves.

2.2 A Framework for Chronicle Analysis

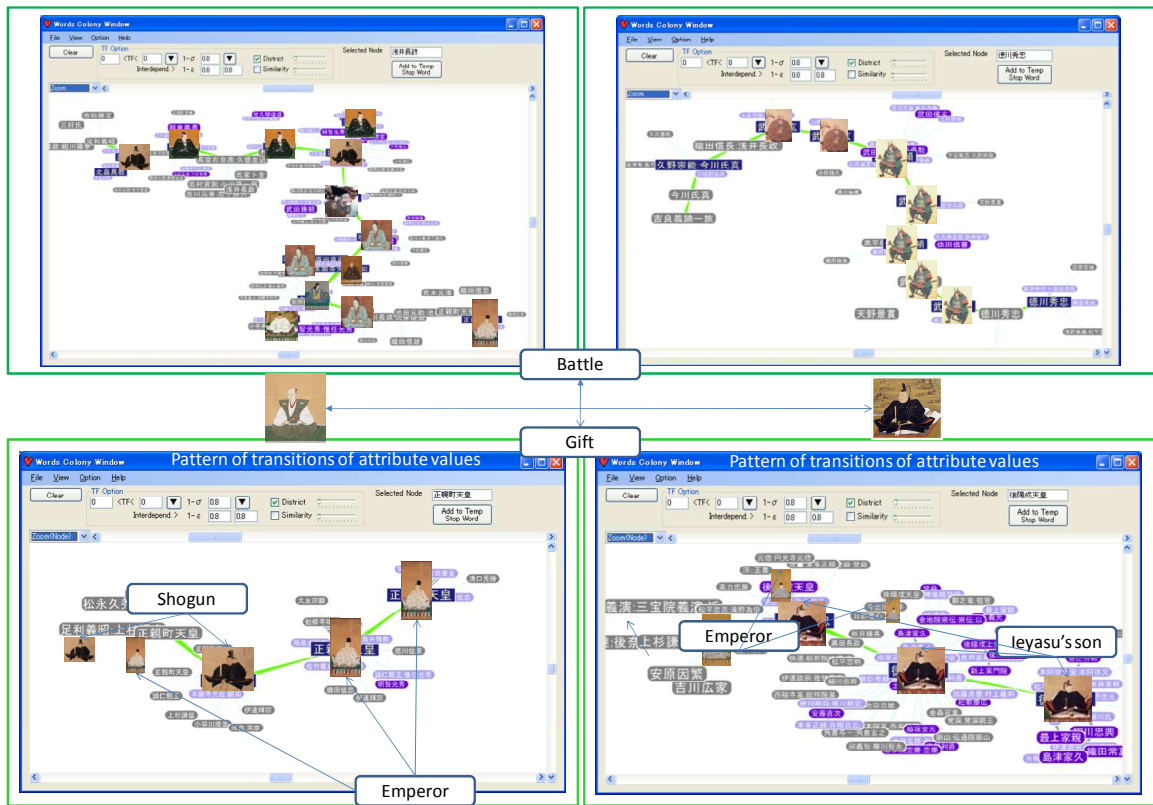
Figure 1 shows a framework to analyze chronicle data and its implementation examples. It consists of three layers. The first layer provides function of filtering data by specifying conditions. The second layer visualizes the distribution of filtered data as a combination of matrix and bar chart graphs. If users can find interesting patterns on the distribution map in the second layer, the third layer represents transitions of relations among focal attribute values by Topic Sequence technique (Akaishi 2007). Topic Sequence is regarded as a graphical plot of a series of texts chunk. Each chunk is visualized by Word Colony (Akaishi 2004), which visualizes the dependency relationships among terms in a text as a graph. Word Colony is regarded as a visual abstract of a relation among terms in a text. Topic Sequence is formed by the concatenation of Word Colonies. Topic Sequence provides overviews of the topical change of chunk text content over time.

At the beginning, users see the patterns of distribution for whole data. It gives an overview of target data set. If users can find interesting patterns on the distribution map by selecting attribute for the vertical axis, then users can narrow their interest down by filtering data or users can see more detail information by visualizing a transition of interesting attribute values. The distribution map or transitions of attribute values suggest users' next keywords for specifying another viewpoint. Reiteration of this process corresponds to a story generation cycle. If users can find interesting patterns of transitions of attribute values, then they are the fundamentals of new stories.

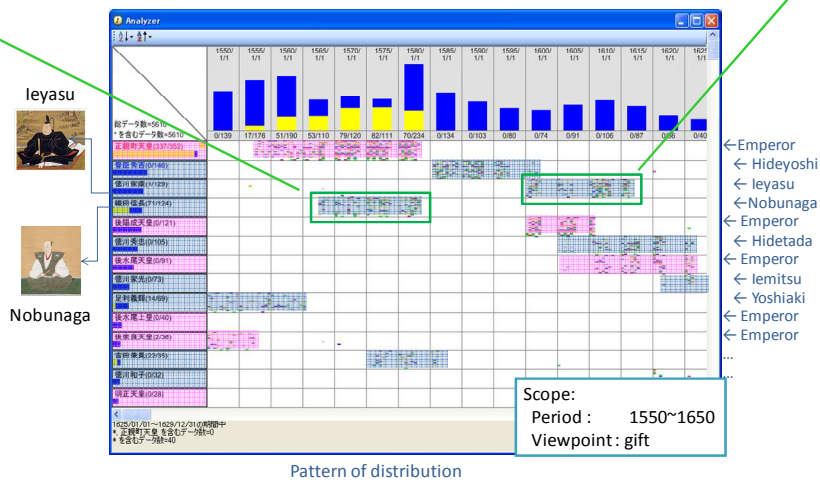
3. CAT APPLICATION

Let assume that a user sees the distribution map of whole data in *Dai-Nihon Shiryo* database and find an interesting keyword "gift". Then records that include keyword "gift" and occurred between 1550 and 1650 are selected from *Dai-Nihon Shiryo* database. The second layer in figure 1 shows human relations as a distribution map. The vertical axis represents names of persons in order of frequency and the bar chart at left part shows the number of records that include the person's name.

The distribution map represents personal activities concerning "gift". It is easy to see the cooccurrence of Japanese emperors and men of power. This distribution suggests a hypothesis that the relationships based on "gift" indicate the relation of authorities and powers.



The third layer: Visualizing a transition of attribute values



The second layer: Visualizing a distribution of data

The first layer: Filtering data

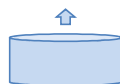


Figure 1. A Visual Analysis Framework for Chronicle Data.

The third layer in figure 1 shows four Topic Sequences representing human relations of two persons, who are *Nobunaga ODA* and *Ieyasu TOKUGAWA*, based on “gift” and “battle”. Major persons in *Nobunaga*’s human relation concerning about “gift” are authorities at that time. On the other hand, in the case of *Ieyasu*’s human relation concerning about “gift”, major persons are authorities at only early days and then *Ieyasu*’s

son took important place at almost days. *Ieyasu* is the first *shogun* in the *Tokugawa* shogunate. He consolidates his son's position as his successor,

Topic Sequences concerning "battle" show that *Nobunaga* is related with many *daimyos* (territorial lords in premodern Japan) while *Ieyasu* is related with a few *daimyos*. In light of knowledge about the territory these *daimyos* had control over, the visualization gives rise to the interpretation that *Nobunaga* fought various enemies all over Japan while *Ieyasu* fought the big enemy next to his territory, which actually corresponds to the history as we know. These patterns help users create new interpretations of history by exploring chronological records.

Users can find several relationships among people based on various viewpoints. It helps to consider the stories or hypotheses of new ideas based on historical event records. Not only each Topic Sequence but also a combination of Topic Sequences supports users to compose more complicated story.

4. CONCLUSION

In this paper, we proposed a framework for visual chronological analysis to support story generation cycle. Based on the proposed framework, we implemented a tool called CAT, which provides a functionality to represent latent patterns in chronicles based on the dynamics of *scope*, *axis* and *focus*. It supports users to interpret new histories based on various viewpoints. We demonstrated that the tool can actually give rise to various interpretations of chronicle records by applying the tool to a comprehensive pre-modern Japanese historical material, *Dai-Nihon Shiryo*.

For the future work, we would like to apply the framework to other domain's chronicle data and evaluate the effectiveness of the system.

REFERENCES

- Akaishi, M. et al. 2004. An associative information retrieval based on the dependency of term cooccurrence. *Lecture Notes in Artificial Intelligence*, volume 3245, pp.295-206.
- Akaishi, M. et. al. 2007, Narrative based Topic Visualization for Chronological Data, *Proc. of International Conference on Information Visualization (IV2007)*, pp.139-144.
- Bringsjord, S. and Ferrucci, D., 1999. *Artificial Intelligence and Literary Creativity: Inside the Mind of Brutus, a Storytelling Machine*. L. Erlbaum Associates Inc., Hillsdale, NJ, USA.
- Brown, J.S., 2005. *Storytelling in organizations: Why storytelling is transforming 21st century organizations and management*. Butterworth-Heinemann.
- Cavazza, M., et al., 2002. *Character-based interactive storytelling*. IEEE Intelligent Systems, Vol. 17, No. 4, pp. 17–24.
- Jon Kleinbergm, 2003. *Bursty and Hierarchical Structure in Streams*. DATA MINING AND KNOWLEDGE DISCOVERY, Vol. 7, No. 4, pp. 373–397.
- Meehan, J.R., 1977. *Tale-spin, an interactive program that writes stories*. Proceedings of the fifth international joint conference on artificial intelligence, pp. 91–98.
- Pemberton, L., 1989. *A modular approach to story generation*. Proceedings of the fourth conference on European chapter of the Association for Computational Linguistics, pp. 217– 224.
- Perez Mike Sharples, Perez y` R., 2001. "MEXICA: A computer model of a cognitive account of creative writing. *Journal of Experimental & Theoretical Artificial Intelligence*, Vol. 13, No. 2, pp. 119–139.
- Riedl, M. and Young, R., "From linear story generation to branching story graphs", *Computer Graphics and Applications*, IEEE, Vol. 26, No. 3, pp. 23–31, 2006.
- Russell Swan and James Allan, 2000. *Automatic Generation of Overview Timelines*. SIGIR '00 Proceedings of the 23rd annual international.
- Turner, S., 1994. *The creative process: A computer model of storytelling and creativity*. Lawrence Erlbaum.
- Wong, P., et al., 2000. *Visualizing sequential patterns for text mining*. In Proc. IEEE Information Visualization.
- Young, R. M., 1999. *Notes on the Use of Plan Structures in the Creation of Interactive Plot*. the AAAI Fall Symposium on Narrative Intelligence.

AUTHOR INDEX

Åhlén, J.....	187	Gómez, A.....	233
Ahmed, H.....	192	Gook, L.....	343
Akaishi, M.....	349	Guillén, C.....	287
Aleksandrowicz, I.....	99	Gupta, R.....	343
Aleynikov, A.....	255	Helli, B.....	297
Ayaquica, I.....	287	Hori, K.....	349
Azmi, S.....	245	Huang, P.....	35, 41
Baik, S.....	276	Huang, Q.....	153
Bauer, T.....	49	Joo, C.....	197
Bautista, C.....	287	Jorda, L.....	83
Belokamenskaya, A.....	337	Karlsson, A.....	19, 228
Belosokhov, D.....	255	Kawaguchi, Y.....	265
Bobkov, A.....	255	Khalid, N.....	303
Bolochko, K.....	271	Khan, A.....	192
Boudaren, M.....	11	Kim, S.....	276
Bu, Y.....	292	Klimenko, A.....	255
Calderón, J.....	233	Klimenko, S.....	255
Cao, S.....	292	Kolditz, O.....	169
Capanna, C.....	83	Kong, D.....	153
Chambelland, J.....	133	Kovalovs, M.....	271
Chan, C.....	41	Kozlíková, B.....	99
Chang, Y.....	202	Krüger, J.....	160
Chen, C.....	35, 41	Lamberti, F.....	67
Chen, S.....	217	Lamy, P.....	83
Choeh, J.....	276	Lee, B.....	276
Daniel, M.....	239	Lee, C.....	276
Delias, P.....	281	Lee, J.....	276
Desbenoit, B.....	133	Lee, M.....	276
Dias, M.....	177	Lee, Y.....	202
Doulamis, A.....	281	Lee, Y.....	340
Dukuzumuremyi, J.....	60	Lemuz, R.....	287
Duplex, B.....	239	Leonov, A.....	255
Ejaz, N.....	276	Letavin, M.....	249
Eremchenko, E.....	255	Liang, Y.....	60
Faure, M.....	260	Lin, P.....	35, 41
Fedorov, S.....	249	Liu, F.....	115
Fischer, T.....	169	López, J.....	233
Frolov, P.....	255	Marco, J.....	27
Fuh, C.....	217	Marin, G.....	260
Fumero, S.....	75	Martins, P.....	3
Furukawa, M.....	123	Matsatsinis, N.....	281
Gavrilov, N.....	337	Matveev, I.....	313
Gesquiere, G.....	83, 133, 239	Michaelis, B.....	319
Glazs, A.....	271	Moghaddam, M.....	297
Goedemé, T.....	331	Moir, C.....	141

Monfrini, E.	11	Vázquez, P.	27
Moustakas, K.	91	Wang, J.	217
Musci, M.	327	Wang, L.	60
Najjar, A.	223	Wang, L.	153
Ogniewski, J.	19	Wang, R.	153
Oh, K.	197	Watanabe, M.	123
Oliveira, C.	3	Xydeas, C.	192
Paillé, D.	260	Yahya, S.	303
Paravati, G.	67	Yamaguchi, J.	177
Perdu, F.	239	Yamamoto, M.	123
Pérez, A.	233	Yang, J.	197
Pérez, F.	75	Yang, X.	115
Pieczynski, W.	11	Yen, W.	202
Raffin, R.	133	Yin, B.	153
Ragnemalm, I.	19, 228	Yoon, J.	340
Rahmat, R.	245	Yu, Z.	202
Renard, M.	260	Zagrouba, E.	223, 308
Rink, K.	169	Zaid, A.	212
Romero, M.	287	Zhang, J.	115
Sanna, A.	67	Zhu, Q.	292
Sato, H.	265	Zou, B.	60
Sato, M.	349	Zuo, X.	292
Savchenko, M.	107		
Savchenko, V.	107		
Schiewe, A.	160		
Seipel, S.	187		
Shariff, Y.	303		
Shevtsov, M.	249		
Silva, A.	3		
Silva, S.	3		
Sochor, J.	99		
Soudani, A.	308		
Staub, G.	346		
Stavropoulos, G.	91		
Suzuki, I.	123		
Takemura, A.	207		
Tamm, G.	160		
Teixeira, A.	3		
Toueti, M.	212		
Tsai, T.	41		
Turlapov, V.	337		
Tzovaras, D.	91		
Umugwaneza, M.	60		
Valdés, D.	287		
Van Beeck, K.	331		
Van den Heuvel, L.	331		
Von Enzberg, S.	319		
Von Itzstein, G.	141		