

LNCS 6455

George Bebis et al. (Eds.)

# Advances in Visual Computing

6th International Symposium, ISVC 2010  
Las Vegas, NV, USA, November/December 2010  
Proceedings, Part III

3  
Part III



Springer

*Commenced Publication in 1973*

Founding and Former Series Editors:

Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

## Editorial Board

David Hutchison

*Lancaster University, UK*

Takeo Kanade

*Carnegie Mellon University, Pittsburgh, PA, USA*

Josef Kittler

*University of Surrey, Guildford, UK*

Jon M. Kleinberg

*Cornell University, Ithaca, NY, USA*

Alfred Kobsa

*University of California, Irvine, CA, USA*

Friedemann Mattern

*ETH Zurich, Switzerland*

John C. Mitchell

*Stanford University, CA, USA*

Moni Naor

*Weizmann Institute of Science, Rehovot, Israel*

Oscar Nierstrasz

*University of Bern, Switzerland*

C. Pandu Rangan

*Indian Institute of Technology, Madras, India*

Bernhard Steffen

*TU Dortmund University, Germany*

Madhu Sudan

*Microsoft Research, Cambridge, MA, USA*

Demetri Terzopoulos

*University of California, Los Angeles, CA, USA*

Doug Tygar

*University of California, Berkeley, CA, USA*

Gerhard Weikum

*Max Planck Institute for Informatics, Saarbruecken, Germany*

George Bebis Richard Boyle Bahram Parvin  
Darko Koracin Ronald Chung Riad Hammoud  
Muhammad Hussain Tan Kar-Han Roger Crawfis  
Daniel Thalmann David Kao Lisa Avila (Eds.)

# Advances in Visual Computing

6th International Symposium, ISVC 2010  
Las Vegas, NV, USA  
November 29 - December 1, 2010  
Proceedings, Part III

**Volume Editors**

George Bebis, E-mail: [bebis@cse.unr.edu](mailto:bebis@cse.unr.edu)

Richard Boyle, E-mail: [richard.boyle@nasa.gov](mailto:richard.boyle@nasa.gov)

Bahram Parvin, E-mail: [parvin@hpcrd.lbl.gov](mailto:parvin@hpcrd.lbl.gov)

Darko Koracin, E-mail: [darko@dri.edu](mailto:darko@dri.edu)

Ronald Chung, E-mail: [rchung@cuhk.edu.hk](mailto:rchung@cuhk.edu.hk)

Riad Hammoud, E-mail: [riad.hammoud@dynavoxtech.com](mailto:riad.hammoud@dynavoxtech.com)

Muhammad Hussain, E-mail: [mhussain@ccis.edu.sa](mailto:mhussain@ccis.edu.sa)

Tan Kar-Han, E-mail: [karhan.tan@hp.com](mailto:karhan.tan@hp.com)

Roger Crawfis, E-mail: [crawfis@cse.ohio-state.edu](mailto:crawfis@cse.ohio-state.edu)

Daniel Thalmann, E-mail: [daniel.thalmann@epfl.ch](mailto:daniel.thalmann@epfl.ch)

David Kao, E-mail: [davidkao@nas.nasa.gov](mailto:davidkao@nas.nasa.gov)

Lisa Avila, E-mail: [lisa.avila@kitware.com](mailto:lisa.avila@kitware.com)

Library of Congress Control Number: 2010939054

CR Subject Classification (1998): I.3, H.5.2, I.4, I.5, I.2.10, J.3, F.2.2, I.3.5

LNCS Sublibrary: SL 6 – Image Processing, Computer Vision, Pattern Recognition, and Graphics

ISSN 0302-9743

ISBN-10 3-642-17276-8 Springer Berlin Heidelberg New York

ISBN-13 978-3-642-17276-2 Springer Berlin Heidelberg New York

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. Duplication of this publication or parts thereof is permitted only under the provisions of the German Copyright Law of September 9, 1965, in its current version, and permission for use must always be obtained from Springer. Violations are liable to prosecution under the German Copyright Law.

[springer.com](http://springer.com)

© Springer-Verlag Berlin Heidelberg 2010  
Printed in Germany

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India  
Printed on acid-free paper 06/3180

# Preface

It is with great pleasure that we present the proceedings of the 6th International, Symposium on Visual Computing (ISVC 2010), which was held in Las Vegas, Nevada. ISVC provides a common umbrella for the four main areas of visual computing including vision, graphics, visualization, and virtual reality. The goal is to provide a forum for researchers, scientists, engineers, and practitioners throughout the world to present their latest research findings, ideas, developments, and applications in the broader area of visual computing.

This year, the program consisted of 14 oral sessions, one poster session, 7 special tracks, and 6 keynote presentations. The response to the call for papers was very good; we received over 300 submissions for the main symposium from which we accepted 93 papers for oral presentation and 73 papers for poster presentation. Special track papers were solicited separately through the Organizing and Program Committees of each track. A total of 44 papers were accepted for oral presentation and 6 papers for poster presentation in the special tracks.

All papers were reviewed with an emphasis on potential to contribute to the state of the art in the field. Selection criteria included accuracy and originality of ideas, clarity and significance of results, and presentation quality. The review process was quite rigorous, involving two – three independent blind reviews followed by several days of discussion. During the discussion period we tried to correct anomalies and errors that might have existed in the initial reviews. Despite our efforts, we recognize that some papers worthy of inclusion may have not been included in the program. We offer our sincere apologies to authors who contributions might have been overlooked.

We wish to thank everybody who submitted their work to ISVC 2010 for review. It was because of their contributions that we succeeded in having a technical program of high scientific quality. In particular, we would like to thank the ISVC 2010 Area Chairs, the organizing institutions (UNR, DRI, LBNL, and NASA Ames), the government and industrial sponsors (Air Force Research Lab, Intel, DigitalPersona, Equinox, Ford, Hewlett Packard, Mitsubishi Electric Research Labs, iCore, Toyota, Delphi, General Electric, Microsoft MSDN, and Volt), the international Program Committee, the special track organizers and their Program Committees, the keynote speakers, the reviewers, and especially the authors that contributed their work to the symposium. In particular, we would like to thank *Air Force Research Lab*, *Mitsubishi Electric Research Labs*, and *Volt* for kindly sponsoring four “best paper awards” this year.

We sincerely hope that ISVC 2010 offered opportunities for professional growth.

# Organization

## ISVC 2010 Steering Committee

Bebis George	University of Nevada, Reno, USA
Boyle Richard	NASA Ames Research Center, USA
Parvin Bahram	Lawrence Berkeley National Laboratory, USA
Koracin Darko	Desert Research Institute, USA

## ISVC 2010 Area Chairs

### Computer Vision

Chang Ronald	The Chinese University of Hong Kong, Hong Kong
Hammoud Riad	DynaVox Systems, USA

### Computer Graphics

Hussain Muhammad	King Saud University, Saudi Arabia
Tan Kar-Han	Hewlett Packard Labs, USA

### Virtual Reality

Crawfis Roger	Ohio State University, USA
Thalman Daniel	EPFL, Switzerland

### Visualization

Kao David	NASA Ames Research Lab, USA
Avila Lisa	Kitware, USA

### Publicity

Erol Ali	Ocali Information Technology, Turkey
----------	--------------------------------------

### Local Arrangements

Regentova Emma	University of Nevada, Las Vegas, USA
----------------	--------------------------------------

### Special Tracks

Porikli Fatih	Mitsubishi Electric Research Labs, USA
---------------	--

## ISVC 2010 Keynote Speakers

Kakadiaris Ioannis	University of Houston, USA
Hollerer Tobias	University of California at Santa Barbara, USA
Stasko John	Georgia Institute of Technology, USA
Seitz Steve	University of Washington, USA
Pollefeys Marc	ETH Zurich, Switzerland
Majumder Aditi	University of California, Irvine, USA

## ISVC 2010 International Program Committee

### (Area 1) Computer Vision

Abidi Besma	University of Tennessee, USA
Abou-Nasr Mahmoud	Ford Motor Company, USA
Agajan Sos	University of Texas at San Antonio, USA
Aggarwal J. K.	University of Texas, Austin, USA
Amayeh Gholamreza	Eyecom, USA
Agouris Peggy	George Mason University, USA
Argyros Antonis	University of Crete, Greece
Asari Vijayan	University of Dayton, USA
Basu Anup	University of Alberta, Canada
Bekris Kostas	University of Nevada at Reno, USA
Belyaev Alexander	Max-Planck-Institut fuer Informatik, Germany
Bensrhair Abdelaziz	INSA-Rouen, France
Bhatia Sanjiv	University of Missouri-St. Louis, USA
Bimber Oliver	Johannes Kepler University Linz, Austria
Bioucas Jose	Instituto Superior Tecnico, Lisbon, Portugal
Birchfield Stan	Clemson University, USA
Bourbakis Nikolaos	Wright State University, USA
Brimkov Valentin	State University of New York, USA
Campadelli Paola	Università degli Studi di Milano, Italy
Cavallaro Andrea	Queen Mary, University of London, UK
Charalampidis Dimitrios	University of New Orleans, USA
Chellappa Rama	University of Maryland, USA
Chen Yang	HRL Laboratories, USA
Cheng Hui	Sarnoff Corporation, USA
Cochran Steven Douglas	University of Pittsburgh, USA
Cremers Daniel	University of Bonn, Germany
Cui Jinshi	Peking University, China
Darbon Jerome	CNRS-Ecole Normale Superieure de Cachan, France
Davis James W.	Ohio State University, USA

Debrunner Christian	Colorado School of Mines, USA
Demirdjian David	MIT, USA
Duan Ye	University of Missouri-Columbia, USA
Doulamis Anastasios	National Technical University of Athens, Greece
Dowdall Jonathan	510 Systems, USA
El-Ansari Mohamed	Ibn Zohr University, Morocco
El-Gammal Ahmed	University of New Jersey, USA
Eng How Lung	Institute for Infocomm Research, Singapore
Erol Ali	Ocali Information Technology, Turkey
Fan Guoliang	Oklahoma State University, USA
Ferri Francesc	Universitat de Valencia, Spain
Ferryman James	University of Reading, UK
Foresti GianLuca	University of Udine, Italy
Fowlkes Charless	University of California, Irvine, USA
Fukui Kazuhiro	The University of Tsukuba, Japan
Galata Aphrodite	The University of Manchester, UK
Georgescu Bogdan	Siemens, USA
Gleason, Shaun	Oak Ridge National Laboratory, USA
Goh Wooi-Boon	Nanyang Technological University, Singapore
Guerra-Filho Gutemberg	University of Texas Arlington, USA
Guevara, Angel Miguel	University of Porto, Portugal
Gustafson David	Kansas State University, USA
Harville Michael	Hewlett Packard Labs, USA
He Xiangjian	University of Technology, Sydney, Australia
Heikkilä Janne	University of Oulu, Finland
Heyden Anders	Lund University, Sweden
Hongbin Zha	Peking University, China
Hou Zujun	Institute for Infocomm Research, Singapore
Hua Gang	Nokia Research Center, USA
Imiya Atsushi	Chiba University, Japan
Jia Kevin	IGT, USA
Kamberov George	Stevens Institute of Technology, USA
Kampel Martin	Vienna University of Technology, Austria
Kamberova Gerda	Hofstra University, USA
Kakadiaris Ioannis	University of Houston, USA
Kettebekov Sanzhar	Keane inc., USA
Khan Hameed Ullah	King Saud University, Saudi Arabia
Kim Tae-Kyun	University of Cambridge, UK
Kimia Benjamin	Brown University, USA
Kisacanin Branislav	Texas Instruments, USA
Klette Reinhard	Auckland University, New Zealand
Kokkinos Iasonas	Ecole Centrale Paris, France
Kollias Stefanos	National Technical University of Athens, Greece

Komodakis Nikos	Ecole Centrale de Paris, France
Kozintsev	Igor, Intel, USA
Kuno	Yoshinori, Saitama University, Japan
Kyungnam Kim	HRL Laboratories, USA
Latecki Longin Jan	Temple University, USA
Lee D. J.	Brigham Young University, USA
Li Chunming	Vanderbilt University, USA
Li Fei-Fei	Stanford University, USA
Lin Zhe	Adobe, USA
Lisin Dima	VidoeIQ, USA
Lee Seong-Whan	Korea University, Korea
Leung Valerie	Kingston University, UK
Leykin Alex	Indiana University, USA
Li Shuo	GE Healthcare, Canada
Li Wenjing	STI Medical Systems, USA
Liu Jianzhuang	The Chinese University of Hong Kong, Hong Kong
Loss Leandro	Lawrence Berkeley National Lab, USA
Ma Yunqian	Honeywell Labs, USA
Maeder Anthony	University of Western Sydney, Australia
Makris Dimitrios	Kingston University, UK
Maltoni Davide	University of Bologna, Italy
Mauer Georg	University of Nevada, Las Vegas, USA
Maybank Steve	Birkbeck College, UK
McGraw Tim	West Virginia University, USA
Medioni Gerard	University of Southern California, USA
Melenchón Javier	Universitat Oberta de Catalunya, Spain
Metaxas Dimitris	Rutgers University, USA
Miller Ron	Wright Patterson Air Force Base, USA
Ming Wei	Konica Minolta, USA
Mirmehdi Majid	Bristol University, UK
Monekosso Dorothy	Kingston University, UK
Mueller Klaus	SUNY Stony Brook, USA
Mulligan Jeff	NASA Ames Research Center, USA
Murray Don	Point Grey Research, Canada
Nait-Charif Hammadi	Bournemouth University, UK
Nefian Ara	NASA Ames Research Center, USA
Nicolescu Mircea	University of Nevada, Reno, USA
Nixon Mark	University of Southampton, UK
Nolle Lars	The Nottingham Trent University, UK
Ntalianis Klimis	National Technical University of Athens, Greece
Or Siu Hang	The Chinese University of Hong Kong, Hong Kong
Papadourakis George	Technological Education Institute, Greece

Papanikolopoulos Nikolaos	University of Minnesota, USA
Pati Peeta Basa	First Indian Corp., India
Patras Ioannis	Queen Mary University, London, UK
Petrakis Euripides	Technical University of Crete, Greece
Peyronnet Sylvain	LRDE/EPITA, France
Pinhanez Claudio	IBM Research, Brazil
Piccardi Massimo	University of Technology, Australia
Pietikäinen Matti	LRDE/University of Oulu, Finland
Porikli Fatih	Mitsubishi Electric Research Labs, USA
Prabhakar Salil	DigitalPersona Inc., USA
Prati Andrea	University of Modena and Reggio Emilia, Italy
Prokhorov Danil	Toyota Research Institute, USA
Prokhorov Pylvanainen Timo	Nokia, Finland
Qi Hairong	University of Tennessee at Knoxville, USA
Qian Gang	Arizona State University, USA
Raftopoulos Kostas	National Technical University of Athens, Greece
Reed Michael	Blue Sky Studios, USA
Regazzoni Carlo	University of Genoa, Italy
Regentova Emma	University of Nevada, Las Vegas, USA
Remagnino Paolo	Kingston University, UK
Ribeiro Eraldo	Florida Institute of Technology, USA
Robles-Kelly Antonio	National ICT Australia (NICTA), Australia
Ross Arun	West Virginia University, USA
Salgian Andrea	The College of New Jersey, USA
Samal Ashok	University of Nebraska, USA
Sato Yoichi	The University of Tokyo, Japan
Samir Tamer	Ingersoll Rand Security Technologies, USA
Sandberg Kristian	Computational Solutions, USA
Sarti Augusto	DEI Politecnico di Milano, Italy
Savakis Andreas	Rochester Institute of Technology, USA
Schaefer Gerald	Loughborough University, UK
Scalzo Fabien	University of California at Los Angeles, USA
Scharcanski Jacob	UFRGS, Brazil
Shah Mubarak	University of Central Florida, USA
Shi Pengcheng	The Hong Kong University of Science and Technology, Hong Kong
Shimada Nobutaka	Ritsumeikan University, Japan
Singh Meghna	University of Alberta, Canada
Singh Rahul	San Francisco State University, USA
Skurikhin Alexei	Los Alamos National Laboratory, USA
Souvenir, Richard	University of North Carolina - Charlotte, USA

Su Chung-Yen	National Taiwan Normal University, Taiwan
Sugihara Kokichi	University of Tokyo, Japan
Sun Zehang	Apple, USA
Syeda-Mahmood Tanveer	IBM Almaden, USA
Tan Tieniu	Chinese Academy of Sciences, China
Tavakkoli Alireza	University of Houston - Victoria, USA
Tavares, Joao	Universidade do Porto, Portugal
Teoh Eam Khwang	Nanyang Technological University, Singapore
Thiran Jean-Philippe	Swiss Federal Institute of Technology Lausanne (EPFL), Switzerland
Tistarelli Massimo	University of Sassari, Italy
Tsechpenakis Gabriel	University of Miami, USA
Tsui T.J.	Chinese University of Hong Kong, Hong Kong
Trucco Emanuele	University of Dundee, UK
Tubaro Stefano	DEI, Politecnico di Milano, Italy
Uhl Andreas	Salzburg University, Austria
Velastin Sergio	Kingston University London, UK
Verri Alessandro	Universitá di Genova, Italy
Wang Charlie	The Chinese University of Hong Kong, Hong Kong
Wang Junxian	Microsoft, USA
Wang Song	University of South Carolina, USA
Wang Yunhong	Beihang University, China
Webster Michael	University of Nevada, Reno, USA
Wolff Larry	Equinox Corporation, USA
Wong Kenneth	The University of Hong Kong, Hong Kong
Xiang Tao	Queen Mary, University of London, UK
Xue Xinwei	Fair Isaac Corporation, USA
Xu Meihe	University of California at Los Angeles, USA
Yang Ruigang	University of Kentucky, USA
Yi Lijun	SUNY at Binghamton, USA
Yu Kai	NEC Labs, USA
Yu Ting	GE Global Research, USA
Yu Zeyun	University of Wisconsin-Milwaukee, USA
Yuan Chunrong	University of Tuebingen, Germany
Zhang Yan	Delphi Corporation, USA
Zhou Huiyu	Queen's University Belfast, UK

## (Area 2) Computer Graphics

Abd Rahni Mt Piah	Universiti Sains Malaysia, Malaysia
Abram Greg	IBM T.J.Watson Reseach Center, USA
Adamo-Villani Nicoletta	Purdue University, USA

Agu Emmanuel	Worcester Polytechnic Institute, USA
Andres Eric	Laboratory XLIM-SIC, University of Poitiers, France
Artusi Alessandro	CaSToRC Cyprus Institute, Cyprus
Baciu George	Hong Kong PolyU, Hong Kong
Balcisoy Selim Saffet	Sabanci University, Turkey
Barneva Reneta	State University of New York, USA
Bartoli Vilanova Anna	Eindhoven University of Technology, The Netherlands
Belyaev Alexander	Max Planck-Institut fuer Informatik, Germany
Benes Bedrich	Purdue University, USA
Berberich Eric	Max-Planck Institute, Germany
Bilalis Nicholas	Technical University of Crete, Greece
Bimber Oliver	Johannes Kepler University Linz, Austria
Bohez Erik	Asian Institute of Technology, Thailand
Bouatouch Kadi	University of Rennes I, IRISA, France
Brimkov Valentin	State University of New York, USA
Brown Ross	Queensland University of Technology, Australia
Callahan Steven	University of Utah, USA
Chen Min	University of Wales Swansea, UK
Cheng Irene	University of Alberta, Canada
Chiang Yi-Jen	Polytechnic Institute of New York University, USA
Choi Min	University of Colorado at Denver, USA
Comba Joao	Univ. Fed. do Rio Grande do Sul, Brazil
Cremer Jim	University of Iowa, USA
Culbertson Bruce	HP Labs, USA
Debattista Kurt	University of Warwick, UK
Deng Zhigang	University of Houston, USA
Dick Christian	Technical University of Munich, Germany
DiVerdi Stephen	Adobe, USA
Dingliana John	Trinity College, Ireland
El-Sana Jihad	Ben Gurion University of The Negev, Israel
Entezari Alireza	University of Florida, USA
Fiorio Christophe	Université Montpellier 2, LIRMM, France
Floriani Leila De	University of Genoa, Italy
Gaither Kelly	University of Texas at Austin, USA
Gao Chunyu	Epson Research and Development, USA
Geist Robert	Clemson University, USA
Gelb Dan	Hewlett Packard Labs, USA
Gotz David	IBM, USA
Gooch Amy	University of Victoria, Canada

Gu David	State University of New York at Stony Brook, USA
Guerra-Filho Gutemberg	University of Texas Arlington, USA
Habib Zulfiqar	National University of Computer and Emerging Sciences, Pakistan
Hadwiger Markus	KAUST, Saudi Arabia
Haller Michael	Upper Austria University of Applied Sciences, Austria
Hamza-Lup Felix	Armstrong Atlantic State University, USA
Han JungHyun	Korea University, Korea
Hao Xuejun	Columbia University and NYSPI, USA
Hernandez Jose Tiberio	Universidad de los Andes, Colombia
Huang Mao Lin	University of Technology, Australia
Huang Zhiyong	Institute for Infocomm Research, Singapore
Joaquim Jorge	Instituto Superior Tecnico, Portugal
Ju Tao	Washington University, USA
Julier Simon J.	University College London, UK
Kakadiaris Ioannis	University of Houston, USA
Kamberov George	Stevens Institute of Technology, USA
Kim Young	Ewha Womans University, Korea
Klosowski James	AT&T Labs, USA
Kobbelt Leif	RWTH Aachen, Germany
Kuan Lee Hwee	Bioinformatics Institute, ASTAR, Singapore
Lai Shuhua	Virginia State University, USA
Lakshmanan Geetika	IBM T.J. Watson Research Center, USA
Lee Chang Ha	Chung-Ang University, Korea
Lee Tong-Yee	National Cheng-Kung University, Taiwan
Levine Martin	McGill University, Canada
Lewis Bob	Washington State University, USA
Li Frederick	University of Durham, UK
Lindstrom Peter	Lawrence Livermore National Laboratory, USA
Linsen Lars	Jacobs University, Germany
Loviscach Joern	Fachhochschule Bielefeld (University of Applied Sciences), Germany
Magnor Marcus	TU Braunschweig, Germany
Majumder Aditi	University of California, Irvine, USA
Mantler Stephan	VRVis Research Center, Austria
Martin Ralph	Cardiff University, UK
McGraw Tim	West Virginia University, USA
Meenakshisundaram Gopi	University of California-Irvine, USA
Mendoza Cesar	NaturalMotion Ltd., USA
Metaxas Dimitris	Rutgers University, USA
Myles Ashish	University of Florida, USA
Nait-Charif Hammadi	University of Dundee, UK

Nasri Ahmad	American University of Beirut, Lebanon
Noma Tsukasa	Kyushu Institute of Technology, Japan
Okada Yoshihiro	Kyushu University, Japan
Olague Gustavo	CICESE Research Center, Mexico
Oliveira Manuel M.	Univ. Fed. do Rio Grande do Sul, Brazil
Ostromoukhov Victor M.	University of Montreal, Canada
Pascucci Valerio	University of Utah, USA
Peters Jorg	University of Florida, USA
Qin Hong	State University of New York at Stony Brook, USA
Razdan Anshuman	Arizona State University, USA
Reed Michael	Columbia University, USA
Renner Gabor	Computer and Automation Research Institute, Hungary
Rosenbaum Rene	University of California at Davis, USA
Rushmeier	Holly, Yale University, USA
Sander Pedro	The Hong Kong University of Science and Technology, Hong Kong
Sapidis Nickolas	University of Western Macedonia, Greece
Sarfraz Muhammad	Kuwait University, Kuwait
Scateni Riccardo	University of Cagliari, Italy
Schaefer Scott	Texas A&M University, USA
Sequin Carlo	University of California-Berkeley, USA
Shead Timothy	Sandia National Laboratories, USA
Sorkine Olga	New York University, USA
Sourin Alexei	Nanyang Technological University, Singapore
Stamminger Marc	REVES/INRIA, France
Su Wen-Poh	Griffith University, Australia
Staadt Oliver	University of Rostock, Germany
Tarini Marco	Università dell'Insubria (Varese), Italy
Teschner Matthias	University of Freiburg, Germany
Tsong Ng Tian	Institute for Infocomm Research, Singapore
Umlauf Georg	HTWG Constance, Germany
Wald Ingo	University of Utah, USA
Wang Sen	Kodak, USA
Wimmer Michael	Technical University of Vienna, Austria
Wylie Brian	Sandia National Laboratory, USA
Wyman Chris	University of Iowa, USA
Yang Qing-Xiong	University of Illinois at Urbana, Champaign, USA
Yang Ruigang	University of Kentucky, USA
Ye Duan	University of Missouri-Columbia, USA
Yi Beifang	Salem State College, USA
Yin Lijun	Binghamton University, USA

Yoo Terry  
Yuan Xiaoru  
Zabulis Xenophon

Zhang Eugene  
Zhang Jian Jun  
Zordan Victor

National Institutes of Health, USA  
Peking University, China  
Foundation for Research and  
Technology - Hellas (FORTH), Greece  
Oregon State University, USA  
Bournemouth University, UK  
University of California at Riverside, USA

### (Area 3) Virtual Reality

Alcañiz Mariano  
Arns Laura  
Balcisoy Selim  
Behringer Reinhold  
Benes Bedrich  
Bilalis Nicholas  
Blach Roland

Blom Kristopher  
Borst Christoph  
Brady Rachael  
Brega Jose Remo Ferreira  
Brown Ross

Bruce Thomas

Bues Matthias  
Chen Jian  
Cheng Irene  
Coquillart Sabine  
Craig Alan

Cremer Jim  
Egges Arjan  
Encarnacao L. Miguel  
Figueroa Pablo  
Fox Jesse  
Friedman Doron  
Froehlich Bernd  
Gregory Michelle  
Gupta Satyandra K.  
Hachet Martin  
Haller Michael  
Hamza-Lup Felix  
Hinkenjann Andre

Technical University of Valencia, Spain  
Purdue University, USA  
Sabanci University, Turkey  
Leeds Metropolitan University UK  
Purdue University, USA  
Technical University of Crete, Greece  
Fraunhofer Institute for Industrial  
Engineering, Germany  
University of Hamburg, Germany  
University of Louisiana at Lafayette, USA  
Duke University, USA  
Universidade Estadual Paulista, Brazil  
Queensland University of Technology,  
Australia

The University of South Australia,  
Australia

Fraunhofer IAO in Stuttgart, Germany  
Brown University, USA  
University of Alberta, Canada  
INRIA, France  
NCSA University of Illinois at  
Urbana-Champaign, USA

University of Iowa, USA  
Universiteit Utrecht, The Netherlands  
Humana Inc., USA  
Universidad de los Andes, Colombia  
Stanford University, USA

IDC, Israel  
Weimar University, Germany  
Pacific Northwest National Lab, USA  
University of Maryland, USA  
INRIA, France  
FH Hagenberg, Austria  
Armstrong Atlantic State University, USA  
Bonn-Rhein-Sieg University of Applied  
Sciences, Germany

Hollerer Tobias	University of California at Santa Barbara, USA
Huang Jian	University of Tennessee at Knoxville, USA
Julier Simon J.	University College London, UK
Klinker Gudrun	Technische Universität München, Germany
Klosowski James	AT&T Labs, USA
Kozintsev	Igor, Intel, USA
Kuhlen Torsten	RWTH Aachen University, Germany
Liere Robert van	CWI, The Netherlands
Majumder Aditi	University of California, Irvine, USA
Malzbender Tom	Hewlett Packard Labs, USA
Mantler Stephan	VRVis Research Center, Austria
Meyer Joerg	University of California, Irvine, USA
Molineros Jose	Teledyne Scientific and Imaging, USA
Muller Stefan	University of Koblenz, Germany
Paelke Volker	Leibniz Universität Hannover, Germany
Pan Zhigeng	Zhejiang University, China
Papka Michael	Argonne National Laboratory, USA
Peli Eli	Harvard University, USA
Pettifer Steve	The University of Manchester, UK
Pugmire Dave	Los Alamos National Lab, USA
Qian Gang	Arizona State University, USA
Raffin Bruno	INRIA, France
Reiners Dirk	University of Louisiana, USA
Richir Simon	Arts et Metiers ParisTech, France
Rodello Ildeberto	University of Sao Paulo, Brazil
Santhanam Anand	MD Anderson Cancer Center Orlando, USA
Sapidis Nickolas	University of Western Macedonia, Greece
Schulze	Jurgen, University of California - San Diego, USA
Sherman Bill	Jurgen, Indiana University, USA
Slavik Pavel	Czech Technical University in Prague, Czech Republic
Sourin Alexei	Nanyang Technological University, Singapore
Stammering Marc	REVES/INRIA, France
Srikanth Manohar	Indian Institute of Science, India
Staadt Oliver	University of Rostock, Germany
Swan Ed	Mississippi State University, USA
Stefani Oliver	COAT-Basel, Switzerland
Sun Hanqiu	The Chinese University of Hong Kong, Hong Kong
Varsamidis Thomas	Bangor University, UK
Vercher Jean-Louis	Université de la Méditerranée, France
Wald Ingo	University of Utah, USA

## XVIII Organization

Yu Ka Chun

Yuan Chunrong

Zachmann Gabriel

Zara Jiri

Zhang Hui

Zhao Ye

Zyda Michael

Denver Museum of Nature and Science,  
USA

University of Tuebingen, Germany

Clausthal University, Germany

Czech Technical University in Prague,  
Czech Republic

Indiana University, USA

Kent State University, USA

University of Southern California, USA

### (Area 4) Visualization

Andrienko Gennady

Apperley Mark

Balázs Csébfalvi

Bartoli Anna Vilanova

Brady Rachael

Benes Bedrich

Bilalis Nicholas

Bonneau Georges-Pierre

Brown Ross

Bühler Katja

Callahan Steven

Chen Jian

Chen Min

Cheng Irene

Chiang Yi-Jen

Chourasia Amit

Coming Daniel

Dana Kristin

Dick Christian

DiVerdi Stephen

Doleisch Helmut

Duan Ye

Dwyer Tim

Ebert David

Entezari Alireza

Ertl Thomas

Floriani Leila De

Fujishiro Issei

Geist Robert

Goebel Randy

Fraunhofer Institute IAIS, Germany

University of Waikato, New Zealand

Budapest University of Technology and  
Economics, Hungary

Eindhoven University of Technology,  
The Netherlands

Duke University, USA

Purdue University, USA

Technical University of Crete, Greece

Grenoble Université , France

Queensland University of Technology,  
Australia

VRVIS, Austria

University of Utah, USA

Brown University, USA

University of Wales Swansea, UK

University of Alberta, Canada

Polytechnic Institute of New York  
University, USA

University of California - San Diego, USA

Desert Research Institute, USA

Rutgers University, USA

Technical University of Munich, Germany  
Adobe, USA

VRVis Research Center, Austria

University of Missouri-Columbia, USA

Monash University, Australia

Purdue University, USA

University of Florida, USA

University of Stuttgart, Germany

University of Maryland, USA

Keio University, Japan

Clemson University, USA

University of Alberta, Canada

Gotz David	IBM, USA
Grinstein Georges	University of Massachusetts Lowell, USA
Goebel Randy	University of Alberta, Canada
Gregory Michelle	Pacific Northwest National Lab, USA
Hadwiger Helmut Markus	VRVis Research Center, Austria
Hagen Hans	Technical University of Kaiserslautern, Germany
Hamza-Lup Felix	Armstrong Atlantic State University, USA
Heer Jeffrey	Armstrong University of California at Berkeley, USA
Hege Hans-Christian	Zuse Institute Berlin, Germany
Hochheiser Harry	University of Pittsburgh, USA
Hollerer Tobias	University of California at Santa Barbara, USA
Hong Lichan	Palo Alto Research Center, USA
Hotz Ingrid	Zuse Institute Berlin, Germany
Jiang Ming	Lawrence Livermore National Laboratory, USA
Joshi Alark	Yale University, USA
Julier Simon J.	University College London, UK
Kohlhammer Jörn	Fraunhofer Institut, Germany
Kosara Robert	University of North Carolina at Charlotte, USA
Laramee Robert	Swansea University, UK
Lee Chang Ha	Chung-Ang University, Korea
Lewis Bob	Washington State University, USA
Liere Robert van	CWI, The Netherlands
Lim Ik Soo	Bangor University, UK
Linsen Lars	Jacobs University, Germany
Liu Zhanping	Kitware, Inc., USA
Ma Kwan-Liu	University of California-Davis, USA
Maeder Anthony	University of Western Sydney, Australia
Majumder Aditi	University of California, Irvine, USA
Malpica Jose	Alcala University, Spain
Masutani Yoshitaka	The University of Tokyo Hospital, Japan
Matkovic Kresimir	VRVis Forschungs-GmbH, Austria
McCaffrey James	Microsoft Research / Volt VTE, USA
McGraw Tim	West Virginia University, USA
Melançon Guy	CNRS UMR 5800 LaBRI and INRIA Bordeaux Sud-Ouest, France
Meyer Joerg	University of California, Irvine, USA
Miksch Silvia	Vienna University of Technology, Austria
Monroe Laura	Los Alamos National Labs, USA
Morie Jacki	University of Southern California, USA

Mueller Klaus	SUNY Stony Brook, USA
Museth Ken	Linköping University, Sweden
Paelke Volker	Leibniz Universität Hannover, Germany
Papka Michael	Argonne National Laboratory, USA
Pettifer Steve	The University of Manchester, UK
Pugmire Dave	Los Alamos National Lab, USA
Rabin Robert	University of Wisconsin at Madison, USA
Raffin Bruno	INRIA, France
Razdan Anshuman	Arizona State University, USA
Rhyne Theresa-Marie	North Carolina State University, USA
Rosenbaum Rene	University of California at Davis, USA
Santhanam Anand	MD Anderson Cancer Center Orlando, USA
Scheuermann Gerik	University of Leipzig, Germany
Shead Timothy	Sandia National Laboratories, USA
Shen Han-Wei	Ohio State University, USA
Silva Claudio	University of Utah, USA
Sips Mike	Stanford University, USA
Slavik Pavel	Czech Technical University in Prague, Czech Republic
Sourin Alexei	Nanyang Technological University, Singapore
Swan Ed	Mississippi State University, USA
Theisel Holger	University of Magdeburg, Germany
Thiele Olaf	University of Mannheim, Germany
Toledo de Rodrigo	Petrobras PUC-RIO, Brazil
Tricoche Xavier	Purdue University, USA
Umlauf Georg	HTWG Constance, Germany
Viegas Fernanda	IBM, USA
Wald Ingo	University of Utah, USA
Wan Ming	Boeing Phantom Works, USA
Weinkauf Tino	Courant Institute, New York University, USA
Weiskopf Daniel	University of Stuttgart, Germany
Wischgoll Thomas	Wright State University, USA
Wylie Brian	Sandia National Laboratory, USA
Yeasin Mohammed	Memphis University, USA
Yuan Xiaoru	Peking University, China
Zachmann Gabriel	Clausthal University, Germany
Zhang Eugene	Oregon State University, USA
Zhang Hui	Indiana University, USA
Zhao Ye	Kent State University, USA
Zhukov Leonid	Caltech, USA

## ISVC 2010 Special Tracks

### 1. 3D Mapping, Modeling and Surface Reconstruction

#### Organizers

Nefian Ara	Carnegie Mellon University/NASA Ames Research Center, USA
Broxton Michael	Carnegie Mellon University/NASA Ames Research Center, USA
Huertas Andres	NASA Jet Propulsion Lab, USA

#### Program Committee

Hancher Matthew	NASA Ames Research Center, USA
Edwards Laurence	NASA Ames Research Center, USA
Bradski Garry	Willow Garage, USA
Zakhor Avideh	University of California at Berkeley, USA
Cavallaro Andrea	University Queen Mary, London, UK
Bouquet Jean-Yves	Google, USA

### 2. Best Practices in Teaching Visual Computing

#### Organizers

Albu Alexandra Branzan	University of Victoria, Canada
Bebis George	University of Nevada, Reno, USA

#### Program Committee

Bergevin Robert	University of Laval, Canada
Crawfis Roger	Ohio State University, USA
Hammoud Riad	DynaVox Systems, USA
Kakadiaris Ioannis	University of Houston, USA, USA
Laurendeau Denis	Laval University, Quebec, Canada
Maxwell Bruce	Colby College, USA
Stockman George	Michigan State University, USA

### 3. Low-Level Color Image Processing

#### Organizers

Celebi M. Emre	Louisiana State University, USA
Smolka Bogdan	Silesian University of Technology, Poland
Schaefer Gerald	Loughborough University, UK
Plataniotis Konstantinos	University of Toronto, Canada
Horiuchi Takahiko	Chiba University, Japan

### Program Committee

Aygun Ramazan	University of Alabama in Huntsville, USA
Battiato Sebastiano	University of Catania, Italy
Hardeberg Jon	Gjøvik University College, Norway
Hwang Sae	University of Illinois at Springfield, USA
Kawulok Michael	Silesian University of Technology, Poland
Kockara Sinan	University of Central Arkansas, USA
Kotera Hiroaki	Kotera Imaging Laboratory, Japan
Lee JeongKyu	University of Bridgeport, USA
Lezoray Olivier	University of Caen, France
Mete Mutlu	Texas A&M University - Commerce, USA
Susstrunk Sabine	Swiss Federal Institute of Technology in Lausanne, Switzerland
Tavares Joao	University of Porto, Portugal
Tian Gui Yun	Newcastle University, UK
Wen Quan	University of Electronic Science and Technology of China, China
Zhou Huiyu	Queen's University Belfast, UK

### 4. Low Cost Virtual Reality: Expanding Horizons

#### Organizers

Sherman Bill	Indiana University, USA
Wernert Eric	Indiana University, USA

#### Program Committee

Coming Daniel	Desert Research Institute, USA
Craig Alan	University of Illinois/NCSA, USA
Keefe Daniel	University of Minnesota, USA
Kreylos Oliver	University of California at Davis, USA
O'Leary Patrick	Idaho National Laboratory, USA
Smith Randy	Oakland University, USA
Su Simon	Princeton University, USA
Will Jeffrey	Valparaiso University, USA

### 5. Computational Bioimaging

#### Organizers

Tavares João Manuel R. S.	University of Porto, Portugal
Jorge Renato Natal	University of Porto, Portugal
Cunha Alexandre	Caltech, USA

## Program Committee

Santis De Alberto	Università degli Studi di Roma “La Sapienza”, Italy
Reis Ana Mafalda	Instituto de Ciencias Biomedicas Abel Salazar, Portugal
Barrutia Arrate Muñoz	University of Navarra, Spain
Calvo Begoña	University of Zaragoza, Spain
Constantinou Christos	Stanford University, USA
Iacoviello Daniela	Università degli Studi di Roma “La Sapienza”, Italy
Ushizima Daniela	Lawrence Berkeley National Lab, USA
Ziou Djemel	University of Sherbrooke, Canada
Pires Eduardo Borges	Instituto Superior Tecnico, Portugal
Sgallari Fiorella	University of Bologna, Italy
Perales Francisco	Balearic Islands University, Spain
Qiu Guoping	University of Nottingham, UK
Hanchuan Peng	Howard Hughes Medical Institute, USA
Pistori Hemerson	Dom Bosco Catholic University, Brazil
Yanovsky Igor	Jet Propulsion Laboratory, USA
Corso Jason	SUNY at Buffalo, USA
Maldonado Javier Melenchón	Open University of Catalonia, Spain
Marques Jorge S.	Instituto Superior Tecnico, Portugal
Aznar Jose M. García	University of Zaragoza, Spain
Vese Luminita	University of California at Los Angeles, USA
Reis Luís Paulo	University of Porto, Portugal
Thiriet Marc	Université Pierre et Marie Curie (Paris VI), France
Mahmoud El-Sakka	The University of Western Ontario London, Canada
Hidalgo Manuel González	Balearic Islands University, Spain
Gurcan Metin N.	Ohio State University, USA
Dubois Patrick	Institut de Technologie Médicale, France
Barneva Reneta P.	State University of New York, USA
Bellotti Roberto	University of Bari, Italy
Tangaro Sabina	University of Bari, Italy
Silva Susana Branco	University of Lisbon, Portugal
Brimkov Valentin	State University of New York, USA
Zhan Yongjie	Carnegie Mellon University, USA

## 6. Unconstrained Biometrics: Advances and Trends

### Organizers

Proença Hugo	University of Beira Interior, Portugal
Du Yingzi	Indiana University-Purdue University Indianapolis, USA

Scharcanski Jacob

Federal University of Rio Grande do Sul

Porto Alegre, Brazil

Ross Arun

West Virginia University, USA

Amayeh Gholamreza

EyeCom Corporation, USA

### **Program Committee**

Júnior Adalberto Schuck

Federal University of Rio Grande do Sul,  
Brazil

Kwolek Bogdan

Rzeszow University of Technology, Poland  
Federal University of Rio Grande do Sul,  
Brazil

Jung Cláudio R.

Alirezaie Javad

Ryerson University, Canada

Konrad Janusz

Boston University, USA

Kevin Jia

International Game Technologies, USA

Meyer Joceli

Federal University of Santa Catarina, Brazil

Alexandre Luís A.

University of Beira Interior, Portugal

Soares Luis

ISCTE, Portugal

Coimbra Miguel

University of Porto, Portugal

Fieguth Paul

University of Waterloo, Canada

Xiao Qinghan

Defense Research and Development

Canada, Canada

Ives Robert

United States Naval Academy, USA

Tamir Samir

Ingersoll Rand Security, USA

## **7. Behavior Detection and Modeling**

### **Organizers**

Miller Ron

Wright-Patterson Air Force Base, USA

Bebis George

University of Nevada, USA

Rosen Julie

Science Applications International

Corporation, USA

Davis Jim

Ohio State University, USA

Lee Simon

Army Research Laboratory, USA

Zandipour Majid

BAE Systems, USA

## Organizing Institutions and Sponsors



imagination at work



# Table of Contents – Part III

## Poster Session

Lunar Image Classification for Terrain Detection .....	1
<i>Heng-Tze Cheng, Feng-Tso Sun, Senaka Buthpitiya,     Ying Zhang, and Ara V. Nefian</i>	
Surface Modeling of the Corpus Callosum from MRI Scans .....	9
<i>Ahmed Farag, Shireen Elhabian, Mostafa Abdelrahman,     James Graham, Aly Farag, Dongqing Chen, and Manuel F. Casanova</i>	
Track Detection for Autonomous Trains .....	19
<i>Michael Gschwandtner, Wolfgang Pree, and Andreas Uhl</i>	
Local Descriptors for Document Layout Analysis .....	29
<i>Angelika Garz, Markus Diem, and Robert Sablatnig</i>	
CT Image Segmentation Using Structural Analysis .....	39
<i>Hiroyuki Hishida, Takashi Michikawa, Yutaka Otake,     Hiromasa Suzuki, and Satoshi Oota</i>	
Phase Space for Face Pose Estimation .....	49
<i>Jacob Foytik, Vijayan K. Asari, R. Cortland Tompkins, and     Menatoallah Youssef</i>	
Contour Based Shape Retrieval .....	59
<i>Levente Kovács</i>	
Illumination Normalization for Robust Face Recognition Using Discrete Wavelet Transform .....	69
<i>Amnart Petpon and Sanun Srisuk</i>	
Feature-Based Lung Nodule Classification .....	79
<i>Amal Farag, Asem Ali, James Graham, Shireen Elhabian,     Aly Farag, and Robert Falk</i>	
Multiple-Object Tracking in Cluttered and Crowded Public Spaces .....	89
<i>Rhys Martin and Ognjen Arandjelović</i>	
Compliant Interframe Coding for Motion-JPEG2000 .....	99
<i>René Rosenbaum and Heidrun Schumann</i>	
EVP-Based Multiple-View Triangulation .....	109
<i>G. Chesi and Y.S. Hung</i>	

An Improved Shape Matching Algorithm for Deformable Objects Using a Global Image Feature .....	119
<i>Jibum Kim and Suzanne M. Shontz</i>	
Multi-scale Topo-morphometric Opening of Arteries and Veins: An Evaluative Study via Pulmonary CT Imaging .....	129
<i>Zhiyun Gao, Colin Holtze, Randall Grout, Milan Sonka, Eric Hoffman, and Punam K. Saha</i>	
Video Event Detection as Matching of Spatiotemporal Projection .....	139
<i>Dong-Jun Park and David Eichmann</i>	
PixelLaser: Computing Range from Monocular Texture .....	151
<i>N. Lesperance, M. Leece, S. Matsumoto, M. Korbel, K. Lei, and Z. Dodds</i>	
A Spatio-spectral Algorithm for Robust and Scalable Object Tracking in Videos .....	161
<i>Alireza Tavakkoli, Mircea Nicolescu, and George Bebis</i>	
Driving Fatigue Detection Using Active Shape Models .....	171
<i>Hernán García, Augusto Salazar, Damián Alvarez, and Álvaro Orozco</i>	
Outlier Removal in Stereo Reconstruction of Orbital Images .....	181
<i>Marvin Smith and Ara Nefian</i>	
Random Sampling Nonlinear Optimization for Camera Self-calibration with Modeling of Intrinsic Parameter Space .....	189
<i>Houman Rastgar, Eric Dubois, and Liang Zhang</i>	
Facial Fraud Discrimination Using Detection and Classification .....	199
<i>Inho Choi and Daijin Kim</i>	
Segmentation of Abdominal Organs Incorporating Prior Knowledge in Small Animal CT .....	209
<i>SooMin Song and Myoung-Hee Kim</i>	
Method of Interest Points Characterization Based C-HOG Local Descriptor .....	219
<i>Manuel Grand-brochier, Christophe Tilmant, and Michel Dhome</i>	
Stereo-Based Object Segmentation Combining Spatio-temporal Information .....	229
<i>Yingdong Ma and Qian Chen</i>	
Fast Motion Estimation Based on Search Range Adjustment Using Neighboring MVDs .....	239
<i>Hyun-Soo Kang and Jae-Hyeung Park</i>	

Towards Computational Understanding of Skill Levels in Simulation-Based Surgical Training via Automatic Video Analysis.....	249
<i>Qiang Zhang and Baoxin Li</i>	
Biomedical Image Retrieval in a Fuzzy Feature Space with Affine Region Detection and Vector Quantization of a Scale-Invariant Descriptor .....	261
<i>Md Mahmudur Rahman, Sameer K. Antani, and George R. Thoma</i>	
Model Distribution Dependant Complexity Estimation on Textures.....	271
<i>Agustín Mailing, Tomás Crivelli, and Bruno Cernuschi-Frías</i>	
Integrating Multiple Uncalibrated Views for Human 3D Pose Estimation .....	280
<i>Zibin Wang and Ronald Chung</i>	
A Novel Histogram-Based Feature Representation and Its Application in Sport Players Classification .....	291
<i>Paolo Spagnolo, Pier Luigi Mazzeo, Marco Leo, and Tiziana D’Orazio</i>	
Facial Expression Recognition Using Facial Features and Manifold Learning .....	301
<i>Raymond Ptucha and Andreas Savakis</i>	
Blurring Mean-Shift with a Restricted Data-Set Modification for Applications in Image Processing .....	310
<i>Eduard Sojka, Jan Gaura, Štepán Šrubař, Tomáš Fabián, and Michal Krumník</i>	
Detecting Straight Line Segments Using a Triangular Neighborhood ....	320
<i>Shengzhi Du, Chunling Tu, and Barend Jacobus van Wyk</i>	
Size Distribution Estimation of Stone Fragments via Digital Image Processing .....	329
<i>Mohammad Salehizadeh and Mohammad T. Sadeghi</i>	
Image Enhancement by Median Filters in Algebraic Reconstruction Methods: An Experimental Study .....	339
<i>Norbert Hantos and Péter Balázs</i>	
3D Curvature-Based Shape Descriptors for Face Segmentation: An Anatomical-Based Analysis .....	349
<i>Augusto Salazar, Alexander Cerón, and Flavio Prieto</i>	
Computational Hemodynamics in Intracranial Vessels Reconstructed from Biplane Angiograms .....	359
<i>Fabien Scalzo, Qing Hao, Alan M. Walczak, Xiao Hu, Yiemeng Hoi, Kenneth R. Hoffmann, and David S. Liebeskind</i>	

Object Distance Estimation Based on Stereo Vision and Color Segmentation with Region Matching . . . . .	368
<i>Guangming Xiong, Xin Li, Junqiang Xi, Spencer G. Fowers, and Huiyan Chen</i>	
Multiscale Information Fusion by Graph Cut through Convex Optimization . . . . .	377
<i>Yinhui Zhang, Yunsheng Zhang, and Zifen He</i>	
A Fast Level Set-Like Algorithm for Region-Based Active Contours . . . . .	387
<i>Martin Maška, Pavel Matula, Ondřej Daněk, and Michal Kozubek</i>	
A Novel Hardware Architecture for Rapid Object Detection Based on AdaBoost Algorithm . . . . .	397
<i>Tinghui Wang, Feng Zhao, Jiang Wan, and Yongxin Zhu</i>	
Using Perceptual Color Contrast for Color Image Processing . . . . .	407
<i>Guangming Xiong, Dah-Jye Lee, Spencer G. Fowers, Jianwei Gong, and Huiyan Chen</i>	
GPU Acceleration of Robust Point Matching . . . . .	417
<i>Chad Mourning, Scott Nykl, Huihui Xu, David Chelberg, and Jundong Liu</i>	
A Wavelet-Based Face Recognition System Using Partial Information . . . . .	427
<i>H.F. Neo, C.C. Teo, and Andrew B.J. Teoh</i>	
A Study of Hierarchical Correlation Clustering for Scientific Volume Data . . . . .	437
<i>Yi Gu and Chaoli Wang</i>	
Subversion Statistics Sifter . . . . .	447
<i>Christoph Müller, Guido Reina, Michael Burch, and Daniel Weiskopf</i>	
A Lossy/Lossless Coding Algorithm Using Histogram . . . . .	458
<i>Sunil Bhooshan and Shipra Sharma</i>	
Stereo Matching in Mean Shift Attractor Space . . . . .	465
<i>Michał Krumnikl</i>	
Undecimated Wavelet Transform-Based Image Interpolation . . . . .	474
<i>Numan Unaldi and Vijayan K. Asari</i>	
The Influence of Multimodal 3D Visualizations on Learning Acquisition . . . . .	484
<i>Phuong T. Do, John R. Moreland, and Dennis P. Korcheck</i>	
Visualizing Gene Co-expression as Google Maps . . . . .	494
<i>Radu Jianu and David H. Laidlaw</i>	

A New Approach for Lighting Effect Rendering . . . . .	504
<i>Catherine Sauvaget and Vincent Boyer</i>	
SemaTime – Timeline Visualization of Time-Dependent Relations and Semantics . . . . .	514
<i>Christian Stab, Kawa Nazemi, and Dieter W. Fellner</i>	
Comics Stylizations of 3D Scenes Using GPU . . . . .	524
<i>Jordane Suarez, Farès Belhadj, and Vincent Boyer</i>	
Discovering Novelty in Gene Data: From Sequential Patterns to Visualization . . . . .	534
<i>Arnaud Sallaberry, Nicolas Pecheur, Sandra Bringay, Mathieu Roche, and Maguelonne Teisseire</i>	
A Differential-Geometrical Framework for Color Image Quality Measures . . . . .	544
<i>Mourad Zéraï and Olfa Triki</i>	
Three Dimensional Reconstruction Using Vertical Constraints from a Photograph . . . . .	554
<i>Satoru Morita</i>	
A Framework for Visual and Haptic Collaboration in Shared Virtual Spaces . . . . .	564
<i>Lei Wei, Alexei Sourin, and Herbert Stocker</i>	
Design and Costs Estimation of Electrical Substations Based on Three-Dimensional Building Blocks . . . . .	574
<i>Eduardo Islas Pérez, Jessica Bahena Rada, Jesus Romero Lima, and Mirna Molina Marín</i>	
Generating Shaded Image with Lighting Using Image Fusion Space . . . . .	584
<i>Satoru Morita</i>	
Automatic Detection of Morphologically Distinct Objects in Biomedical Images Using Second Generation Wavelets and Multiple Marked Point Process . . . . .	594
<i>Hiroshi Hatsuda</i>	
Imaging-Based Computation of the Dynamics of Pelvic Floor Deformation and Strain Visualization Analysis . . . . .	604
<i>Christos E. Constantinou, Linda McLean, Ellen Kuhl, and Bertha Chen</i>	
Exploiting Multiple Cameras for Environmental Pathlets . . . . .	613
<i>Kevin Streib and James W. Davis</i>	

On Supervised Human Activity Analysis for Structured Environments . . . . .	625
<i>Banafshe Arbab-Zavar, Imed Bouchrika, John N. Carter, and Mark S. Nixon</i>	
Human Behavior Analysis at a Point of Sale . . . . .	635
<i>R. Sicre and H. Nicolas</i>	
<b>Author Index . . . . .</b>	<b>645</b>

# Lunar Image Classification for Terrain Detection

Heng-Tze Cheng<sup>1</sup>, Feng-Tso Sun<sup>1</sup>, Senaka Buthpitiya<sup>1</sup>,  
Ying Zhang<sup>1</sup>, and Ara V. Nefian<sup>2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Carnegie Mellon University

<sup>2</sup> Intelligent Robotics Group, NASA Ames Research Center

{hengtze.cheng, lucas.sun, senaka.buthpitiya}@sv.cmu.edu,  
joy.zhang@sv.cmu.edu, ara.nefian@nasa.gov

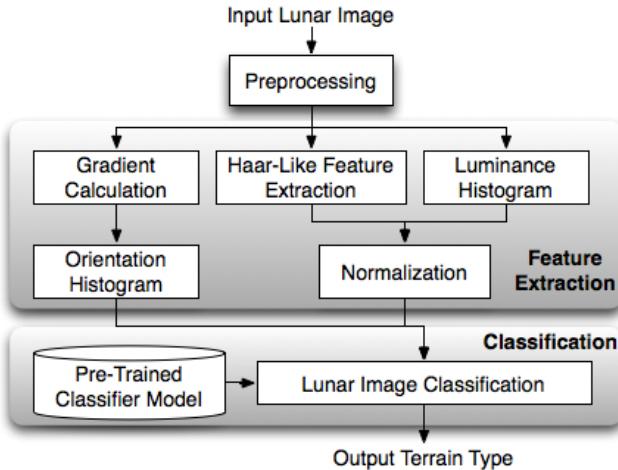
**Abstract.** Terrain detection and classification are critical elements for NASA mission preparations and landing site selection. In this paper, we have investigated several image features and classifiers for lunar terrain classification. The proposed histogram of gradient orientation effectively discerns the characteristics of various terrain types. We further develop an open-source Lunar Image Labeling Toolkit to facilitate future research in planetary science. Experimental results show that the proposed system achieves 95% accuracy of classification evaluated on a dataset of 931 lunar image patches from NASA Apollo missions.

## 1 Introduction

Due to the rapid growth of image acquisition technology, large amount of planetary images from satellites are available for aerospace and planetary research. Understanding the semantic content in orbital images is therefore important for further analysis in planetary science. For example, terrain classification can facilitate landing site selection in robotic or manned mission on the Moon or Mars. In this work, we focus on lunar image classification in images collected by Apollo missions.

Lunar image classification is challenging because the shapes and colors of each type of terrain vary with locations, and the images are affected by dust and granular noise in the scanning process. Furthermore, the brightness and shadow caused by the angle of sunlight can significantly change the appearance of lunar surface. Inspired by previous work on face detection and object recognition [2]–[5], we combine both the techniques in image processing and the domain knowledge in planetary science to overcome the challenges.

In this paper, we present an automatic lunar image classification system with the following contributions. First, we have investigated multiple combinations of features and classifiers and compare their effectiveness in classifying terrains including craters, flat areas, and shadows. Second, we show that the proposed image feature—histogram of gradient orientation—is robust and effective for differentiating signatures of different terrains on the Moon. The system is one of the early attempts [6]–[8] to address the challenges in lunar terrain classification, and can be applied to images from the Moon, Mars, or other planets in space missions (e.g. LCROSS and HiRISE). Third, to facilitate future research in planetary science and imagery, we develop an open-source Lunar Image Labeling Toolkit (LILT) with graphical user interface



**Fig. 1.** System architecture of lunar image classification

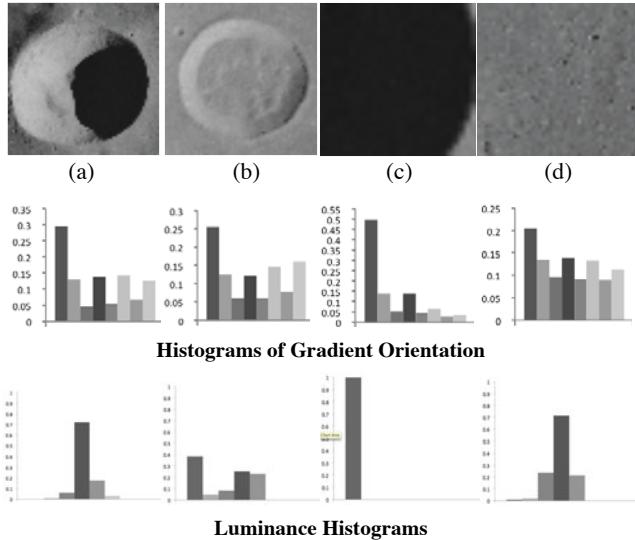
implemented in Java. We hope researchers can utilize the LILT toolkit to efficiently annotate large amount of planetary images for supervised learning.

The paper is organized as follows. We discuss related work in Section 2. In Section 3, we describe the proposed system architecture including feature extraction and classification algorithms. We present the lunar image labeling toolkit in Section 4, and show the experimental results in Section 5. We conclude the paper in Section 6.

## 2 Related Work

Our work is inspired by several previous attempts to crater or geological feature detection [6]–[8]. In [6], a feature template family is created by varying the resizing scale of continuously scalable detectors, and a testing image is classified as the most correlated template. The work has limitations when images are noisy or taken under varying sunlight angle. In [7], object-based analysis is used to extract textually or spectrally homogeneous regions. The method allows the identification of crater walls and crater floors separately; however, it requires high learning costs and is susceptible to noise. In [8], automatic crater detection is implemented using a boosting algorithm that select a small set of features characterizing the presence or absence of craters. The approach is validated by data sets of Mars surface captured by Mars Orbiter Camera.

There has also been extensive work on face, human, or object recognition [1]. In [3], rectangle features are used as a weak classifier initially, and then enhanced by incremental training using AdaBoost. A cascade filter is further used to discard the background of images rapidly and refine the detection results. In [4], object recognition is modeled as a machine translation problem. K-means is used for clustering followed by the EM algorithm for learning a lexicon of image words. Though some promising results are reported, it is mentioned that the approaches have certain limitations and does not work well on some objects. Locally normalized histogram of



**Fig. 2.** Examples of lunar terrain types and the corresponding feature examples at the bottom: (a) crater with shadow (b) crater without shadow (c) shadow area (d) flat area

gradient orientations is adopted in [2] for person detection. The feature reduces false positive rates by an order of magnitude relative to the best Haar wavelet based detector from [5]. Inspired by these related works, we aim at lunar terrain classification and evaluate our system using various features and classifiers.

### 3 System Design and Methods

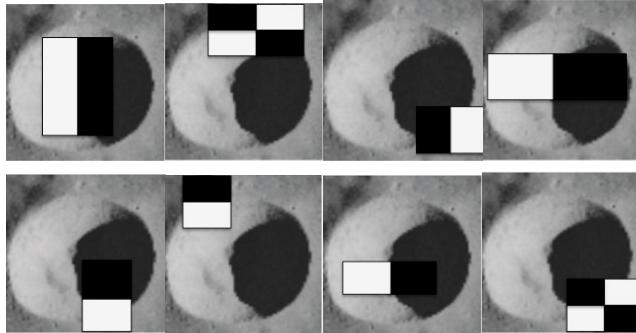
The system architecture is shown in Fig. 1. The goal of our system is to classify an input image patch to one of the four classes shown in Fig. 2. In this section, we describe each component of our approach in detail.

#### 3.1 Preprocessing

First, all the images are converted to grayscale since lunar images mainly differ in luminance rather than color distributions. Then, for each input image, we apply Gaussian blurring filter to the entire image to reduce high-frequency noise and camera artifacts. Gaussian filtering is done by the convolution of each pixel in the input image with a Gaussian kernel and then summing to produce the output image. In experiments, we empirically use the 3-by-3 filter window as the Gaussian kernel.

#### 3.2 Image Feature Extraction

**Histogram of Gradient Orientation.** Histogram of gradient orientation has been used in the field of human detection [2]. While gradient is useful for edge detection, using the magnitude of gradient as a feature is less robust due to the variability of



**Fig. 3.** The eight Haar-like masks used for feature extraction

luminance contrast. In contrast with magnitude, the orientation of gradient is of lower granularity and thus less susceptible to high-frequency noises. The computation of gradient orientation can be formulated as:

$$\mathbf{v}^* = \arg \max_{\mathbf{v}} |L(n(p_i, \mathbf{v})) - L(p_i)| \quad (1)$$

where  $\mathbf{v} \in \{(i, j) \mid -1 \leq i \leq 1, -1 \leq j \leq 1, i, j \in \mathbb{Z}\}$  is the orientation of the gradient, which can be one of the following vectors: (1,0), (1,1), (0,1), (-1,1), (-1,0), (-1,-1), (0,-1), (1,-1).  $p_i$  denotes the coordinate of a pixel, and  $n(p_i, \mathbf{v})$  denotes the neighboring pixel given by  $(p_i + \mathbf{v})$ .  $L(p_i)$  is the luminance of the pixel  $p_i$ . According to eq. (1), for each pixel  $p_i$  in the input image patch, we compute the difference of luminance between the pixel  $p_i$  and each of its eight neighboring pixels,  $n(p_i, \mathbf{v})$ . The direction with the maximum difference is the gradient orientation. After computing the orientation of gradient at each pixel, we sum up the occurrence of each orientation and form a statistical histogram. Since there are eight possible directions, the resulting histogram is an 8-dimensional vector as shown in Figure 2.

The intuition behind the use of this feature is inspired by the characteristics of lunar terrains. While craters are generally of circular shape, flat areas have no specific shape or edges, and shadow areas can have obvious edges from darkness to the area with sunlight. These characteristics can be distinguished by the gradient orientation histograms of image patches.

**Haar-like Features.** Haar-like features are widely used in object and face detection. The features are computed by probing each image patch  $P$  with a set of 8 different masks shown in Figure 3. In these masks, the black and white areas represent -1 and +1, respectively. When we apply a Haar-like feature mask on a specific image patch  $P$ , the computation of a single normalized feature value can be formulated as:

$$f(P) = \int_{[0,1]^2} P(x)\Omega(x)dx \quad (2)$$

where  $\Omega(x)$  denotes one of the Haar-like masks. The image patch  $P$  and the mask are images defined in the interval  $[0,1]^2$ . The computation of these features is intensive but can be accelerated if the integral image is pre-computed. In our implementation, we applied the 8 distinct Haar-like feature masks and aggregated eight values into an 8-dimensional feature vector.

**Luminance Histogram.** The luminance histogram represents the luminance distribution of the image. The raw luminance histogram is of 256 bins. To reduce the dimension, we cluster every 32 successive luminance levels into one bin and obtain an 8-dimensional feature vector as shown in Figure 2.

### 3.3 Lunar Image Classification

**K-Nearest Neighbor Classifier.** In our work, we adopt  $k$ -NN as one of our classification algorithms. Specifically, for each testing image patch, we calculate the distance from each of the image patches in the training dataset. The distance is obtained by computing the L2-distance (i.e. Euclidean distance) between two feature vectors.

**Decision Tree Classifier.** In addition to  $k$ -NN classifier, we also adopt the decision tree classifier. Specifically, a decision tree is tiered into three layers. The topmost tier's decision tree decides if a particular patch belongs to the most common class in the training data. If positive, the classification is finished. Otherwise, the decision tree at the next tier is used to classify the patch as belonging to the second most common class in the training data or not. This process is repeated until a decision tree returns a positive result or all decision trees are exhausted. In cases where all decision trees are exhausted without obtaining a positive classification from any of the decision trees, the patch is assigned a default classification. The default classification in this case is the most common class in the training data.

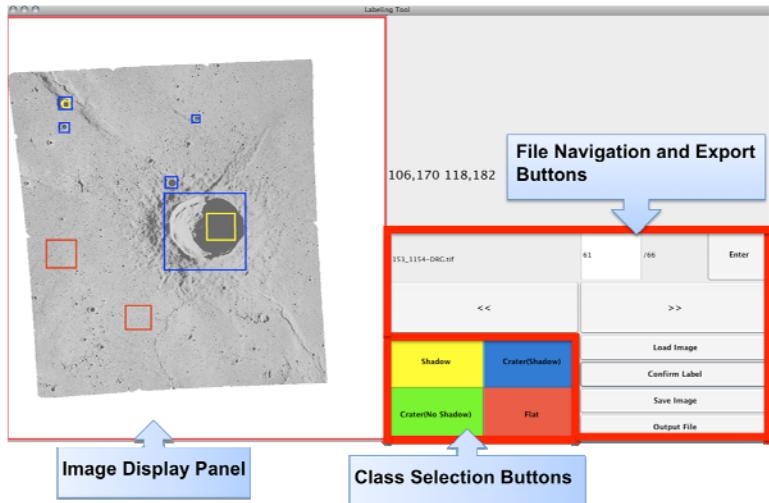
## 4 Lunar Image Labeling Tool

Training image classifiers requires large amount of training data. Since there is no labeled lunar image database available, we decided to develop the Lunar Image Labeling Toolkit (LILT) that allow us to annotate terrain types and facilitate future research in the related field. The LILT toolkit is open-source and is available on our website<sup>1</sup>.

The graphical user interface for terrain type labeling is developed in Java Swing. Figure 4 shows the user interface that consists of a file navigation section, a class labeling section, and a display panel. A user can intuitively choose a class, drag the desired zone, or undo operations. We use the LILT toolkit to label 931 lunar image patches of interest from 66 raw images. The toolkit supports various input image formats such as PNG, TIFF, or JPEG, and outputs a ground-truth file containing file name, patch coordinates, and the terrain type of the image patch.

---

<sup>1</sup> <http://www.ece.cmu.edu/~hengtze/projects/lunar/LILT.zip>



**Fig. 4.** User interface of the Lunar Image Labeling Toolkit

## 5 Experimental Results

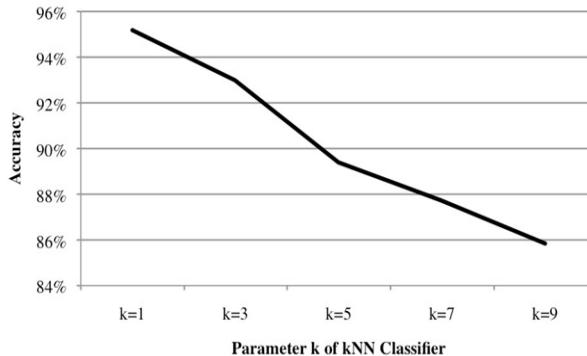
We evaluate the system on a dataset that consists of 931 lunar image patches from NASA Apollo missions.

**Table 1.** Accuracy with different feature and classifier combinations

Method	Decision Tree	k-Nearest Neighbor
Luminance Histogram	59.47%	56.76%
Haar-like Feature	64.45%	68.47%
Gradient Orientation	85.73%	95.56%

### 5.1 Evaluation on Feature and Classifier Combinations

First, we evaluate the performance using different combinations of image features and classification algorithms to find which ones are more effective. As shown in Table 1, luminance histogram does not work well in general because the luminance distribution is not necessarily consistent across images of the same terrain type. One terrain type can be dark when the sunlight is blocked and bright otherwise. Haar-like features roughly capture the edges or texture changes and slightly improve the accuracy to around 65%, but fail to recognize terrains with highly varying sizes since the set of masks is fixed. Among the three features we have tested, histogram of gradient orientation yields the best performance, which is 85.73% using the decision tree, and 95.56% using the *k*-nearest neighbor classifier. The results show that the histogram of gradient orientation effectively learns the signature of shapes and edge directions of different terrain types. Furthermore, since only the distribution of gradient orientation (rather than magnitude) is computed, the feature is more robust and scale-invariant.

**Fig. 5.** Accuracy with different  $k$  in  $k$ -NN classifier

As for the comparison of classification algorithms, from Table I we can see that  $k$ -NN generally performs better than decision trees. When using histogram of gradient orientation as features,  $k$ -NN improves the overall accuracy by approximately 10%. Therefore, we choose gradient orientation histogram and  $k$ -NN as our proposed method for subsequent experiments.

**Table 2.** Classification accuracy using different percentage of training data

Percentage of Training Data	10%	20%	30%	50%	90%
Accuracy	93.58	94.26	94.81	95.03	95.51

## 5.2 Cross Validation

We further conduct several experiments to test the best-performing system, which consists of gradient orientation histogram as feature and  $k$ -NN classifier, under different parameter settings. As shown in Figure 5, the  $k$ -NN classifier performs best when  $k=1$ , and the accuracy decreases as  $k$  increases. One reason is that some craters without shadows are of subtle edges and are more likely to be misclassified as flat areas; therefore, increasing  $k$  brings more potentially misleading patch candidates. As a result, we empirically set  $k$  to 1 for subsequent evaluations. Then, we test the performance under the condition that only a small amount of training data is available. As shown in Table II, even using only 10% of training data (approximately 90 image patches), we still achieve 93.58% of accuracy. This shows the robustness of the proposed method. We also performed 10-fold and leave-one-out cross validation with the results shown in Table III. The overall accuracy is 93.67% and 95.7%, respectively.

**Table 3.** Classification accuracy measured by cross-validation

	10-fold Cross Validation	Leave-one-out Cross Validation
Accuracy	93.67	95.51

## 6 Conclusions and Future Work

In this paper, we have investigated image feature and classifiers for lunar terrain classification. Evaluated on a large dataset of 931 lunar image patches from Apollo missions, the proposed system using histogram of gradient orientation and  $k$ -NN classifier achieved a high accuracy of 95.56%. We have also shown that the proposed approach outperforms Haar-like features and luminance histogram by around 30%. The system, along with the open-source Lunar Image Labeling Toolkit, can be applied to larger orbital image mining package for imagery from NASA space missions such as LCROSS and HiRISE.

For future work, we plan to focus on automatic terrain patch detection using scale-invariant image descriptors. Furthermore, we plan to extend the system architecture for recognizing a wider category of terrain types on the moon and other planets. We will also test the performance using multiple feature fusion and other generative or discriminative classifiers.

## References

1. Datta, R., Joshi, D., Li, J., Wang, J.Z.: Image Retrieval: Ideas, Influences, and Trends of the New Age. *ACM Computing Surveys* 40(2), 1–60 (2008)
2. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 181–184 (2005)
3. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, pp. 511–518 (2001)
4. Duygulu, P., Barnard, K., Freitas, N.d., Duygulu, P., Barnard, K., Forsyth, D.: Object Recognition as Machine Translation: Learning a Lexicon for a Fixed Image Vocabulary. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) *ECCV 2002. LNCS*, vol. 2353, pp. 349–354. Springer, Heidelberg (2002)
5. Mohan, A., Papageorgiou, C., Poggio, T.: Example-based object detection in images by components. *IEEE Trans. Pattern Analysis and Machine Intelligence* 23(4), 349–361 (2001)
6. Burl, M.C., Merline, W.J., Bierhaus, E.B., Colwell, W., Chapman, C.R.: Automated Detection of Craters and Other Geological Features. In: Proc. International Symp. Artificial Intelligence Robotics and Automation in Space (2001)
7. Stepinski, T.F., Ghosh, S., Vilalta, R.: Automatic Recognition of Landforms on Mars Using Terrain Segmentation and Classification. In: Todorovski, L., Lavrač, N., Jantke, K.P. (eds.) *DS 2006. LNCS (LNAI)*, vol. 4265, pp. 255–266. Springer, Heidelberg (2006)
8. Martins, R., Pina, P., Marques, J., Silveira, M.: Crater Detection by a Boosting Approach. *IEEE Geoscience and Remote Sensing Letters* 6(1), 127–131 (2009)

# Surface Modeling of the Corpus Callosum from MRI Scans

Ahmed Farag<sup>1</sup>, Shireen Elhabian<sup>1</sup>, Mostafa Abdelrahman<sup>1</sup>, James Graham<sup>1</sup>,  
Aly Farag<sup>1</sup>, Dongqing Chen<sup>1</sup>, and Manuel F. Casanova<sup>2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering

<sup>2</sup> Department of Psychiatry

University of Louisville

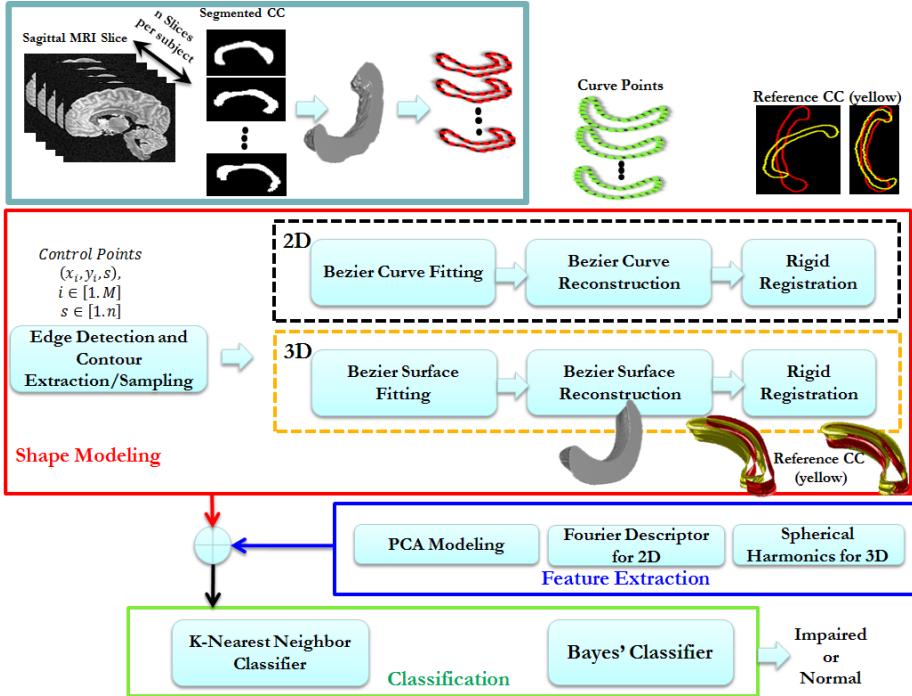
aafara04@louisville.edu

www.cvip.uofl.edu

**Abstract.** In this paper, the Bezier curve and surface are used to model the shape of the Corpus Callosum (CC) region from T1-weighted clinical MRI scans. We derive a closed form solution for the Bezier coefficients in 2D and 3D Euclidean spaces. The coefficients of the models are used for reconstruction of the CC contours and surfaces with varying degrees of accuracy, and constitute basis for discrimination between populations, and ways to enhance elastic registration of the CC. The discrimination ability of the Bezier curves and surfaces are evaluated against the Fourier Descriptors (FD) and Spherical Harmonics (SH) approaches.

## 1 Introduction

The corpus callosum has been used for various anatomical and behavioural studies of the brain for over three decades (e.g., [1]). The reader may refer to various research centers worldwide that deal with neuroimaging for examples of the vast literature that involves this structure. We will not list such literature here due to space limitations. The ongoing research in our group aims to model the corpus callosum and develop robust discriminatory measures that can distinguish the brains of normal and impaired subjects from neuroimaging. Fig. 1 shows a block diagram of the overall analysis system. A segmented stack of corpus callosum slices at the sagittal plane are fed into an edge detector (e.g., the Canny operator or an Active Contour Algorithm) in order to extract the CC contours, and a set of control points are obtained by contour sampling [2][3]. Contour and surface modeling approaches are deployed to represent the CC; including the Bezier Curves [4][5], Fourier Descriptors[6][7] and the Spherical Harmonics [8][9]. Rigid registration based on generalized Procrustes analysis [10] is employed to filter out similarity transformations (scaling, rotation and translation) from the CC contours and surfaces. In addition, AAM and ASM models [e.g., [11][12]] have been studied. Various classification approaches, such as: K-nearest neighbor (KNN) and Bayesian classifiers, are employed to hold the greatest discriminatory abilities between the impaired and normal subjects [13]. In this paper we focus on the modeling issue, i.e. the red



**Fig. 1.** A Block diagram of the analysis system of the CC from T1-weighted MRI scans. From segmented CC, contours and surface of the CC structure are modeled using various parametric methods, features are extracted and optimized to suite various classification approaches to distinguish between normal and abnormal CC.

block. We will also provide preliminary comparison of the Bezier surface and contours to the Fourier Descriptors and Spherical Harmonics (highlighted in the bottom section of Fig. 1).

In our earlier work [3], we dealt with the 2D part of the shape modeling problem where only one slice is given per subject, as extension to this work we present a mathematical formulation of 2D and 3D CC shape modeling using Bezier theory. This study can be further extended to study the features derived from these shape models which lend benefit for discrimination between populations such as normal and autistic populations from a clinical MRI study. The paper will deal very briefly with classification of normal and autistic brains using the features extracted from the Bezier curves and surfaces.

## 2 Modeling the Corpus Callosum

In order to perform classification of the CC using machine learning, a proper model is needed, and a set of features that can capture the essence of the structure is required. The focus of this research is on approaches that are robust in terms of ability to discriminate normal and abnormal CC, and the execution time. Among the models tried are 3D mesh models which enable quantitative deformation analysis, and contour

based models that are 2D outlines of the CC. A 3D mesh is more descriptive in terms of a holistic representation of the CC and enables techniques, such the finite elements, to be used to quantify deformations of the structures. Likewise, a 3D mesh may be used in the context of elastic registration to describe the deformation (or differences) in CC in abnormal vs. normal ones. In addition, a 3D mesh may be the basis for complete modeling using active shape/active appearance (ASM/AAM) models of the CC based on an annotated population (e.g., [11] [12]). Likewise, approaches such spherical harmonics may lend credence to the discrimination between normal and abnormal structures. All these ideas have been tried by our group and are part of the drive towards creating a robust and effective approach for automatic diagnosis based on neuroimaging. These approaches, thus far, have shown to be computationally extensive and did not deliver the accuracy rates that are expected. Indeed, 2D analysis based on contours of the CC has shown to hold promise and provide competitive accuracy to extensive approaches using 3D mesh analysis [2][7].

## 2.1 Corpus Callosum Contours

The Corpus Callosum can be represented as a parametric curve which can be defined as follows; Let  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M\}$  be a set of points in the 2-dimensional Euclidean space, a corpus Callosum curve can be defined in terms of these points as:

$$\mathcal{C}: \mathbf{x}(t) = f_0(t)\mathbf{x}_0 + f_1(t)\mathbf{x}_1 + \dots + f_M(t)\mathbf{x}_M = \sum_{k=0}^M f_k(t)\mathbf{x}_k \quad (1)$$

Where  $f_k(t)$  are continuous functions defined on the interval  $t \in [0,1]$ . The points  $\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M$  are called control point and  $f_k(t)$  are called the basis functions of the curve  $\mathcal{C}$ . Bézier curves are one of the most popular representations for curves which satisfy convex hull property, it can be defined as follows;

$$\mathbf{x}(t) = \sum_{k=0}^M B_k^M(t)\mathbf{x}_k \quad t \in [0,1] \quad (2)$$

where the basis functions  $B_k^M(t)$  are the Bernstein polynomials defined by;

$$B_k^M(t) = \binom{M}{k} t^k (1-t)^{M-k} \quad (3)$$

Equations (2) and (3) can be combined to express the Bezier curve a polynomial (explicit function in the parameter  $t$ ) instead of a sum of Bernstein polynomials. Binomial theorem can be applied to the definition of the curve followed by rearrangement to yield,

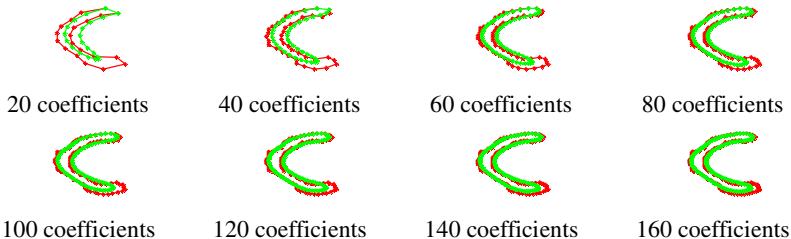
$$\mathbf{x}(t) = \sum_{j=0}^M c_j t^j \quad t \in [0,1] \quad (4)$$

wherethe coefficients for the expansion of Bernstein polynomials into powers of  $t$  are given as follows, (see [2][3] for proof).

$$c_j = \frac{M!}{(M-j)!} \sum_{i=0}^j \frac{(-1)^i \mathbf{x}_{j-i}}{i! (j-i)!} \quad (5)$$

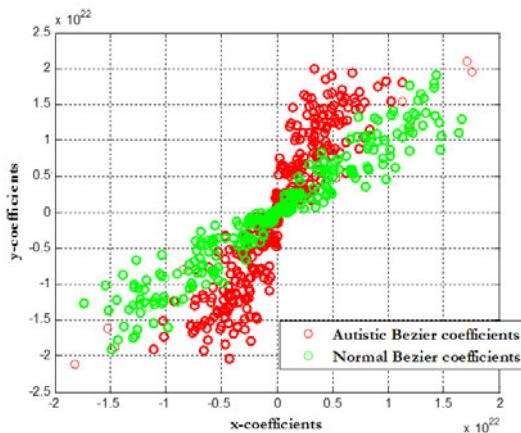
The form in Eq. 5 lends a great benefit for parameterization of the cc contours, and can be the basis for various analysis in the  $t$ -domain and other mappings, including the frequency domain.

Fig. 2 shows fitting of corpus callosum for different number of coefficients. We note that the reconstructed curves maintain the main features of the original data. In our case we wanted the Bezier curves to capture the general shape, perimeter or circumference and volume of the original cc contours.



**Fig. 2.** Bezier curve reconstruction of the corpus callosum using different number of coefficients, the original curve is shown in red while the reconstructed one is shown in green

Fig. 3 shows the scatter plot of the Bezier curve coefficients belonging to autistic and normal subjects. This figure gives an impression of the discrimination power of Bezier coefficients when classification is of concern (more results are in [2][3]).



**Fig. 3.** A scatter plot of the Bezier curve coefficients of autistic and normal subjects

## 2.2 Corpus Callosum Surfaces

In ways of generalization of the Bezier contours models, a Corpus Callosum in 3D can be represented as a parametric surface. Let  $\{\mathbf{x}_1, \mathbf{x}_2, \dots, \mathbf{x}_M\}$  be a set of points in the 3-dimensional Euclidean space, a corpus Callosum surface can be defined in terms of these points as:

$$\text{S: } \mathbf{x}(u, v) = (u, v) = \sum_{i=0}^n \sum_{j=0}^m f_i^n(u) f_j^m(v) \mathbf{x}_{ij}$$

where  $f_k(u, v)$  are continuous functions defined on the interval  $u, v \in [0,1]$ . The points  $\mathbf{x}_{ij}$ 's are called control point defined on a spatial support and  $f_k(u, v)$  are called the basis functions of the surface. Bézier surfaces are one of the most popular representations for surfaces which satisfy convex hull property, it can be defined as follows;

$$\mathbf{x}(u, v) = \sum_{i=0}^n \sum_{j=0}^m B_i^n(u) B_j^m(v) \mathbf{x}_{ij} \quad u, v \in [0,1] \quad (6)$$

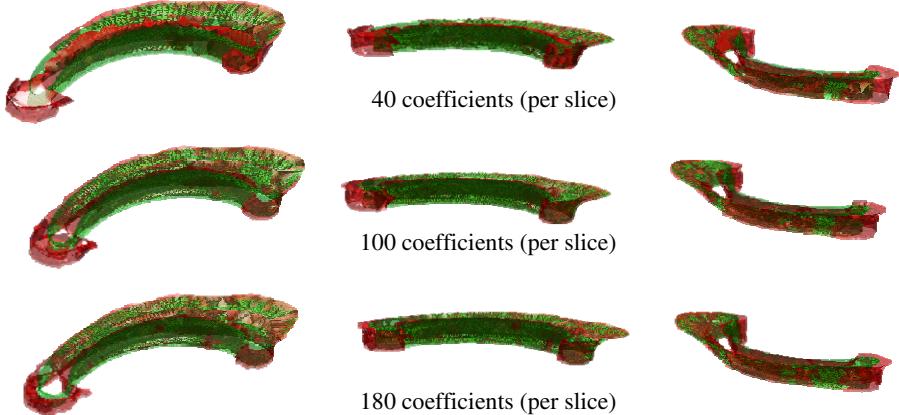
where the basis functions  $B_k^M(t)$  are the Bernstein polynomials defined by;

$$B_k^M(t) = \binom{M}{k} t^k (1-t)^{M-k} \quad (7)$$

where the coefficients for the expansion of Bernstein polynomials into powers of  $u$  and  $v$  are given as follows, (see Appendix for proof)

$$c_{\alpha\beta} = \frac{n!}{(n-\alpha)!} \frac{m!}{(m-\beta)!} \left( \sum_{k=0}^{\alpha} \sum_{l=0}^{\beta} \frac{(-1)^k (-1)^l}{k! l! (\alpha-k)! (\beta-l)!} \mathbf{x}_{(\alpha-k)(\beta-l)} \right) \quad (8)$$

Figure 4 shows a sample of CC surfaces along with their reconstructions using different number of Bezier coefficients.



**Fig. 4.** Three different view of Bezier surface reconstruction of the corpus callosum using different number of coefficients, i.e. control points per slice, the original surface is shown in red while the reconstructed one is shown in green

### 3 Surface Parameterization in the Frequency Domain

In this section, we briefly introduce the mathematical fundamental of Fourier descriptor [6][7][3], and spherical harmonics (SPHARM) [8][9]. The real-value SPHARM is adopted in this paper.

### 3.1 Fourier Descriptor

Fourier descriptor represents the shape in terms of its spatial frequency content. The boundary/contour of a corpus callosum, which can be the control points or the reconstructed points for Bezier curve fitting, is represented as a periodic function which can be expanded using a Fourier series, hence we obtain a set of coefficients which capture the shape information. Let the corpus callosum be represented as a parametric curve  $\mathcal{C} = \{x(t), y(t)\}$  such that  $t \in [0,1]$ , by considering a complex image plane, we will end up with a one dimensional function  $u(t) = x(t) + jy(t) = u(t + nT)$  for  $n = 1, 2, \dots$  from which the Fourier series expansion is obtained. In this work, the Fourier descriptor of the mean autistic  $\overline{\mathcal{C}_A}$  and the mean normal  $\overline{\mathcal{C}_N}$  colosums are obtained used for classification. See [2][3][7] for examples of using the FD for classification of normal and autistic brains.

### 3.2 Surface Parameterization

Let  $M$  and  $S^2$  be a 3D corpus callosum surface and a unit sphere, respectively.  $M$  and  $S^2$  are generated by creating polygonal meshes. The surface parameterization could be done in the following way. A point  $U = (u, v, w) \in S^2$  is mapped to  $X = (x, y, z) \in M$  using a mapping  $U$ . The mapping is realized by a deformable surface algorithm which preserves the topological connectivity of the meshes. Then,  $U$  is parameterized by the spherical coordinates:  $(u, v, w) = (\sin \theta \cos \varphi, \sin \theta \sin \varphi, \cos \theta)$ , where,  $(\theta, \varphi) \in N = [0, \pi] \otimes [0, 2\pi]$ .  $\theta$  is the polar angle, and  $\varphi$  is the azimuthal angle.

### 3.3 Spherical Harmonic Representation

In mathematics, the spherical harmonics are the angular portion of a set of solutions to Laplace's equation in a system of spherical coordinates. For a given  $f(x, y, z)$ , the Laplacian equation  $\nabla^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2} + \frac{\partial^2 f}{\partial z^2} = 0$ , then under the spherical coordinate, the Laplacian equation could be written as following.

$$\nabla^2 f = \frac{1}{r^2} \frac{\partial}{\partial r} \left( r^2 \frac{\partial f}{\partial r} \right) + \frac{1}{r^2 \sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial f}{\partial \theta} \right) + \frac{1}{r^2 \sin^2 \theta} \frac{\partial^2 f}{\partial \varphi^2} = 0$$

On a unit sphere ( $r = 1$ ), the above equation could be simplified as:

$$\nabla^2 f = \frac{1}{\sin \theta} \frac{\partial}{\partial \theta} \left( \sin \theta \frac{\partial f}{\partial \theta} \right) + \frac{1}{\sin^2 \theta} \frac{\partial^2 f}{\partial \varphi^2} = 0;$$

There are  $2l + 1$  eigen functions  $Y_{lm}(|m| \leq l)$ .  $Y_{lm}$  is the spherical harmonic of degree  $l$  and order  $m$ , and it is defined as:

$$Y_l^m = \begin{cases} \sqrt{2} N_l^m \cos(m\varphi) P_l^m \cos(\theta) & m > 0 \\ N_l^0 P_l^0 \cos(\theta) & m = 0 \\ \sqrt{2} N_l^{|m|} \sin(|m|\varphi) P_l^{|m|} \cos(\theta) & m < 0 \end{cases}$$

where  $N_l^{|m|} = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}}$   $P_l^{|m|}$   $\cos(\theta)$  is a normalization coefficient (see [8][9]).

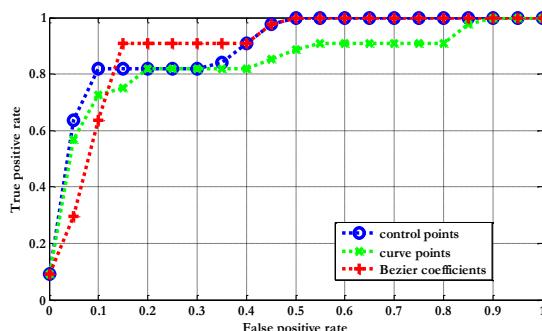
## 4 Applications

In this section we will use the classification as an application of the CC modeling which we propose. The classification task involves measuring the degree of similarity

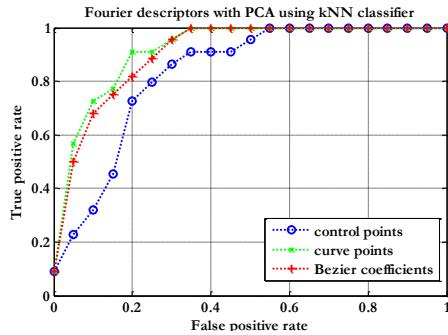
of a test or unknown corpus callosum and trained models of autistic and normal subjects; these models can be: In 2D (1) Fourier descriptors of the mean shapes, (2) Fourier descriptors of the contour points (control or reconstructed points), (3) eigen-configurations of the contour points (control or reconstructed points) or (4) eigen-configurations of the Bezier coefficients. In 3D (1) Spherical harmonics of the mean shapes, (2) Spherical harmonics of the surface points (control or reconstructed points), (3) eigen-configurations of the surface points (control or reconstructed points) or (4) eigen-configurations of the Bezier surface coefficients.

We should point out that the circumference of the corpus callosum of the two populations was quite distinct. In fact, for the set of data we used in this paper, 100% classification rate was obtained based on using the circumferences (on all data). In practice, we may not always have non-intersecting contours of the corpus callosum and a single feature classification based on the circumference alone cannot be expected to be robust. Hence, the feature vector may include the circumference together with the Bezier coefficients and the Fourier descriptors. Distance classification (using the L2 norm) was used, however other classifiers such as SVM and AdaBoost may be used as well [13].

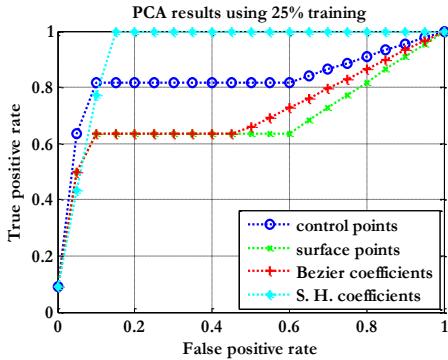
In this paper, the Receiver Operating Characteristic (ROC) has been used to visualize organizing and selecting classifiers based on their performance. The experimental results of the classifiers for different features have been tested on training and testing data from the 16 autistic and 22 normal contours. It can be observed that Bezier coefficients modeled by principle component analysis (PCA) outperform the CC control points. It can also be shown that the reconstructed curve has lost discriminatory information when compared to Bezier coefficients as shown in Fig. 5. From Fig. 6 it is clear that the Fourier descriptor of the Bezier coefficients and the reconstructed curve points modeled by principle component analysis (PCA) outperform the Fourier descriptor CC control points. In case of 3D from Fig. 7 it is clear that the spherical harmonics coefficients modeled by principle component analysis (PCA) out perform all other features, and that Bezier surface coefficient is comparable to the CC surface control points. It can also be shown that the reconstructed surface has lost discriminatory information when compared to Bezier surface coefficients.



**Fig. 5.** ROC curves for classification using control points of CC contour, reconstructed Bezier curve points and Bezier coefficients



**Fig. 6.** ROC curves for classification using Fourier descriptor of control points of CC, reconstructed Bezier curve points, and Bezier curve coefficients



**Fig. 7.** ROC curves for classification using control points of CC surface, reconstructed Bezier surface points , Bezier surface coefficients, and the spherical harmonics (S.H.) coefficients

## 5 Conclusion

This paper describes a generalized approach to model the CC using Bezier contour and surface approaches. A closed form solution for the parametric representation of the Bezier surface is derived, along the lines of the contour coefficients. These models may be used for reconstruction purposes and for discrimination between populations. In 2D we studied the Bezier curves and Fourier Descriptors, and in 3D we studied the Bezier surfaces and Spherical Harmonics. A generalized approach for analysis of the CC (as depicted in Fig. 1) form neuroimaging is now in place and used for ongoing studies to discriminate between normal and impaired subjects in which structural analysis hold great promise, such as the studies of traumatic brain injury and autism. We will report on progress of these studies in the future.

## References

1. Clarke, J.M., Zaidel, E.: Anatomical-behavioral relationships: Corpus callosum morphometry and hemispheric specialization. *Behavioural Brain Research* 64(1-2), 185–202 (1994)
2. Farag, A.: Shape Analysis of the Corpus Callosum of Autism and Normal Subject in Neuro Imaging, Master of Engineering, University of Louisville (August 2009)
3. Farag, A., Elhabian, S., Abdelrahman, M., Graham, J., Farag, A., Chen, D., Casanova, M.F.: Shape Modeling of the Corpus Callosum. In: IEEE Engineering in Medicine and Biology, EMBC 2010, Buenos Aires, Argentina, August 31-September 4 (2010)
4. O'Rourke, J.: Computational Geometry in C, 2nd edn. Cambridge University Press, Cambridge (September 1998)
5. Hardy, A., Steeb, W.: Mathematical tools in computer graphics with C<sup>++</sup> implementations. World Scientific, Singapore (May 2008)
6. Zahn, C.T., Roskies, R.Z.: Fourier descriptors for plane close curves. *IEEE Trans. Computers* C-21, 269–281 (1972)
7. Abdelmunim, H., Farag, A., Casanova, M.F.: Frequency-Domain Analysis of the Human Brain for Studies of Autism. In: Proc., 7th IEEE International Symposium on Signal Processing and Information Technology, ISSPIT 2007, Cairo, Egypt, December 15-18, pp. 1198–1203 (2007)
8. Chung, M.K., Wang, S., Dalton, K.M., Davidson, R.J., Robbins, S., Evans, A.C.: Tensor-based cortical morphometry via weighted spherical harmonic representation. *IEEE Transactions On Medical Imaging* 27(8) (August 2008)
9. Khairy, K., Howard, J.: Spherical harmonics-based parametric deconvolution of 3D surface images using bending energy minimization. *Journal of Medical Image Analysis* 12, 217–227 (2008)
10. Goodall, C.: Procrustes methods in the statistical analysis of shape. *Journal of the Royal Statistical Society B* 53(2), 285–339 (1991)
11. Edwards, G.J., Cootes, T.F., Taylor, C.J.: Advances in Active Appearance Models. In: Proc. International Conference on Computer Vision, ICCV 1999, pp. 137–142 (1999)
12. Matthews, I., Baker, S.: Active Appearance Models Revisited. *International Journal of Computer Vision*, 135–164 (2004)
13. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification, 2nd edn. Wiley, Chichester (2001)

## Appendix: Bezier Surface Coefficients

Bezier surface can be defined in terms of Bernstein polynomials as follows;

$$\begin{aligned} x(u, v) &= \sum_{i=0}^n \sum_{j=0}^m B_i^n(u) B_j^m(v) x_{ij} \\ &= \sum_{i=0}^n \sum_{j=0}^m \binom{n}{i} u^i (1-u)^{n-i} \binom{m}{j} v^j (1-v)^{m-j} x_{ij} \end{aligned}$$

According to the Binomial theorem, it is possible to expand any power of  $x + y$  into a sum of the form,  $(x + y)^n = \sum_{i=0}^n \binom{n}{i} x^{n-i} y^i$ . Hence we can expand  $(1 - u)^{n-i}$  as follows;

$$(1 - u)^{n-i} = (1 + (-u))^{n-i} = \sum_{k=0}^{n-i} \binom{n-i}{k} (1)^{n-i-k} (-u)^k = \sum_{k=0}^{n-i} \binom{n-i}{k} (-1)^k u^k$$

therefore,

$$u^i(1-u)^{n-i} = u^i \sum_{k=0}^{n-i} \binom{n-i}{k} (-1)^k u^k = \sum_{k=0}^{n-i} \binom{n-i}{k} (-1)^k u^{k+i}$$

Similarly

$$v^j(1-v)^{m-j} = v^j \sum_{l=0}^{m-j} \binom{m-j}{l} (-1)^l v^{l+j}$$

Thus

$$x(u, v) = \sum_{i=0}^n \sum_{j=0}^m \binom{n}{i} \left( \sum_{k=0}^{n-i} \binom{n-i}{k} (-1)^k u^{k+i} \right) \binom{m}{j} \left( \sum_{l=0}^{m-j} \binom{m-j}{l} (-1)^l v^{l+j} \right) x_{ij}$$

Where

$$\binom{n}{i} \binom{n-i}{k} = \frac{n!}{i!(n-i)!} \frac{(n-i)!}{i!(n-i-k)!} = \frac{n!}{i! k! (n-(i+k))!}$$

Let  $\alpha = i + k \rightarrow i = \alpha - k$ , we have:

$$\binom{n}{i} \binom{n-i}{k} = \frac{n!}{(\alpha-k)!k!(n-\alpha)!} \quad \text{and} \quad \binom{m}{j} \binom{m-j}{l} = \frac{m!}{(\beta-l)!l!(m-\beta)!}$$

Thus,

$$\begin{aligned} x(u, v) &= \sum_{i=0}^n \sum_{j=0}^m \left( \sum_{k=0}^{n-i} \binom{n}{i} \binom{n-i}{k} (-1)^k u^{k+i} \right) \left( \sum_{l=0}^{m-j} \binom{m}{j} \binom{m-j}{l} (-1)^l v^{l+j} \right) x_{ij} \\ &= \sum_{\alpha=0}^n \sum_{\beta=0}^m \left( \sum_{k=0}^{n-\alpha+k} \frac{n!}{(\alpha-k)!k!(n-\alpha)!} (-1)^k u^{k+\alpha-k} \right) \left( \sum_{l=0}^{m-\beta+l} \frac{m!}{(\beta-l)!l!(m-\beta)!} (-1)^l v^{l+\beta-l} \right) x_{(\alpha-k)(\beta-l)} \end{aligned}$$

from the factorial rules, to evaluate  $(\alpha - k)!$  we need to have  $\alpha \geq k$ , hence the third summation should be upper bounded by  $\alpha$ , same case for the fourth summation.

$$\begin{aligned} x(u, v) &= \sum_{\alpha=0}^n \sum_{\beta=0}^m \left( \sum_{k=0}^{\alpha} \frac{n!}{(\alpha-k)!k!(n-\alpha)!} (-1)^k u^\alpha \right) \left( \sum_{l=0}^{\beta} \frac{m!}{(\beta-l)!l!(m-\beta)!} (-1)^l v^\beta \right) x_{(\alpha-k)(\beta-l)} \\ &= \sum_{\alpha=0}^n \sum_{\beta=0}^m \frac{n!}{(n-\alpha)!(m-\beta)!} \left( \sum_{k=0}^{\alpha} \sum_{l=0}^{\beta} \frac{(-1)^k (-1)^l}{k! l! (\alpha-k)! (\beta-l)!} x_{(\alpha-k)(\beta-l)} \right) u^\alpha v^\beta \end{aligned}$$

Thus the Bezier surface coefficients can be given by;

$$c_{\alpha\beta} = \frac{n!}{(n-\alpha)!} \frac{m!}{(m-\beta)!} \left( \sum_{k=0}^{\alpha} \sum_{l=0}^{\beta} \frac{(-1)^k (-1)^l}{k! l! (\alpha-k)! (\beta-l)!} x_{(\alpha-k)(\beta-l)} \right)$$

■

# Track Detection for Autonomous Trains

Michael Gschwandtner<sup>1</sup>, Wolfgang Pree<sup>2</sup>, and Andreas Uhl<sup>1</sup>

<sup>1</sup> Multimedia Signal Processing and Security Lab (WaveLab)

Department of Computer Sciences

University of Salzburg, Austria

{mgschwan, uhl}@cosy.sbg.ac.at

<sup>2</sup> C. Doppler Laboratory Embedded Software Systems

University of Salzburg, Austria

wolfgang.pree@cs.uni-salzburg.at

**Abstract.** This paper presents a way to efficiently use lane detection techniques - known from driver assistance systems - to assist in obstacle detection for autonomous trains. On the one hand, there are several properties that can be exploited to improve conventional lane detection algorithms when used for railway applications. The heavily changing visual appearance of the tracks is compensated by very effective geometric constraints. On the other hand there are additional challenges that are less problematic in classical lane detection applications. This work is part of a sensor system for an autonomous train application that aims at creating an environmentally friendly public transportation system.

**Keywords:** lane detection, autonomous vehicle, machine vision, train.

## 1 Introduction

Autonomous transportation systems like ULTra<sup>1</sup> (Urban Light Transport) are trying to create an economically efficient and environmentally friendly transportation system. Most of the currently available (or under construction) Personal Rapid Transit (PRT) systems are designed to be operated on special guide-ways (i.e. railways) that are sometimes even elevated. Thus the implementation of such a system is bound to a huge amount of initial costs. The ULTra system states constructions costs per mile between £5M and £10M. Another solution would be to operate autonomous cars on the streets along with the normal traffic. This however vastly increases the complexity of the scenarios that have to be handled by the autonomous cars. Furthermore, it is a substantial threat to the safety of passengers and outside traffic participants (especially children). The operation of autonomous vehicles on separate guide-ways seems to be a reasonable but expensive solution, because the path of the PRT vehicle does not conflict with the normal traffic and the track is predetermined which basically reduces the control commands to *stop* or *go*.

---

<sup>1</sup> <http://www.ultraprt.com>

A cost effective solution that interferes only marginally with normal traffic seems to be somewhere between the PRT and autonomous cars. We need a system that can drive along a fixed path but does not require the building costs for a whole dedicated guide-way/track. A possible solution to this is the use of existing railroad lines (smaller branch lines) that are then operated autonomously with smaller passenger trains (so called *train-lets* [3]).

Such a system obviously consists of many parts ranging from high-level train-let control to low-level sensor data acquisition. In this paper, we are going to look at how lane detection known from autonomous cars can be used to aid obstacle detection and also to provide basic region of interest information for other sensors.

We present an algorithm that uses simple, yet efficient geometric constraints derived from the unique properties of *railroad tracks* to allow fast and robust track detection. The obtained position information can then be used for obstacle detection and sensor-fusion. In Section 2 we look at previous work that is related to lane detection. Section 3 compares the properties of streets with railroad tracks. The basic idea for lane detection adapted to autonomous trains is explained in Section 4. In Section 5 we present several examples of working and problematic scenarios. We conclude with an outlook how this system can be potentially improved in Section 6.

## 2 Previous Work on Lane/Track Detection

Lane detection for driver assistance systems is a topic that gained a lot of attention during the last ten years ([2][4][5][6][7][11][12]). Some approaches work on the acquired images directly which represent a perspective projection of the scene ([2][11][1]) and some perform a conversion of the scene into a top down view called Inverse Perspective Mapping ([4][5][7][9]). Most systems use a simple lane model to describe the lane which also applies to railroads. However only few systems are designed specifically for railroad detection ([8][10]). An obstacle detection system for trains is proposed in [10]. In [8] a vision based system for collision avoidance of rail track maintenance vehicles is proposed. It is based on detecting railroad tracks by applying techniques similar to lane detection for driver assistance systems. The spatial period of the sleepers and the distance between the rails are used to calibrate the camera parameters. A piecewise quadratic function is then fitted to candidate rail-pixels in the acquired (perspective) image and compared to the previous frame. However to the best of our knowledge no fully functional vision based obstacle detection system for railways exists to date. This work uses the well researched field of lane detection and tracking and extends it to the field of train applications.

## 3 Comparison

While the basic task of lane detection in street and railway scenarios is the same, there are several properties that require special attention and may help us to

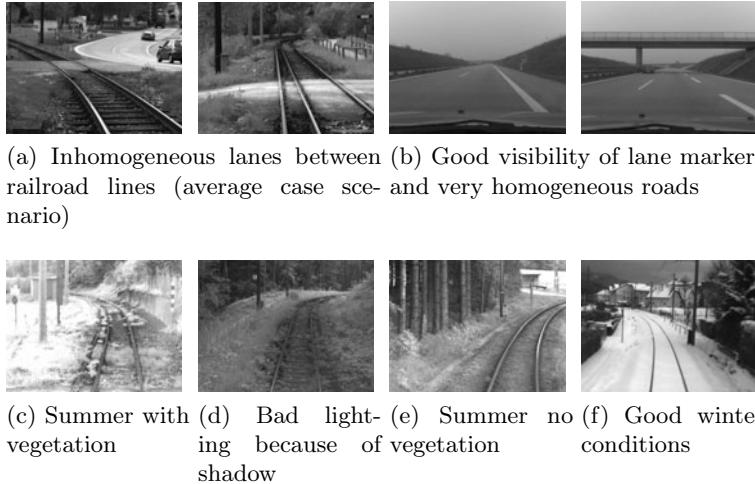
**Table 1.** Properties of street lanes and *railway tracks*

Street	Railway
variable lane width	fixed lane width
variable lateral offset	zero lateral offset
varying type of lane markings	fixed “lane markings”
general lane appearance is relatively homogeneous	several different (inhomogeneous) “lanes”
lane markings are designed for optimal visibility	visibility is not guaranteed
lane markings have no volume and thus don’t cast shadows on the ground	tracks have a certain height and thus cast shadow on the ground
construction sites and obstacles can easily change the path of a car within a road	vehicle path through the world coordinate system is fixed
the speed of a car is generally adapted to weather and visibility conditions	automatic trains are operated at nearly constant speed independent of most weather conditions
the horizontal movement of a car-mounted camera is low due to the low height of the vehicle	swinging of the train cause substantial change of the camera position along the path

improve the robustness of automatic train systems. Table 1 lists some important differences, which are discussed subsequently.

First of all, the lane width on railways is obviously fixed along the whole path. Otherwise the train would not be able to drive the complete track. The width of lane markings on streets, however, depends primarily on the type of road and is generally limited by a minimum width required by law, which in turn depends on the country the street belongs to. The lateral offset of a car relative to the center of the lane is variable, which is especially true for wider lanes. We will later see that a fixed lateral offset can be exploited to reduce the possible track candidates.

Road lane markings are designed in a way that should be optimally visible to a human observer (Figure 1b), and they are continuously improved. In contrast, the only purpose of railroad tracks is to guide the train. It is just a side effect if the tracks are easily visible to an observer. Although in many situations the tracks are very prominent, in general visibility is affected by changes in lighting and weather much stronger than road lane markings. An advantage of rails over lane markings is that they are constantly grinded each time a train rolls over them. This keeps the top of the rails from corroding. The appearance of the street itself is also very homogeneous (Figure 1b) compared to the track bed of railways (Figure 1a). The track bed, or in general the space between the rails, consists for example of gravel, asphalt, snow or even grass (Figure 1a). The last significant difference in the visual appearance is the volume of the tracks. Lane



**Fig. 1.** Comparison between road lanes and *railway tracks*, and overview of different scenarios

markings are flat and basically have no volume or height. This means that they can not cast shadows on the ground and thus the detection of the lane marking itself is invariant to the position of the sun, if no other object casts a shadow on the street. Tracks however have a certain height, i.e. several centimeters, and thus cast shadows which create additional edges in the image and *weaken* the visual appearance of the real edge.

If we combine the fact that the lateral offset is fixed and that the position of the track can not be changed easily, it is clear that the train always moves on a fixed path with respect to the world coordinate system. This allows a much tighter integration of a-priori information like the prediction of the vehicle position at upcoming points in time. However this is only true for the whole vehicle, since trains have a strong trend to swing left/right especially at higher speeds. This is probably due to the great mass in combination with the suspension that is designed to make the ride comfortable for the passengers. This strong swinging combined with the fact that the train is considerably higher than a regular car results in a displacement of the camera system that can not be predicted easily. This means that even if two frames are acquired at the exact same position of the vehicle at two different times the position and orientation of the camera with respect to the world coordinate system is not the same.

A final but very significant property are the weather conditions. Trains are operated at constant speed over a much greater range of weather conditions than a normal car. For example, even if the track is nearly fully covered with snow the trains are still operated with no or only a slight reduction in speed, because they need to keep their timetable.

## 4 Track Detection

### 4.1 Motivation

Obviously a train does not need to do lane keeping but there are two reasons why our autonomous train needs a track detection mechanism:

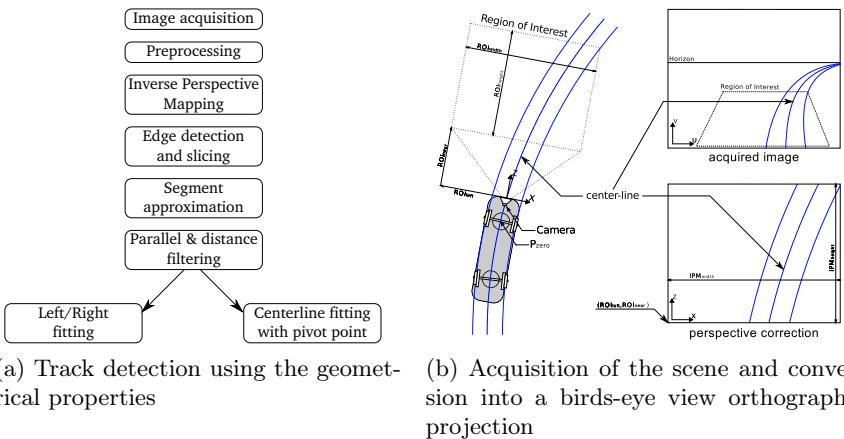
1. Provide a region of interest to the other sensor systems.

We are developing a system that uses multi-sensor fusion to achieve reliable and robust decisions. To be able to detect obstacles on or close to the track (which we will call the extended clearance gauge) the system needs to know which part of the sensor data belongs to our physical region of interest. To achieve this the track detector provides the exact position of the detected track in the image to the other sensors (RADAR, LIDAR, Stereoscopic system). Those sensors will be calibrated relative to a point on the train and thus can interpret the track information in their own coordinate system.

2. We assume that most obstacles have a connection to the ground (wheels, feet, ...) and thus would at some point disrupt the appearance of the track. Depending on the camera position, camera angle and distance of the object to the camera those discontinuities in the track are going to appear farther or closer from the train. With sufficient distance from the train those discontinuities get closer to the real distance of the obstacle. Thus this method is best suited for long distance obstacle detection where stereo and LIDAR scanners do not work properly.

### 4.2 Inverse Perspective Mapping

Based on the observations of the railway track properties we are able to design algorithms that are optimized for those scenarios. A schematic overview of the



**Fig. 2**

algorithm is shown in Figure 2a. By transformation of the perspective view into a birds-eye orthogonal view (Inverse Perspective Mapping IPM 4) we gain the ability to directly check all the geometric constraints that our algorithm requires. In addition, the transformation also makes it easier for appearance based algorithms to find matching regions because the perspective projection does no longer deform the objects depending on their location and thus the track width remains constant over the whole image.

Figure 2b shows the IPM step. The camera acquires the scene in front of the train which is transformed through a perspective projection (Figure 2b acquired image). While it is possible to find parallel lines in perspective images 11 it is much simpler if one has the undistorted view. As our algorithm heavily relies on the fact that the tracks are parallel with constant distance at all times, it makes sense to perform an IPM prior to the track detection. We also mentioned that the train undergoes a swinging which translates and rotates the camera in the world coordinate system. To be able to correctly calculate an Inverse Perspective Mapping we use the camera parameters to calculate the perspective projection of every point in the birds eye view. This is slightly more complicated than warping the input image but gives us more flexibility in dealing with the moving camera and non-planar surfaces (which are assumed by image warping techniques).

To calculate the IPM we define a region of interest in world coordinates (for example: 6 meters left, 6 meters right and from 5 to 35 meters in front of the camera). Currently we also require the world in front of the camera to be a planar surface and thus assume a  $z$ -Coordinate of zero. This however can be changed in the future once the real curvature becomes available through the LIDAR scanner. The extrinsic and intrinsic camera parameters are calibrated offline because they are not going to change once the system is in place.

$$\begin{pmatrix} x \\ y \\ z \end{pmatrix} = \begin{pmatrix} ROI_{left} + \frac{ROI_{width} * x_{ipm}}{IPM_{width}} \\ ROI_{near} + \frac{ROI_{length} * y_{ipm}}{IPM_{height}} \\ 0 \end{pmatrix} \quad (1)$$

$$\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = T_{ext} + R_{ext} * \begin{pmatrix} x \\ y \\ z \end{pmatrix} \quad (2)$$

$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{pmatrix} \frac{x' * f_x}{z'} + c_x \\ \frac{y' * f_y}{z'} + c_y \end{pmatrix} \quad (3)$$

We need to scale and offset the points in the IPM image to fit in the desired ROI (as seen in equation 1).  $ROI_{left}$  denotes the outermost left point of our (physical) region of interest and  $ROI_{near}$  defines the closest point of our region of interest. Combined with the width ( $ROI_{width}$ ) and height ( $ROI_{height}$ ) the region of interest is completely defined in physical space (see Figure 2b), because for now we assume the ground to be a flat surface. In equation 2 the extrinsic camera parameters  $R_{ext}$  (rotation of the camera) and  $T_{ext}$  (translation of the camera) are used to transform the world coordinates of our ROI into the camera coordinate system. Those points are then projected onto the image plane by

using the intrinsic camera parameters  $f_x$ ,  $f_y$  focal length and  $c_x$ ,  $c_y$  center of projection (equation 3). This finally establishes a relation between the points in the IPM image ( $x_{ipm}$ ,  $y_{ipm}$ ) and the points in the acquired image ( $u$ ,  $v$ ). This is done for every pixel in the IPM image and thus reverses the perspective projection of the camera as seen in Figure 2b *perspective correction*.

### 4.3 Algorithm

Once the IPM image has been created a *Difference of Gaussian* (DoG) filter is applied, which has been proven to be the most versatile in our tests, with rather big kernel sizes (17 for the first filter and 13 for the second) to account for the motion blur and the probably wrong focus of the fix-focus camera. This filtered image is thresholded to create a binary image which is then split vertically into 10 equally sized slices. This breaks up longer lines and allows us to find parallel line segments. It also makes the next step more robust against erroneous edges from the DoG filter.

After this step we got a binary image with blobs that are not higher than  $\frac{1}{10}th$  of the image. Those blobs are approximated by fitting a straight line segment to the boundary points. One of the strongest properties of railroad tracks is the constant distance of the tracks and thus the parallel line constraint of short track segments. In the next step the algorithm deletes all candidates that do not have a second candidate within the correct track distance and the correct angle. Those candidates with more than one correct partner are ranked higher than those with fewer partner candidates.

This already creates very good candidates (on average less than 100 track parts where about 20% to 50% do not belong to an actual track). We are using two versions of our algorithm which differ in the last step. The first one selects the correct candidates by recursively combining track candidates from different slices and fitting a 2<sup>nd</sup> order polynomial to them. Finally two longest combined candidates with the correct distance are chosen as the left and right rails. The second algorithm calculates a center-line for each parallel pair of rail segments and does RANSAC fitting of a 2<sup>nd</sup> order polynomial to points of the center-lines and the center of the pivot point ( $P_{zero}$ ) of the leading axle. This property is derived from the zero lateral offset constraint in Table II which forces the center-line of the track to always pass through  $P_{zero}$ . We also know that the camera is fixed relative to  $P_{zero}$  and thus all center-lines have to pass roughly through a single point outside the IPM image.

## 5 Examples

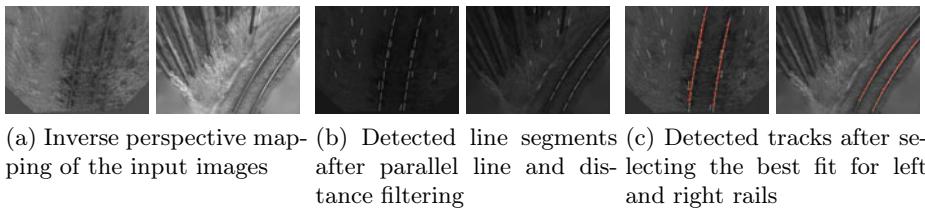
For our tests we used a *Basler scout sca1300-32gm* grayscale camera capable of acquiring 32 frames per second at a resolution of 1296x966 pixels.

Our tests have shown that this algorithm performs very well even in challenging scenarios. An overview of common scenarios is provided in Figure 11. We can see that the amount of vegetation that is allowed on the track has a large

impact (Figure 1d) on the appearance of the space between the tracks. Figure 1e shows one of the best case scenarios for track detection where the top of the rail reflects light quite well. But as already mentioned in Section 3 one can not rely on this feature. Under bad lighting conditions the brightness difference between the tracks and the surrounding area gets problematically low (Figure 1d). This remains a challenging scenario.

### 5.1 Individual Fitting

In Figure 3 we can see the output of the first variant which fits the left and right rails individually. The inverse perspective mapped images in Figure 3a represent an area of 8x30 meters starting 12 meters in front of the camera. After the detection of possible track segments and applying the geometric constraint filtering that searches for parallel partner segments with correct distance, we can see (Figure 3b) that most of the remaining candidates do actually belong to the left and right rails. The correct segments are finally determined by fitting the polynomial to various combinations of the candidate segments. Those with the highest score are chosen as shown in Figure 3c.

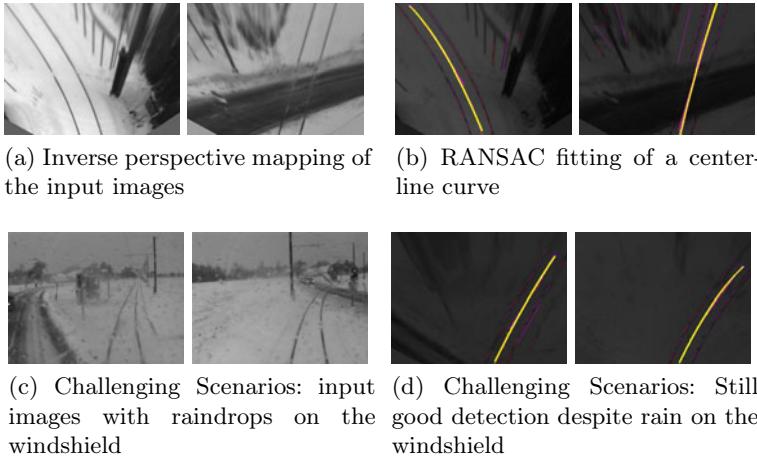


**Fig. 3.** Examples of fitting individual curves to the two tracks

### 5.2 Center-Line Fitting through a Pivot Point

The individual fitting is supplemented by the fitting of a single curve to the center-line between the left and right rails. In Figure 4b the output of the center-line fitting is shown. The right image in 4a shows a track that is covered with snow and some track candidates that can not be detected because the edges are too weak (Figure 4a right image on the street). But the knowledge that the center-line of the track must pass through the pivot point still allows the algorithm to robustly detect the track.

One of the core assumptions of this algorithm is the fact that most tracks will not be detected as a whole but rather in parts. So a core task is to find track elements that belong together even if they are not connected. We can see such scenarios where a track could never be detected in one piece in Figure 4c. Those images belong to the worst case scenarios. Although the final system is going to cope with raindrops physically it should still be able to manage those scenarios. We can see in Figure 4d that the current version of the detector is already able to find the correct position of the tracks.



**Fig. 4.** Fitting of a single curve to the common center-line of the left and right tracks

### 5.3 Reliability of the Results

Applying the track detection to each frame independently is of course more prone to errors than using the knowledge of previous frames to filter out wrong candidates (i.e. with Kalman filtering). However to show how the algorithm performs on our data we applied the detection on every frame without further knowledge of the context. Every frame was classified by us as either correct or incorrect. The test was applied on a dataset acquired in winter with average weather conditions. The tracklength is approximately 15 kilometers and the train was driving at a speed of about 40 kilometers/hour. The data was recorded at  $\sim 10$  frames per second. The track was incorrectly detected in 297 frames of a total of 13610 recorded frames. This means that the track has been correctly detected 97,81 percent of the time without using temporal constraints.

## 6 Conclusions

We have presented a novel technique that uses track detection to find the exact location of railroad tracks in an image which can be used by another system to actually detect obstacles along those tracks. The algorithm combines several techniques from lane detection in automotive systems and extends them by applying simple but strong geometric constraints that are provided by the use case (railway) itself. Those geometric constraints allow a reduction of processing cost in the final fitting stage and also generate more robust output even in very challenging scenarios.

However the initial edge detection is rather simple and not designed for the custom rail properties. This will be improved in future versions which should again increase robustness. Additionally the detection will be filtered by using the knowledge of previous frames and the estimated motion of the train to increase robustness.

## References

1. Cheng, H.-Y., Yu, C.-C., Tseng, C.-C., Fan, K.-C., Hwang, J.-N., Jeng, B.-S.: Environment classification and hierarchical lane detection for structured and unstructured roads. *IET Computer Vision* 4(1), 37–49 (2010)
2. Danescu, R., Nedevschi, S.: Probabilistic lane tracking in difficult road scenarios using stereovision. *Trans. Intell. Transport. Sys.* 10(2), 272–282 (2009)
3. Gebauer, O., Pree, W.: Towards autonomously driving trains (2008),  
<http://www.softwareresearch.net/fileadmin/src/docs/publications/C086.pdf>, Workshop for Research on Transportation Cyber-Physical Systems
4. Huang, F., Wang, R.-C.: Low-level image processing for lane detection and tracking. In: Arts and Technology. Lecture Notes of the Institute for Computer Sciences, Social Informatics and Telecommunications Engineering, vol. 30, pp. 190–197. Springer, Heidelberg (2010)
5. Jiang, G.Y., Choi, T.Y., Hong, S.K., Bae, J.W., Song, B.S.: Lane and obstacle detection based on fast inverse perspective mapping algorithm. In: IEEE International Conference on Systems, Man and Cybernetics, vol. 4, pp. 2969–2974 (2000)
6. Lim, K.H., Seng, K.P., Ang, L.-M., Chin, S.W.: Lane detection and Kalman-based linear-parabolic lane tracking. In: International Conference on Intelligent Human-Machine Systems and Cybernetics, vol. 2, pp. 351–354 (2009)
7. Lipski, C., Scholz, B., Berger, K., Linz, C., Stich, T., Magnor, M.: A fast and robust approach to lane marking detection and lane tracking. In: IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 57–60 (2008)
8. Maire, F.: Vision based anti-collision system for rail track maintenance vehicles. In: AVSS, pp. 170–175 (2007)
9. Tan, S., Dale, J., Anderson, A., Johnston, A.: Inverse perspective mapping and optic flow: A calibration method and a quantitative analysis. *Image and Vision Computing* 24(2), 153–165 (2006)
10. Fraunhofer Institut Verkehrs und Infrastruktursysteme. Hinderniserkennung für Schienenfahrzeuge (2005),  
<http://www.ivi.fhg.de/frames/german/projects/produktbl/hinderniserkennung.pdf>
11. Wang, Y., Teoh, E.K., Shen, D.: Lane detection and tracking using b-snake. *Image Vision Comput.* 22(4), 269–280 (2004)
12. Zhou, Y., Xu, R., Hu, X., Ye, Q.: A robust lane detection and tracking method based on computer vision. *Measurement Science and Technology* 17(4), 736 (2006)

# Local Descriptors for Document Layout Analysis

Angelika Garz, Markus Diem, and Robert Sablatnig

Institute of Computer Aided Automation, Computer Vision Lab  
Vienna University of Technology, Austria  
[{garz,diem,sab}@caa.tuwien.ac.at](mailto:{garz,diem,sab}@caa.tuwien.ac.at)

**Abstract.** This paper presents a technique for layout analysis of historical document images based on local descriptors. The considered layout elements are regions of regular text and elements having a decorative meaning such as headlines and initials. The proposed technique exploits the differences in the local properties of the layout elements. For this purpose, an approach drawing its inspiration from state-of-the-art object recognition methodologies – namely Scale Invariant Feature Transform (SIFT) descriptors – is proposed. The scale of the interest points is used for localization. The results show that the method is able to locate regular text in ancient manuscripts. The detection rate of decorative elements is not as high as for regular text but already yields to promising results.

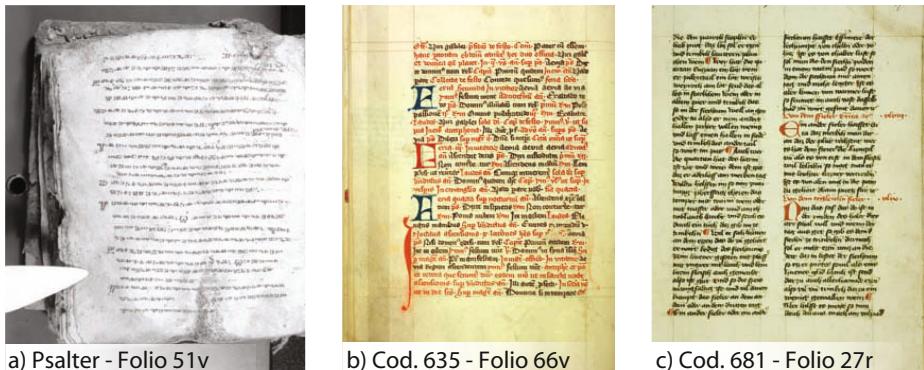
## 1 Introduction

The method presented in this paper aims at analyzing the layout of ancient manuscripts. It was developed for the analysis of the Old Church Slavonic Psalter of Demetrius (*Cod. Sin. Slav. 3N*), a Glagolitic manuscript originating from the 11<sup>th</sup> century [1]. The Psalter consists of 145 folios with a front page (recto, r) and a back page (verso, v). Additionally, the approach is applied to two manuscripts from the Austrian Stiftsbibliothek Klosterneuburg - the Codex Claustroneoburgensis 635<sup>1</sup> (Cod. 635), a manuscript consisting of 120 folios written in Latin, and the Codex Claustroneoburgensis 681<sup>2</sup> (Cod. 681), which has 167 folios written in German and Latin. The manuscripts considered in this paper comprise the following layout elements: decorative elements – such as initials and headlines – and areas of regular text. Fig. [1] shows one folio of each of the three manuscripts.

The identification and localization of decorative elements and text regions has two areas of application. On the one hand, it serves as input for Optical Character Recognition (OCR) systems [2] as it localizes regions of interest. Identifying decorative elements and text areas allows indexing manuscripts according to their contents. On the other hand, the extraction of decorative elements supports human experts studying ancient handwriting. Having identified the decorative elements of a whole manuscript, experts are enabled to retrieve and to directly work on them without the need of browsing through the document images, searching and extracting the decorative elements manually.

<sup>1</sup> Permalink: <http://manuscripta.at?ID=830>

<sup>2</sup> Permalink: <http://manuscripta.at?ID=885>



**Fig. 1.** Example folios of the three manuscripts

In the following, definitions of the terms are given, which describe decorative elements of the Psalter. Fig. 2 compares a Glagolitic **I'** and a **P** in use as decorative initial and small initial, in headlines and in the regular text.

**Decorative elements** are characters that do not belong to the regular text but have a decorative meaning such as initials or headlines. There exist two types of initials:

**Small initials** are characters having a different aspect ratio compared to letters in the regular text. The individual continuous strokes are longer for small initials than for characters in the text body and the shapes of the initials are angular.

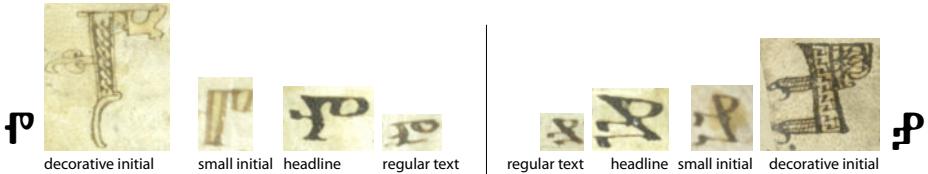
**Decorative initials** are large initials illuminated with tendrils, bows and hachure and cover more than two lines of regular text.

Subsequently, the term initial is used to refer to both, decorative and small initials. Both types of initials are optionally characterized by having outlines instead of single strokes.

**Headlines** are usually highlighted with yellow wash and have a different aspect ratio and mostly angular shapes. Additionally, non-Glagolitic letters – e.g. Cyrillic – occur in headlines.

The manuscripts Cod. 635 and Cod. 681 contain initials which are not decorated, they are written in red or blue ink and consist of bold and narrow continuous strokes. The initials are either in the left margin of the regular text or embedded in the text area as a drop cap, which means that the initial is situated within the margin and runs several lines deep into the text (see Fig. 1 a, b). Furthermore, Cod. 681 contains headlines which are written in red ink but are not different in their local structure to the regular text.

Over the intervening centuries, ancient manuscripts decay by reason of inappropriate storing conditions and materials they consist of, such as parchment or paper [3]. Thus, the manuscripts consist of veined writing support that is



**Fig. 2.** Glagolitic  $\text{F}^{\text{v}}$  and  $\text{P}$  as decorative initial, small initial, headline and character in regular text

heterogeneously textured and stained due to degradation processes, mold and moisture [3]. Additionally, the pages suffer from scratches as well as crease and are corrugated, due to characteristics of the writing support and – if applicable – because of being bound as a book. Furthermore, uneven lighting during the digitization process and fading-out of ink are challenges to cope with [4] [3]. As a result, traditional binarization-based methods used for layout analysis do not only detect characters and decorative elements, but also background clutter. This especially applies to document images having a low dynamic range, which is the case if the ink is faded-out or the paper is stained and, therefore, the contrast between characters and background diminish.

Thus, a method that is robust and takes into account the characteristics of the layout elements is proposed. Considering these layout elements as objects having intra-class similarities at the local level, an approach drawing its inspiration from the field of recent object recognition methods is chosen to analyze the layout.

The remainder of the paper is organized as follows. In the subsequent section, related work is discussed. Then, the methodology proposed is detailed in Section 3. In Section 4, results are depicted, followed by a conclusion drawn in Section 5.

## 2 Related Work

Bourgeois and Kaileh [5] propose a document analysis system that retrieves meta data such as initials, illustrations, text regions or titles. The images are first binarized and then, binary features linked to shape, geometry and color are extracted for each connected component.

In [6], a system for the indexation of ancient book collections is introduced using a texture-based method for characterizing printed documents from the Renaissance, where text and graphical areas are identified by distinct distribution of orientations.

Ramel et al. present [7] a user-driven layout analysis system for ancient printed books. They propose a two-step method, that creates a mapping of connected components to a shape map and a mapping for background areas.

Pareti et al. present an automatic retrieval system for graphical drop caps, which was developed in the course of the MADONNE project [8]. Methodologies of document image analysis are investigated in order to create an approach for the classification and indexing of drop caps.

### 3 Methodology

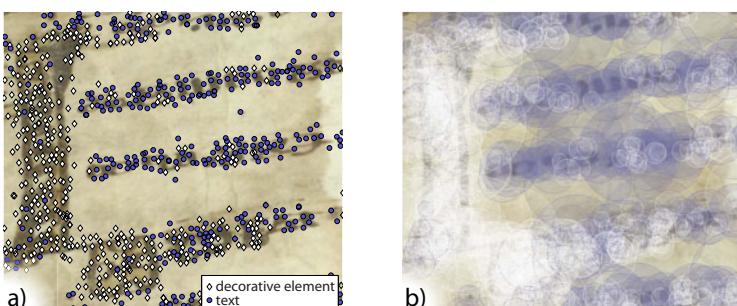
Contrary to recent approaches presented in the previous section, a binarization-free layout analysis methodology is proposed in this paper. This methodology is inspired by state-of-the-art object recognition systems and is composed of two principal steps, namely feature extraction and classification at the first stage and afterwards localization of the layout elements. Both tasks are based upon the extraction of interest points which are computed by means of the Difference-of-Gaussian (DoG).

#### 3.1 Feature Extraction and Classification

In order to extract local descriptors independent to scale changes, a scale-space is computed. The scale-space was proposed by Lindeberg [9] and is constructed by convolving the image repeatedly with a Gaussian kernel. Differencing successive levels of the Gaussian scale-space results in a DoG, which is a second order derivative scale-space which allows detecting blob like regions at local extrema. Having found the interest points of a document image, local descriptors can be computed at their location.

The proposed system implements the Scale Invariant Feature Transform (SIFT) which was proposed by Lowe [10]. High-dimensional features are computed by means of the image's gradient magnitude and gradient orientation. In order to compute them rotationally invariant, the main orientation is estimated for each interest point. Normalizing the feature vector according to the main orientation allows for a representation that is independent to rotational changes.

The local descriptors represent parts of characters such as circles, junctions, corners or endings as well as whole characters and parts of text lines. Fig. 3 shows a patch of a document image overlaid with visual representations of SIFT descriptors. The dots and diamonds in the left image denote the locations of the descriptors, whereas the circles in the right image indicate their scale. Homogeneous regions such as background are not detected at all by local descriptors.



**Fig. 3.** a) Classified SIFT descriptors with their assigned classes, b) descriptors with their respective scales

For the datasets considered in this paper, two logical classes are defined: regular text and decorative elements. This means that headlines and initials belong to the same class for the Psalter whereas headlines are considered as part of the regular text for the other manuscripts. This decision is based on the characteristics of the layout elements. The local descriptors are directly classified by a Support Vector Machine (SVM) using a Radial Basis Function as kernel.

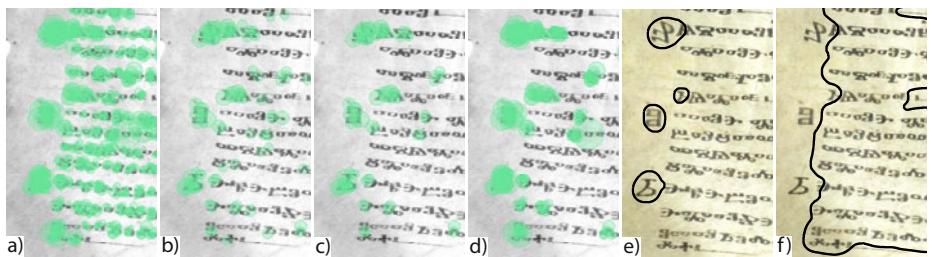
### 3.2 Localization

The fact that interest points represent parts of or even whole characters is exploited for the localization of the layout elements. However, the class decision is only done for dedicated positions of the interest points and, therefore, the location and expansion of whole objects cannot be directly transferred from the positions of the interest points.

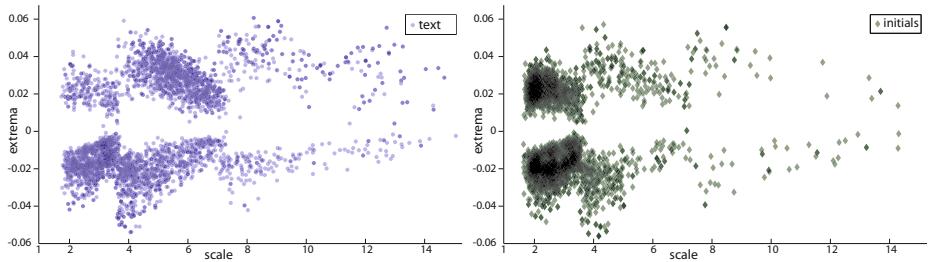
In Fig. 3 b), the SIFT descriptors are visualized by means of circles indicating the scale of their respective interest points. White circles indicate decorative elements, whereas blue circles denote text areas. A voting that rejects incorrect interest points and establishes a class decision for ambiguous regions, where interest points of different classes vote for, needs to be done.

The proposed localization algorithm consists of six stages: (1) constituting marker points, which are interest points of the second octave of the DoG, (2) conflating the marker points with the remaining interest points, (3) region-based processing to reject layout element candidates that are to small, (4) weighting of the interest points based on their scale, (5) establishing score maps for each class, and finally (6) region-based processing. Fig. 4 gives an overview of the different stages of the localization algorithm.

As can be seen in Fig. 5, three octaves are established for the scale-space. For illustration purposes, the x-axis shows the scale, whereas the extrema are depicted on the y-axis. For this application, the interest points of the second octave most reliably indicate the correct class. Thus, these interest points are



**Fig. 4.** a)-e) show the decorative element-class, a) illustrates the descriptors classified by the SVM, in b), the marker points are shown as selected by the octave, c) depicts the marker points after removing single occurrences, d) shows the marker points merged with overlapping interest points and e) presents the final result of the localization algorithm. In f) the final result for the regular text class is given.



**Fig. 5.** The scales of the classified descriptors (x-axis) plotted against their extremes (y-axis). Left text, right decorative elements.

chosen as markers denoting possible locations of layout elements. The extent of the supposed element is determined by the pixel areas the interest points cover.

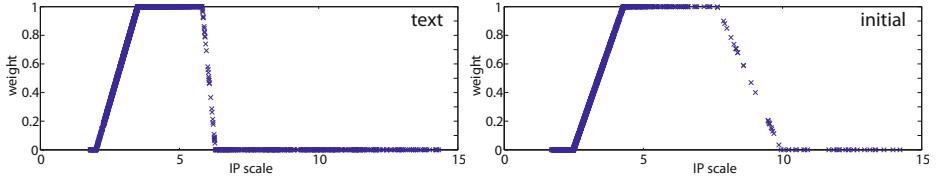
Marker points having a classification score inferior to 0.6 are rejected to prevent unreliable markers where the class decision is ambiguous. This threshold is chosen as because the counter score then is equal or less than 0.4, which means that the difference between the two scores is at least 0.2. The resulting marker points are shown in Fig. 4 b).

Subsequently, it is defined that at least two marker points must vote for an area to be considered as reliable. Therefore, only marker points having an overlap of at least 25 % are taken into account. Applying this, isolated marker points are rejected, see Fig. 4 c).

Having determined possible locations of layout elements, the before unregarded interest points are merged with the marker points to establish a precise localization. Hence, interest points overlapping at least 25 % with one marker point are added to the final set of interest points used for the subsequent processing stages.

Next, a region-based processing is applied to remove small or sparse regions. For this, overlapping interest points are joined to regions by generating a score map for each respective class as described later. Morphological operations are applied to remove areas smaller than the mean marker point. Afterwards, sparse regions, which are defined as regions having less than 10 interest points voting for it, are rejected. This reduces the number of falsely detected isolated regions.

In the fourth stage, the interest points are voted based on their scale. A linear weighting function is established, which is applied to the descriptor's maximum classification score gained from the SVM. Fig. 6 (left) shows the weighting function for decorative elements and in Fig. 6 (right), the weighting function for regular text is illustrated. The function has different slopes for small and large interest points. The inflection points of the function are determined by the minimum and maximum scales of the marker points. As can be seen, the voting functions for regular text and decorative elements have to be dissimilar due to the sizes of the characters of each class.



**Fig. 6.** Weight functions. Left: regular text, right: decorative element

The scale-based voting is motivated by the fact that the various scales represent different parts of the layout elements. Applying the weighting function, interest points which correspond to whole characters (in the case of regular text, small initials and headlines) or parts of decorative elements are stressed whereas the impact of interest points having other scales is diminished. The smallest interest points weighted by 0 are background clutter, dots, small structures of characters and speckles of the parchment. The large scales weighted by 0 represent whole decorative initials, spots and stains as well as ripples of the parchment.

As fifth step, a score map for each respective class is established based on the interest points' scales and weighted classification scores. The score map  $map_c$  for every class  $c$  is constructed by spatially weighting the scores with the Gaussian distribution  $G(x_i, y_i, \sigma_i)$ :

$$map_c = \sum_{i \in c} \omega_i \cdot G(x_i, y_i, \sigma_i) \quad (1)$$

with  $\sigma_i$  corresponding to the radius of the  $i^{th}$  local descriptor of class  $c$  and  $\omega_i$  representing the descriptor's score.

Hence, at all locations of interest points, a weighted score distribution having the same size as the interest point is generated. For each pixel in the score map, the values are accumulated. This results in one score map for each respective class representing the accumulated score for each pixel indicating the expectation belonging to the particular class.

Finally, the maximum score value of both score maps determines the class, the pixel belongs to. As applying this voting scheme may lead to small isolated regions, a final morphological operation is applied to remove these regions from the result.

## 4 Results

In this section, results are depicted. For each manuscript, a random sample of 100 pages, which have varying layouts and writing styles, is used as test set.

The evaluation is based on manually tagged ground truth, where text areas and decorative elements are brushed with gray value corresponding to their class index. The evaluation of the localization is not carried out at positions having

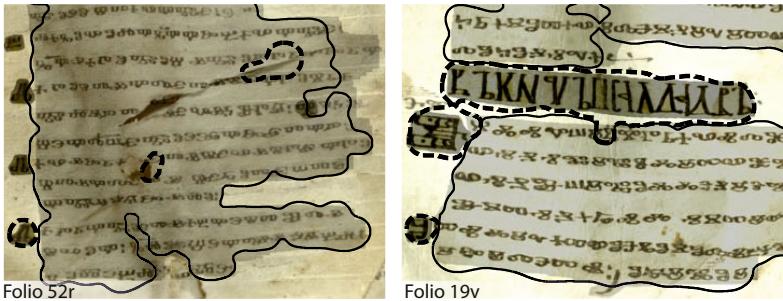


Fig. 7. Psalter - classification results



Fig. 8. Cod. 635 - classification results

overlapping class labels. Thus, a 20 px margin is added to the blobs in each ground truth image (having a mean resolution of  $2850 \times 3150$ ). This technique is motivated by two considerations which are subsequently given. On the one hand, manually tagged ground truth is tainted with noise. This noise occurs especially in border regions of overlapping classes. On the other hand – depending on the data – the classes may have a fuzzy or overlapping region border which renders an exact ground truth segmentation impossible.

Fig. 7, Fig. 8, Fig. 9 illustrate sample results obtained by the proposed methodology for each of the manuscripts. The dark blobs denote the ground truth whereas the black contours indicate the detected regions. Hereby, the lighter blobs and the continuous contours mark the regular text class, whereas the darker blobs and the dashed lines describe the regions of decorative elements.

In Table 1, the  $F_{0.5}$ -score, precision and recall of the proposed method is given. The evaluation is done per pixel. It has to be considered for the interpretation of the results that 92 % (Psalter), 96.2 % (Cod. 635) respectively 99 % (Cod. 681) of the classified pixels belong to the regular text class. First, the results for the proposed method are given (a). If additional background pixels (eg. line spacings that are not tagged as text in the ground truth but are classified by the approach as text areas) are not considered as false classification, a better  $F_{0.5}$ -score is obtained (see b).



Fig. 9. Cod. 681 - classification results

Table 1.  $F_{0.5}$ -score, Precision and Recall for the different stages

	Psalter			Cod. 635			Cod. 681		
	$F_{0.5}$ -score	Precision	Recall	$F_{0.5}$ -score	Precision	Recall	$F_{0.5}$ -score	Precision	Recall
a)	0.9135	0.9243	0.8731	0.9718	0.9689	0.9837	0.9745	0.9738	0.9773
b)	0.9453	0.9652	0.8731	0.9779	0.9765	0.9837	0.9774	0.9775	0.9773
c)	0.9304	0.9388	0.8985	0.9784	0.9744	0.9947	0.9790	0.9786	0.9806
d)	0.6293	0.6670	0.5132	0.7350	0.7653	0.6343	0.6769	0.7117	0.5660
e)	0.8726	0.9616	0.6370	0.9929	0.9919	0.9966	0.9959	0.9961	0.9949
f)	0.9522	0.9709	0.8843	0.9941	0.9937	0.9956	0.9953	0.9961	0.9923
g)	0.1982	0.1675	0.7419	0.7040	0.7523	0.5602	0.4028	0.3908	0.4591
h)	0.4987	0.4569	0.7868	0.7354	0.7509	0.6792	0.4178	0.3926	0.5615

The result for the regular text class (see c) is superior to the one of the decorative elements class (see d). Concerning the decorative element class, the results are not as satisfying as for the regular text class. In case of the Psalter, this is due to the fact, that for headlines and small initials, the difference to regular text is partly only the size and the angularity of the shapes as not all characters are written outlined. Even for humans who are no experts in the Glagolitic language, the differentiation between regular text on the one hand and small initials or headlines on the other hand is not a trivial task (see Fig. 4). Concerning the other manuscripts, the reason for the performance can be traced back to the fact that the initials lack small structures and, therefore, less interest points are established at these layout elements.

For Table 1 e-h), the evaluation is done per descriptor. Rows e, f) give the performance for the classified descriptors of the regular text class for every image. In e), the evaluation is given before applying the voting scheme on the interest points and in f) after applying it. It can be seen that the algorithm reduces the number of incorrect classified interest points.

As for the text, the voting scheme for rejecting incorrectly classified descriptors improves the evaluation results for the decorative element descriptors (see Table I g (before voting), h (after voting)).

## 5 Conclusion

In this paper, we described a layout analysis algorithm for ancient handwritten documents inspired by methodologies used in modern object recognition systems. The proposed approach exploits the fact that decorative elements can be considered as objects having similar local structures. This also applies to regular text.

This methodology is able to deal with the challenges of ancient manuscripts. Amongst these are stains and tears as well as faded-out ink. The approach shows accurate results for regular text if the object localization problem is solved.

We introduced a cascading voting scheme for the localization of layout elements based on reliable marker points, which successively reduces the number of falsely detected regions.

Future work includes improvements of the localization algorithm for decorative elements and – for the Psalter – the partitioning of decorative elements according to their topological structure, namely decorative initials, small initials and headlines.

## References

1. Miklas, H., Gau, M., Kleber, F., Diem, M., Lettner, M., Vill, M., Sablatnig, R., Schreiner, M., Melcher, M., Hammerschmid, E.G.: St. Catherine's Monastery on Mount Sinai and the Balkan-Slavic Manuscript Tradition. In: Slovo: Towards a Digital Library of South Slavic Manuscripts, Boyan Penev, pp. 13–36 (2008)
2. Diem, M., Sablatnig, R.: Recognizing Characters of Ancient Manuscripts. In: Proceedings of IS&T SPIE Conference on Computer Image Analysis in the Study of Art (2010) (accepted)
3. Kleber, F., Sablatnig, R., Gau, M., Miklas, H.: Ancient document analysis based on text line extraction. In: Proceedings of the 19th International Conference on Pattern Recognition (ICPR 2008), pp. 1–4 (2008)
4. Likforman-Sulem, L., Zahour, A., Taconet, B.: Text line segmentation of historical documents: a survey. IJDAR 9, 123–138 (2007)
5. Bourgeois, F.L., Kaileh, H.: Automatic metadata retrieval from ancient manuscripts. In: Marinai, S., Dengel, A.R. (eds.) DAS 2004. LNCS, vol. 3163, pp. 75–89. Springer, Heidelberg (2004)
6. Journet, N., Eglin, V., Ramel, J.Y., Mullot, R.: Text/graphic labelling of ancient printed documents. In: Proc. ICDAR, pp. 1010–1014 (2005)
7. Ramel, J.Y., Leriche, S., Demonet, M.L., Busson, S.: User-driven page layout analysis of historical printed books. IJDAR 9, 243–261 (2007)
8. Pareti, R., Uttama, S., Salmon, J.P., Ogier, J.M., Tabbone, S., Wendling, L., Adam, S., Vincent, N.: On defining signatures for the retrieval and the classification of graphical drop caps. In: Proc. DIAL (2006)
9. Lindeberg, T.: Scale-Space Theory: A Basic Tool for Analysing Structures at Different Scales. Journal of Applied Statistics 21(2), 224–270 (1994)
10. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60, 91–110 (2004)

# CT Image Segmentation Using Structural Analysis

Hiroyuki Hishida<sup>1</sup>, Takashi Michikawa<sup>1</sup>, Yutaka Ohtake<sup>1</sup>,  
Hiromasa Suzuki<sup>1</sup>, and Satoshi Oota<sup>2</sup>

<sup>1</sup> The University of Tokyo

{hishida,michi,yu-ohtake,suzuki}@den.rcast.u-tokyo.ac.jp

<sup>2</sup> RIKEN BioResource Center

oota@riken.jp

**Abstract.** We propose a segmentation method for blurred and low-resolution CT images focusing physical properties. The basic idea of our research is simple: two objects can be easily separated in areas of structural weakness. Given CT images of an object, we assign a physical property such as Young's modulus to each voxel and create functional images (e.g., von Mises strain at the voxel). We then remove the voxel with the largest value in the functional image, and these steps are reiterated until the input model is decomposed into multiple parts. This simple and unique approach provides various advantages over conventional segmentation methods, including preciousness and noise robustness. This paper also demonstrates the efficiency of our approach using the results of various types of CT images, including biological representations and those of engineering objects.

## 1 Introduction

Image segmentation is a fundamental problem in image processing. We are particularly interested in the problem of decomposing an object into its component parts, which involves skeletal decomposition, mechanical assembly and so on. Images typical of those that motivated this research are shown in Figs. 5(a) and 6(a) which are CT images of mouse skeleton parts. Bones themselves have a complex structure of compact bone, spongy bone and cartilage. This makes the image very ambiguous in addition to the usual limitations of CT image quality such as noise, contrast, resolution and artifacts. In particular, images of areas between two bones are not so clear enough to be decomposed. If the portions to be decomposed were limited, a template-based approach [1] would be the best choice. However, our aim here is to segment the various types of CT images - a task for which a template cannot be easily prepared.

Despite the numerous image segmentation methods that exist, none of them satisfies our objectives as discussed in Section 2. In this paper, we propose a CT image decomposition method that can be applied to various CT representations including biological and engineering objects. The technique is based on the assumption that the interfaces in an object placed between components should be

structurally weak. We compute strain which tends to be large in structurally weak areas<sup>1</sup>. We use a voxel-based structural analysis system [4] that can take a CT volumetric image directly and evaluate the strain of each voxel to output the strain distribution in the same volumetric image structure. Accordingly, any image processing methods can be applied to the output, which is referred to as a *functional image* in this research.

Main advantage of our approach is its high performance. Our method can be used to segment low-contrast CT images - a task that is difficult with conventional methods. This is based on the use of structural analysis for CT volume segmentation. Indeed, we have applied our method to various types of CT images, including those of living things and engineering object.

Though there exist some methods to use structural analysis in image processing problems [5][6], our approach is completely different from them in terms of usage of the structural analysis. We apply the structural analysis to an input image to *generate a new image* which is then used to segment the input image.

This paper has five sections. We will discuss related work in Section 2, explain the algorithm in Section 3, outline the results and engage in related discussion in Section 4, and give the conclusion in Section 5.

## 2 Related Work

The simplest method of segmentation is the thresholding approach. Since each voxel has a CT value corresponding to its density, we can extract parts with a certain threshold value. However, the CT value distribution of a material overlaps with those of other materials and contains noise, making it difficult to find the optimal threshold to distinguish regions.

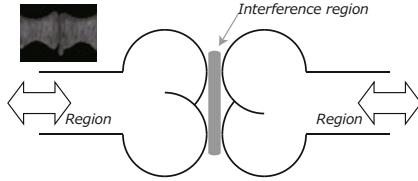
Kass et al. [7] proposed the *snakes* algorithm, which is an active contour method that minimizes spline energy for segmentation. However, it is difficult to detect exact edges. Pardo et al. [8] proposed a method for CT images of the human knee by adding information of gradient to that of region. Sebastiana et al. [9] analysed CT images of human carpal bones and succeeded in region segmentation in a narrow interference region. However, in the former, they have to adjust the weights when they apply to other data, and in the later, they need high CT gradients to segment interference regions.

Vincent et al. [10] introduced the *watershed* approach, which uses pixel gradient values to extract regions in the same way as water flooding. This tends to converge in local optimization. Grau et al. [11] proposed a method for MR images of the human brain and applied it to other CT data. Hahn et al. [12] dealt with CT images of the human anatomy with high-level performance. However in the former, erroneous segmentation occurred in some interference regions, and in the later, it is not suitable to segment homogeneity regions.

Boykov et al. [13] introduced the *graph-cut* algorithm, which uses the max-flow min-cut theorem to segment graph edges and is a powerful and highly popular

---

<sup>1</sup> For those who are not familiar with stress, strain, and structural analysis, please refer to standard textbooks e.g. [2], [3]. We also give appendices A and B.



**Fig. 1.** Analogy of breaking (*Left upper image is CT image of mouse's tail*)

segmentation method that allows many user-defined parameters. Liu et al. [14] analysed binarized CT images of human foot and also adjusted erroneously segmented areas. Shamma et al. [15] studied CT images of multi-material machine parts avoiding sink- and source-setting problems through histogram usage. Additionally, the graph-cut is widely used to segment image: i.e. [16]. However, in the former, its energy function needs to be modified for application to other CT volumes, and in the later, they can apply only to the CT images with sufficiently thick objects.

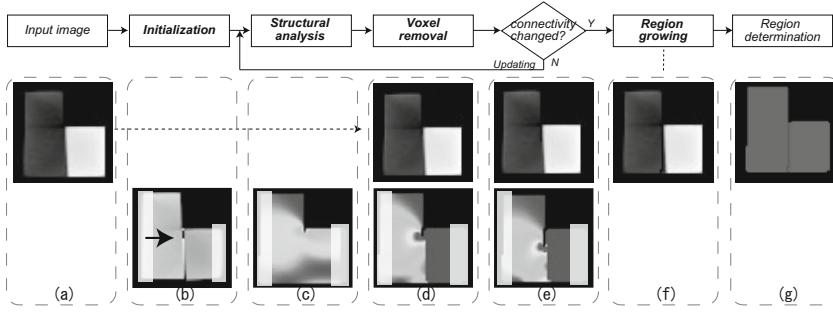
Baiker et al. [1] proposed to use articulated skeleton atlas for segmentation. They achieved exact segmentation results in various mice. However, this method required manually segmented models for estimation.

In addition to the above approaches, conventional methods have combined many different image-processing techniques. By way of example, Simina et al. [17] proposed a method based on seeded region growing [18] for CT images of human pelvic bones, and Campadelli et al. [19] proposed FMM (*the Fast Marching Method*) [20]. Other region segmentation methods were described in [21] to avoid erroneous slices in mesh model creation. Totally, disadvantages of conventional method are lack of versatility, in fact we have to largely modify their function or have to search optimal parameters.

### 3 Algorithm

In this section, we introduce a method to segment a single CT image into two or more regions. Figure 2 illustrates the method's analogy. In this figure, the region shows a set including compact bone and spongy bone, and the interference region is cartilage. We will break up the interference region.

Fig. 2 outlines the framework of the method. Our algorithm's scope is to segment various types of CT data without major modification semi-automatically. Before computation, we know the number of regions and materials in input model. Our algorithm consists of four steps. Step 1 is for initialization as shown in Fig 2(a), in which material parameters are set based on voxel CT values. Step 2 is for structural analysis [3], Appendix B as shown in Fig 2(b) and (c), and involves creating a functional image of von Mises strain [2], Appendix A in a static elastic condition. Step 3 is for voxel removal as shown in Fig 2(c), (d), (e), and (f) - we remove the voxel with the highest value in the functional image, which is a volume whose voxels have the values of the structural analysis results. After these iterations, we



**Fig. 2.** Framework of our method (The top row shows a flowchart of our method, the middle row shows CT images, and the bottom row shows functional images. We iterate structural analysis and voxel removal until the connected component is changed. (a) is input image, (b) is parameter setting, (c) is functional image in 1st iteration, (d) is 7th iterated result, (e) is 14th iterated result, (f) is divided image, and (g) is result.)

go to Step 4 which is for region growing as shown in Fig. 2(g). After the voxel removal of Step 3, the rest of the voxel sets are divided into regions, so we assume these sets as seeds and perform calculation for region growing.

Hereafter, we denote input images as  $D$ , voxels as  $x \in D$ , and the CT value of  $x$  as  $V(x)$ . We also assume that  $V(x) \geq 0$ . Here we explain our method using Fig. 2(a) as the input image. Details of this image are provided in Table 1.

### 3.1 Initialization

We extract region of interest (ROI) as  $D_f$  from  $D$  by selecting the largest component of binarized volumes using threshold  $\theta$ .  $\theta$  is decided experimentally only to remove noise as much as possible. Then small cavities in the component is also included into the ROI.

In order to apply structural analysis, we need to set boundary conditions (constraints and external forces) to some voxels in  $D_f$  and material parameters (Young's modulus and Poisson's ratio) to the rest. We assume the former as  $D_c$  and the latter as  $D_m$  ( $D_f = D_m \cup D_c$ ). As a matter of policy, we set this boundary condition to make interference region to have high von Mises strain [2], Appendix A just like we are doing in real world.

We set the same Young's modulus to each material (mentioned in Section 4.1), and also set the same Poisson's ratio to all voxels. Figure 2(b) shows the result of material properties setting.

### 3.2 Structural Analysis

We calculate the von Mises strain of each voxel  $x$  of  $D_m$  as  $\rho(x)$  using structural analysis in a static elastic condition [3, Appendix B]. We redefine the voxel set of  $D_m$  in the  $i$ -th iteration as  $D_m(i)$  because structural analysis is iterated while updating  $D_m$  at every iteration step.  $D_m(i)$  is the resulting image after structural analysis. In Fig. 2(b), the constraint is set to the rectangle without arrow and

the load to the rectangle with arrow direction, and Fig. 2(c) is produced as the structural analysis result and pixels represent their von Mises strain value.

### 3.3 Voxel Removal

Here, we define the result of structural analysis as a functional image that has von Mises strain values. In this regard, Fig. 2(c) is a functional image  $M(1)$  in the first iteration  $i = 1$ , and its voxel  $x$  has  $\rho(x)$ . We assume the voxel  $\hat{x}_i$ , which has the highest value in  $M(i)$  will be broken, so we remove  $\hat{x}_i$  from  $D_m(i)$ .

$$D_m(i+1) = D_m(i) \setminus \{\hat{x}_i\} \quad (1)$$

Next, we count the number  $k$  of connected voxel sets in  $D_m(i+1)$  in order to check region separation because voxels are removed in Eq. 1. If  $k \geq 2$ , we go to region growing (in the next step). Otherwise, we go back to structural analysis (in the previous step) with  $D_m$  updated. Figure 2(d) and (e) show a CT images and functional images in the 7th and 14 th iteration.

### 3.4 Region Growing

When  $n$  voxels  $\{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n\}$  are removed, the input data can be segmented into  $k$  parts. Now,  $D_m$  has been segmented into  $k \geq 2$  connected voxel sets  $R(j)$ ,  $j = 1, 2, \dots, k$  (Eq. 2). Figure 2(f) shows the segmented voxel sets of Fig. 2(a). In the case of Fig. 2, we iterated these steps 35 times.

$$\bigcup_{j=1}^k R(j) = D_m(1) \setminus \{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n\} \quad (2)$$

Thus, for each  $\hat{x}_i$  in  $\{\hat{x}_1, \hat{x}_2, \dots, \hat{x}_n\}$ , we calculate the shortest distance from itself to each region  $R(j)$  and update the nearest region by adding  $\hat{x}_i$  with  $R(j) := R(j) \cup \{\hat{x}_i\}$ . If there are more than one nearest regions, we calculate the absolute value of the difference between the CT value of  $\hat{x}_i$  and that of each region, and choose the smallest one as  $R(j)$ . The region segmentation result here is shown in Fig. 2(g). This is also the region growing result of Fig. 2(f).

## 4 Results and Discussion

### 4.1 Experimental Results

We applied our method to several types of CT images listed in Table 1. Fig. 3 is lapped splice welded aluminum plates and has two regions with low CT value gradient. We assumed the aluminum plates as the region and assumed welded area as an interference region, as these areas have low stiffness. Fig. 4 is phantom human carpal bones and has seven regions. Fig. 5 is mouse's femur and has two regions with curved interference region. Fig. 6 is mouse's shoulder and has two regions with radical geometry changing. In these examples, we assumed cartilage areas as interference regions because they have low stiffness.

**Table 1.** Experimental condition and results. ( $N_r$ : Number of regions;  $\theta$ : Threshold of peeling;  $T_h$ : High Young's modulus;  $T_l$ : Low Young's modulus;  $R_p$ : Poisson's ratio;  $n$ : Number of iterations;  $T(1)$ : Calculation time for first iteration;  $T$ : Total calculation time. \* is signed short condition, otherwise unsigned char condition.)

Data	Size	$N_r$	$\theta$	$T_h$ [Pa]	$T_l$ [Pa]	$t_y$	$R_p$	$n$	$T(1)$	$T$
Fig. 2	72 × 72	2	140	$7.03 \times 10^{10}$	$2.81 \times 10^7$	140	0.3*	35	2.0 sec.	70 sec.
Fig. 3	346 × 334	3	200	$6.9 \times 10^{10}$	$7.7 \times 10^9$	233	0.3*	63	30 sec.	30 min.
Fig. 4	64 × 33	7	135	$2.7 \times 10^4$	$1.1 \times 10^3$	165	0.3*	280	5.0 sec.	25 min.
Fig. 5	73 × 60	2	110	$2.7 \times 10^4$	$1.1 \times 10^3$	203	0.3*	35	3.0 sec.	2.0 min.
Fig. 6	81 × 105	2	90	100	10	188	0.3*	34	3.0 sec.	2.0 min.

These experiments were conducted on a 3.00 GHz Intel(R) Xeon(R) X5365 processor and 32.0 GB of RAM. We performed structural analysis using VOX-ELCON 2010[4], which is based on image-based CAE[22][23] and one of voxel analysis. The input numbers and calculation results for all examples are shown in Table 1.

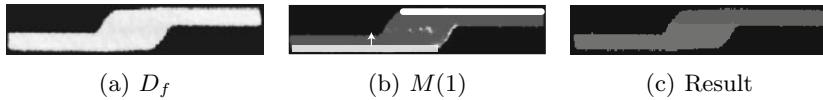
In the initialization step (outlined in Section 3), we set Young's modulus of voxel as  $Y_h$  in case which CT value of voxel is higher than  $t_y$ , otherwise we set it as  $Y_l$ . We set two Young's modulus based on the official documentation except Fig. 6. We set  $t_y$  to make almost interference region to have  $Y_l$ . And we set all Poisson's ratio as 0.3.

In Figs. 3, 4, 5, and 6, we input CT images as shown in (a). After initialization, we set constraints to white rectangle and set the load to the white rectangle with arrow which is load direction to make high von Mises strain in the interference region. (b) is structural analysis results, which are functional images  $M(1)$  including  $D_c$  and von Mises strain distribution. After iteration, we have the region-segmented images shown in (c).

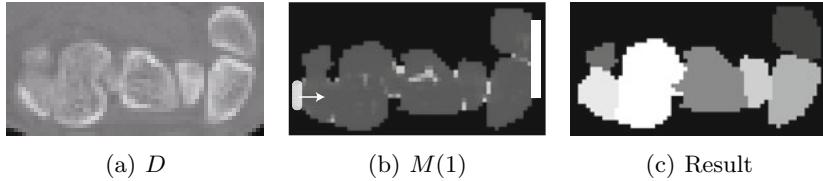
## 4.2 Evaluation of Results

**Comparison with other methods.** Our algorithm worked with difficult CT image, which the conventional method[9] could not segment, as shown in Fig. 4(c). We also compare our method to the graph-cut approach. From our experiments, the energy function of the graph-cut which is suitable to Fig. 5(a) is not suitable to Figs. 3(a) and 4(a).

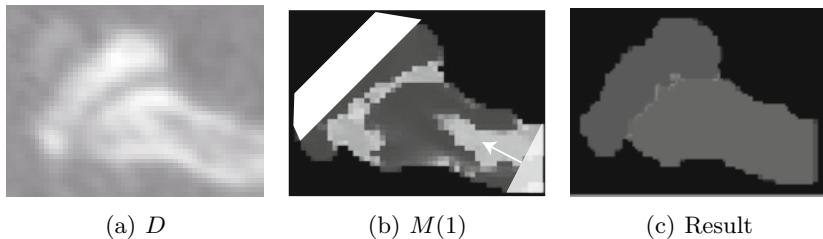
**Timing.** Timing results are shown in Table 1. Since our method is not optimized yet, there is still much room for improvement. Timing is estimated as  $T_t = (f(D_f) + T_o) \times n$ , where  $T_t$  is total calculation time,  $f(D_f)$  is time for structural analysis,  $T_o$  is time for other computing, and both  $n$  and  $D_f$  are shown in Section 3. From this equation and Table 1, the number of iteration and time for structural analysis are dominant in timing. We think that we have to remove multi voxels in every iteration or calculate structural analysis faster to make our algorithm more practical.



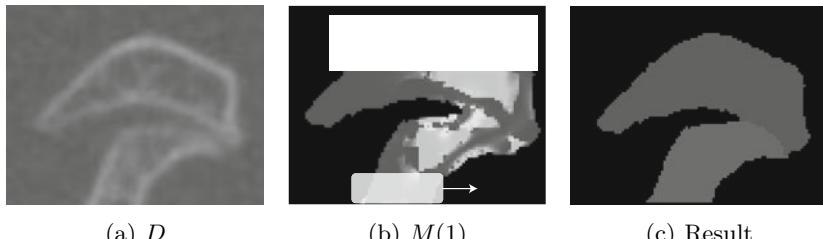
**Fig. 3.** Lapped splice welded aluminum plates ((a) is peeled-input image.)



**Fig. 4.** Phantom human carpal bones



**Fig. 5.** Mouse's femur



**Fig. 6.** Mouse's shoulder

**Robustness.** Our method is robust in terms of low resolution. An example of this is shown in Fig. 4(a), which has seven regions in a small  $64 \times 33$  area. The conventional methods [9], despite its high resolution, does not produce good results for this image.

**Sensitivity to inaccurate analysis.** We verified our method using different Young's modulus within the same boundary condition as shown in Table II. We test three types of parameter setting to Fig. 4, which are biological setting:  $T_h : 2.7 \times 10^4$  and  $T_l : 1.1 \times 10^3$ , machine setting:  $T_h : 6.9 \times 10^{10}$  and  $T_l : 7.7 \times 10^9$ ,

and simple setting:  $T_h : 100$  and  $T_l : 10$ . However, segmentation results were same. This indicates that material parameters are not sensitive to our method, thereby highlighting its advantages in terms of parameter setting.

In addition to such rough setting of material parameters, our structural analysis itself is very inaccurate for examining real physical characteristics of such objects as bones that have inhomogeneous structure of non-linear physical properties. However, as proven by the experimental results of rough setting of parameters, our segmentation method does not require an accurate analysis and can be carried out with a simple linear structural analysis.

## 5 Conclusion

We have proposed a method for segmentation of CT data by using structural analysis. First, material parameters are set to CT images and suitable boundary conditions are specified, then the level of von Mises strain is calculated. As the next step, we remove the voxel with the highest value in the functional image to imitate compression / breaking. We repeated these procedures until the object is segmented. We verified this method by applying it to various types of examples. The results confirmed that the approach works with difficult images that are hard to process using the conventional technique.

In the future work, we would like to improve boundary condition setting. Considering cases in which human knowledge cannot be used, an algorithm is needed to estimate the boundary condition. And we also would like to extract our algorithm to 3D in practical computational time.

## Acknowledgment

We appreciate Dr. Keizo Ishii, Quint corporation, for his technical support and for providing us with their VOXELCON software. One of the authors was supported through the Global COE Program "Global Center of Excellence for Mechanical Systems Innovation," by the Ministry of Education, Culture, Sports, Science and Technology. This research was partially supported by JSPS Grant-in-Aid for Scientific Research (A), 22246018, 2010 and by MEXT Grant-in-Aid for Scientific Research on Innovative Areas, 1201, 2010.

## References

1. Baiker, M., Milles, J., Vossepoel, A., Que, I., Kaijzel, E., Lowik, C., Reiber, J., Dijkstra, J., Lelieveldt, B.: Fully automated whole-body registration in mice using an articulated skeleton atlas. In: 4th IEEE International Symposium on Biomedical Imaging: From Nano to Macro. ISBI 2007, pp. 728–731 (2007)
2. Prévost, J.: Mechanics of a Continuous porous media. International Journal of Engineering Science 18, 787–800 (1980)
3. Zienkiewicz, O., Morice, P.: The finite element method in engineering science. McGraw-Hill, London (1971)

4. Quint (Voxelcon), <http://www.quint.co.jp/pro/vox/index.htm>
5. Hamasaki, T., Yoshida, T.: Image Segmentation based on Finite Element Analysis (*in Japanese*). IEIC Technical Report (Institute of Electronics, Information and Communication Engineers) 100, 1–8 (2000)
6. Pham, Q.C., Vincent, F.C.P.C.P., Magnin, I.: A FEM-based deformable model for the 3D segmentation and tracking of the heart in cardiac MRI. In: Proceedings of the 2nd International Symposium on Image and Signal Processing and Analysis. ISPA 2001, pp. 250–254 (2001)
7. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1, 321–331 (1988)
8. Pardo, X., Carreira, M., Mosquera, A., Cabello, D.: A snake for CT image segmentation integrating region and edge information. Image and Vision Computing 19, 461–475 (2001)
9. Sebastian, T., Tek, H., Crisco, J., Kimia, B.: Segmentation of carpal bones from CT images using skeletally coupled deformable models. Medical Image Analysis 7, 21–45 (2003)
10. Vincent, L., Soille, P.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Transactions on Pattern Analysis and Machine Intelligence 13, 583–598 (1991)
11. Grau, V., Mewes, A., Alcaniz, M., Kikinis, R., Warfield, S.: Improved watershed transform for medical image segmentation using prior information. IEEE Transactions on Medical Imaging 23, 447 (2004)
12. Hahn, H., Wenzel, M., Konrad-Verse, O., Peitgen, H.: A minimally-interactive watershed algorithm designed for efficient CTA bone removal. Computer Vision Approaches to Medical Image Analysis, 178–189 (2006)
13. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient nd image segmentation. International Journal of Computer Vision 70, 109–131 (2006)
14. Liu, L., Raber, D., Nopachai, D., Commean, P., Sinacore, D., Prior, F., Pless, R., Ju, T.: Interactive separation of segmented bones in ct volumes using graph cut. In: Metaxas, D., Axel, L., Fichtinger, G., Székely, G. (eds.) MICCAI 2008, Part I. LNCS, vol. 5241, pp. 296–304. Springer, Heidelberg (2008)
15. Shammaa, M., Suzuki, H., Ohtake, Y.: Extraction of isosurfaces from multi-material CT volumetric data of mechanical parts. In: Proceedings of the 2008 ACM Symposium on Solid and Physical Modeling, pp. 213–220. ACM, New York (2008)
16. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22, 888–905 (2000)
17. Vasilache, S., Najarian, K.: Automated bone segmentation from Pelvic CT images. In: IEEE International Conference on Bioinformatics and Biomedicine Workshops. BIBMW 2008, pp. 41–47 (2008)
18. Adams, R., Bischof, L.: Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence, 641–647 (1994)
19. Casiraghi, E., Campadelli, P., Pratissoli, S.: Fully Automatic Segmentation of Abdominal Organs from CT Images using Fast Marching Methods (2008)
20. Sethian, J., et al.: Level set methods and fast marching methods, vol. 11 (2003)
21. Wang, L., Greenspan, M., Ellis, R.: Validation of bone segmentation and improved 3-D registration using contour coherency in CT data. IEEE Transactions on Medical Imaging 25 (2006)
22. Kikuchi, N., et al.: Adaptive finite element methods for shape optimization of linearly elastic structures. Computer Methods in Applied Mechanics and Engineering 57, 67–89 (1986)

23. Noboru, B., Philip, M.: Generating optimal topologies in structural design using a homogenization method. Computer Methods in Applied Mechanics and Engineering 71, 197–224 (1988)
24. Baldonado, M., Chang, C., Gravano, L., Paepcke, A.: The Stanford digital library metadata architecture. International Journal on Digital Libraries 1, 108–121 (1997)

## A Stress and Strain

In continuum mechanics[2], deformation of a continuous body  $B$  is analyzed. The deformation at a point  $x$  of  $B$  is represented as a vector  $u(x)$ . A typical problem is to find  $u(x)$  under the boundary conditions such that some parts of  $B$  are fixed and external forces (load) are applied to some parts of  $B$ . The fundamental equations in the continuum mechanics are given in terms of strain and stress. The strain is the change of deformation and thus given in the form of strain tensor as  $\epsilon[\epsilon_{ij}] = (\partial u / \partial x \cdot \partial u / \partial x^T) / 2$ . The relation between stress and strain is expressed by Hooke's law for linear elastic materials and can be defined as  $\sigma[\sigma_{ij}] = c_{ijkl}\epsilon_{kl}$ . The coefficients are determined by two material constants; Young's modulus and Poisson's ratio. In order to evaluate the total quantity of the stress exerted at a point, assuming  $\sigma_i$  as an eigen value of the stress tensor, a quantity  $\sqrt{\frac{1}{2}\{(\sigma_1 - \sigma_2)^2 + (\sigma_2 - \sigma_3)^2 + (\sigma_3 - \sigma_1)^2\}}$  called the equivalent stress or von Mises stress is commonly used. If von Mises stress and strain exceed some limit, the yielding of materials begins at the point.

## B FEM

Among various kinds of methods to obtain stress and strain, FEM (Finite Element Method)[3] is the most popular numerical method to obtain stress and strain. In FEM,  $B$  is decomposed into a set of elements and the stress and strain are computed for each of the elements under given boundary conditions. Thus von Mises stress is also computed for each of the elements.

# Phase Space for Face Pose Estimation

Jacob Foytik, Vijayan K. Asari,  
R. Cortland Tompkins, and Menatoallah Youssef

Computer Vision and Wide Area Surveillance Laboratory, Department of Electrical  
and Computer Engineering, University of Dayton, Dayton, Ohio

**Abstract.** Face pose estimation from standard imagery remains a complex computer vision problem that requires identifying the primary modes of variance directly corresponding to pose variation, while ignoring variance due to face identity and other noise factors. Conventional methods either fail to extract the salient pose defining features, or require complex embedding operations. We propose a new method for pose estimation that exploits oriented Phase Congruency (PC) features and Canonical Correlation Analysis (CCA) to define a latent pose-sensitive subspace. The oriented PC features serve to mitigate illumination and identity features present in the imagery, while highlighting alignment and pose features necessary for estimation. The proposed system is tested using the Pointing'04 face database and is shown to provide better estimation accuracy than similar methods including Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and conventional CCA.

## 1 Introduction

Head pose estimation is a relatively trivial task for the human visual system. From an early age, people have the natural ability to effortlessly recognize the orientation and movements of a human head which allow for non-verbal communication and a heightened sense of awareness. However, creating an automatic face pose estimation system based on standard imagery has proven to be a complex problem. The difficulty of the problem lies in the task of adequately extracting latent *pose* information that is invariant to face identity change and other noise factors. In terms of subspace methodology, the extracted features must maximize the correlation of an image with respect to a given pose variable. It must also maintain the ability to generalize over face identity, allowing accurate pose estimation for individuals not trained by the system.

Many techniques exist for estimating face pose as summarized by Murphy-Chutorian and Trivedi in [1]. We focus on the manifold embedding class of techniques, where we consider each input image as one sample in a high dimensional space. It is well known that the given input patterns are not in a form that is suitable for identifying pose, but can be transformed to a more compact space that contains a higher density of valuable features. Since the orientation of the head has three degrees of freedom, *yaw*, *pitch*, and *roll*, the observed high-dimensional image should theoretically lie in a low-dimensional space constrained by the allowable pose variation [1]. Furthermore, the face images vary smoothly as pose

changes, creating a continuous manifold in the subspace. Therefore, it is possible to create a transform to project the raw face images to the subspace and embed the features onto a pose manifold to predict the orientation of the face.

Common methods of dimensionality reduction and manifold embedding techniques that have been applied to face pose estimation include Principal Component Analysis (PCA), Linear Discriminant Analysis (LDA), and their nonlinear kernel variants [2,3,4]. PCA creates a subspace based on the primary modes of variation in the input patterns, which relies on the assumption that pose changes account for the majority of the image variance. This method is suitable for estimating the face pose of a single individual trained by the system, but lacks the ability to generalize over multiple face identities. The unsupervised nature of the algorithm does not permit direct association of image variance with the corresponding pose information. Supervised methods such as LDA utilize class parameters to maximize the signal-to-noise ratio of the training set. In the case of pose estimation, features describing pose variation are considered signal, while identity, lighting, and other noise features are deemed noise. LDA, though a supervised method, is a class based approach and is not suitable for the continuous nature of pose changes.

We present a new method for estimating face pose from images using Phase Congruency (PC) features [5] and Canonical Correlation Analysis (CCA) [6]. *Localized* PC features have been shown to provide invariance to illumination and image contrast in the case of feature extraction for face recognition [7]. It is our hypothesis that *global* PC features will highlight the spatial configuration of facial features and the overall contour of the face, which are proven to be essential in the orientation estimation process performed by the human visual system [8]. Additionally, CCA is shown to more effectively identify pose variances due to the supervised correlation based methodology. PC features are projected to a CCA subspace to define a latent pose feature space and pose manifold used for estimation.

The paper is organized as follows. Section 2 describes the theoretical approach of the proposed method which includes a brief review of both Phase Congruency and Canonical Correlation Analysis calculations. Next, section 3 describes the overall process and operation of the system. In section 4 the experimental procedure and results are given, which include a comparison of subspace methods with regard to pose estimation and analysis of estimation accuracy using PC features. Finally, conclusions will be made in section 5.

## 2 Theoretical Approach

### 2.1 Phase Congruency Features

Standard intensity images of faces contain a vast variety of information, many of which does not pertain to the task of face pose estimation and hinders system accuracy. We seek to find a better representation of the face image that increases pose sensitivity, while mitigating variance due to other factors. There have been several techniques that have used edge based operators to perform this task,

such as [9] and [10]. Both methods use oriented Gabor filters to create an edge image that is more lighting invariant. However, such gradient based operators fail to detect a large proportion of features within an image. Such methods generally exploit features found from analysis of the image frequency *magnitude*, as opposed to the frequency *phase*. It has been shown that the phase component of the Discrete Fourier Transform is more important in the image reconstruction process than the magnitude component [11]. Thus, we explore the use of the phase congruency transform for defining a pose sensitive representation.

Phase congruency provides a metric that is independent of the overall magnitude of the signal, allowing invariance to image illumination and contrast. This is obtained by defining the significance of features in an image in terms of the phase of the Fourier components. Phase congruent features are perceived at points in an image where the Fourier components are maximally in phase [5]. Mathematically, the phase congruency function in terms of the Fourier series expansion of a signal at a given location  $x$  is

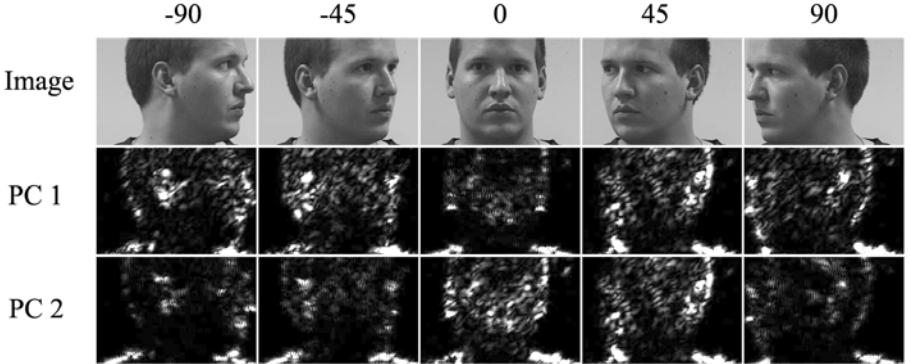
$$\text{PC}(x) = \frac{\sum_n A_n \cos(\phi_n(x) - \bar{\phi}(x))}{\sum_n A_n} \quad (1)$$

where  $A_n$  is the amplitude of the  $n^{th}$  Fourier component,  $\phi_n(x)$  is the *local* phase of component  $n$ , and  $\bar{\phi}(x)$  is the weighted mean of all local phase angles at the point being considered [5]. A more simplistic explanation is given by the equation

$$\text{PC}(x) = \frac{|E(x)|}{\sum_n A_n} \quad (2)$$

where  $|E(x)|$  is termed the *Local Energy* at a given point  $x$ . This energy term is formulated through complex addition of the Fourier components at  $x$ , where each component consists of amplitude  $A_n$  and phase  $\phi_n(x)$ . It is noted that if all Fourier components are completely in phase, then the *local energy* term will approach  $\sum_n A_n$ , resulting in  $\text{PC}(x) = 1$ .

The phase congruency technique used in this paper is based on the work developed by Peter Kovesi [5], in which the local energy is approximated using logarithmic Gabor functions. The Gabor filters allow for arbitrarily large bandwidth filters of varying orientation to be constructed which permits the selection of feature specific frequencies. Figure 11 shows the resulting phase congruency images for two distinct filter orientations for a set of pose varying faces. It is important to note that different orientations of the phase congruency transform highlight different face features as evident in the figure. For example, when dealing with profile images ( $-90^\circ$  and  $90^\circ$ ), the first phase congruency orientation appears to be more selective of the ear features than the second orientation. Therefore we must identify which orientations of the filter are most effective in the pose estimation problem. This selection process is performed empirically by identifying the filter orientations that maximize correlation with the pose variable obtained from Canonical Correlation Analysis.



**Fig. 1.** Illustration of Phase Congruency features of two orientations. Face images were obtained from the *Pointing'04* face database [12].

## 2.2 Canonical Correlation Analysis

Canonical correlation analysis contains many interesting properties that distinguish it from other subspace methods. Though other subspace methods such as PCA and LDA have received widespread attention and proven broad applicability, they are not suitable in all situations. The PCA methodology is to define orthogonal basis vectors that minimize the reconstruction error [13]. This technique is not well suited for the task of regression, where we define a mapping  $\phi : \mathbf{x} \rightarrow \mathbf{y}$ . In the case of pose estimation, we wish to map the input image to the pose variable. There is no way to be sure that the extracted features from PCA best describe the underlying relationship of  $\mathbf{x}$  and  $\mathbf{y}$ . The PCA process is only concerned about the reconstruction error, and may have even discarded the desired underlying relationship [14].

Canonical correlation analysis was first proposed by H. Hotelling [6][15], and was intended for identifying the most basic, or *canonical*, forms of correlation between two multivariate sets. Given two zero-mean random variables  $\mathbf{x} \in \mathbb{R}^p$  and  $\mathbf{y} \in \mathbb{R}^q$ , CCA finds pairs of directions  $\mathbf{w}_x$  and  $\mathbf{w}_y$  that maximize the correlation between the projections  $x = \mathbf{w}_x^T \mathbf{x}$  and  $y = \mathbf{w}_y^T \mathbf{y}$ . More specifically, the objective of CCA is to maximize the correlation coefficient given by:

$$\rho_{xy} = \frac{E[xy]}{\sqrt{E[x^2]E[y^2]}} = \frac{\mathbf{w}_x^T \mathbf{C}_{xy} \mathbf{w}_y}{\sqrt{\mathbf{w}_x^T \mathbf{C}_{xx} \mathbf{w}_x \mathbf{w}_y^T \mathbf{C}_{yy} \mathbf{w}_y}} \quad (3)$$

where  $\mathbf{C}_{xx} \in \mathbb{R}^{p \times p}$  and  $\mathbf{C}_{yy} \in \mathbb{R}^{q \times q}$  are the within-set covariance matrices of  $\mathbf{x}$  and  $\mathbf{y}$ , and  $\mathbf{C}_{xy} \in \mathbb{R}^{p \times q}$  denotes the between-set covariance matrix [14].

Though equation 3 cannot be written as a Rayleigh quotient, it can be shown that the critical points of this function coincide with the critical points of a Rayleigh quotient of the form [16]

$$r = \frac{\mathbf{w}^T \mathbf{A} \mathbf{w}}{\mathbf{w}^T \mathbf{B} \mathbf{w}}, \quad (4)$$

where  $\mathbf{w} = [\mathbf{w}_x \mathbf{w}_y]^T$ . This property is upheld only when matrices  $\mathbf{A}$  and  $\mathbf{B}$  take the form of [16]

$$\mathbf{A} = \begin{bmatrix} 0 & \mathbf{C}_{xy} \\ \mathbf{C}_{yx} & 0 \end{bmatrix}, \quad \mathbf{B} = \begin{bmatrix} \mathbf{C}_{xx} & 0 \\ 0 & \mathbf{C}_{yy} \end{bmatrix}. \quad (5)$$

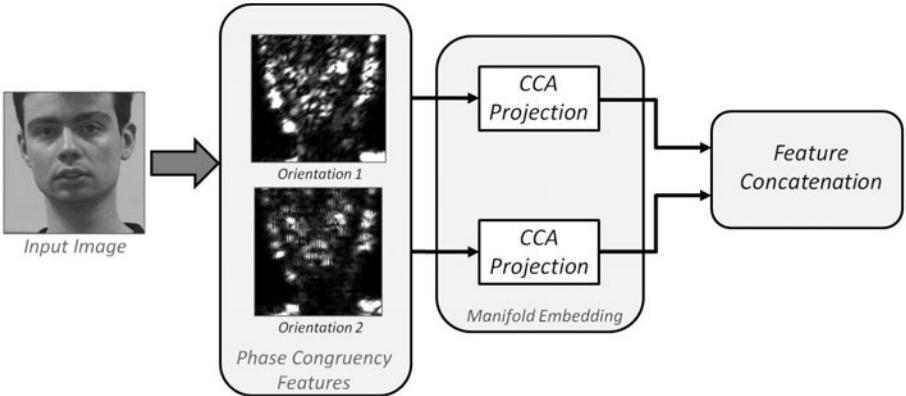
Thus, the basis vectors that maximize  $\rho$ , denoted as  $\mathbf{w}^*$ , are the same vectors that maximize  $r$  [16]. It is also shown by Borga [16] that the Rayleigh maximization problem can be reformulated as a generalized eigenproblem of the form

$$\mathbf{Aw} = \mu \mathbf{Bw}. \quad (6)$$

The maximum values  $\rho(\mathbf{w}^*)$  are called the canonical correlations and are equivalent to  $\rho(\mathbf{w}^*) = r(\mathbf{w}^*) = \mu(\mathbf{w}^*)$ . It is important to note that the number of canonical correlations is limited to the minimum dimension of the input variables. That is, the number of basis vectors extracted is  $\min(p, q)$ . In the case of pose estimation,  $\mathbf{x}$  is the input image with  $p$  denoting image resolution, while  $\mathbf{y}$  is the pose variable with  $q$  denoting the degrees of face pose movement. In the case of our tests, we only analyze pose changes in the yaw direction, setting  $q = 1$ . Thus, CCA is limited to only extracting one canonical correlation.

### 3 Algorithmic Process

The core of the pose estimation system lies in the feature extraction and concatenation process. In order to extract pose-sensitive features, we must first train basis vectors that identify the maximum mode of correlation in the phase congruency images. This is achieved by first transforming the training images to oriented phase congruency images using the methods described in section 2.1. This results in a labeled training set of pose varying face images consisting of a set of  $I$  phase congruency images of varying orientation and scale, as well as the ground truth pose labels. For training, each set of phase congruency images of distinct orientation and scale are treated as independent feature sets, resulting in the extraction of independent canonical correlation basis vectors. As stated before, the number of feature vectors that CCA can extract is limited by the observation variable dimension. In our case, the pose estimation systems are tested for yaw pose changes, resulting in a 1 dimensional vector, permitting only one feature vector to be extracted. However, by performing CCA on different representations of the input images (*i.e.*, *oriented phase congruency images*), we are able to extract multiple feature vectors to describe the data. It is then possible to retain only the orientation information that is most influential in the pose estimation process by analyzing the mean square error of each orientation independently. The estimation process uses the trained basis vectors to project the probe phase congruency features to the pose sensitive subspace. This process is shown in figure 2. As shown in the figure, the input image is transformed to oriented phase congruency images, where each orientation is projected to feature space using the corresponding trained feature vectors. The feature scores



**Fig. 2.** Feature extraction flow chart. Face images are transformed to phase congruency features, then projected to the feature space using orientation dependent CCA transforms. Feature scores are then concatenated to formulate a single feature vector describing the image in the pose-sensitive subspace.

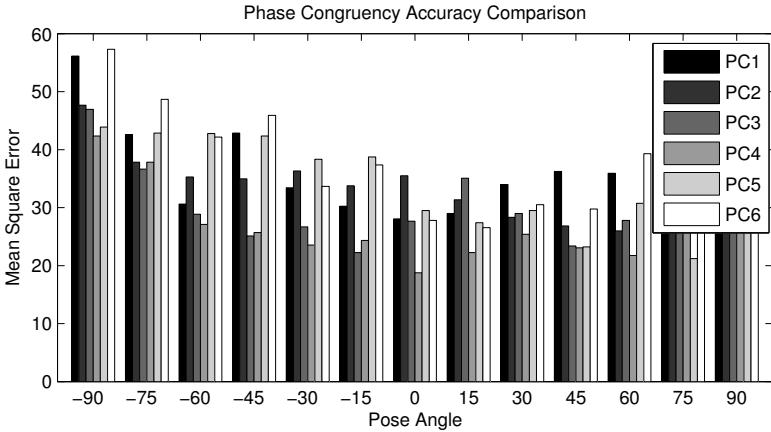
for each orientation are then concatenated to make a single multi-dimensional feature score. Pose estimation is then performed in the latent subspace by finding the nearest neighbor between the probe image and the training images.

## 4 Experimental Results

The proposed pose estimation system was implemented and compared to both PCA and LDA based methods. The systems were tested using the *Pointing'04* face database [12]. The database provides labeled pose varying face images in both the yaw and pitch direction. However, for our tests we limit the scope to yaw changes from  $-90^\circ$  to  $90^\circ$ . The input images were cropped and down sampled to  $64 \times 42$  resolution, resulting in a 2688 dimensional observation vector. It is noted that the *Pointing'04* database contains registration errors that mimic “real-life” input that would be obtained from a face detection algorithm. No effort was made to register the images, allowing for a practical analysis of the various methods.

Analysis was performed using a leave-one-out strategy. The database consists of 15 people with 2 separate imaging sessions per person. Each iteration of the leave-one-out process involved training the systems using 29 sessions, while reserving the remaining for testing. For PCA, raw images were input to the training process, and the two most weighted components were retained for testing. The raw images and ground truth pose labels were input to the LDA system, where the two most significant components were retained for testing as well. For the proposed system, termed PC/CCA, the images were transformed to phase congruency images of one scale and six orientations. The two most accurate orientations were chosen for testing.

A Euclidean nearest neighbor classifier was used in feature space to determine the closest match between the probe and training image set. The mean square

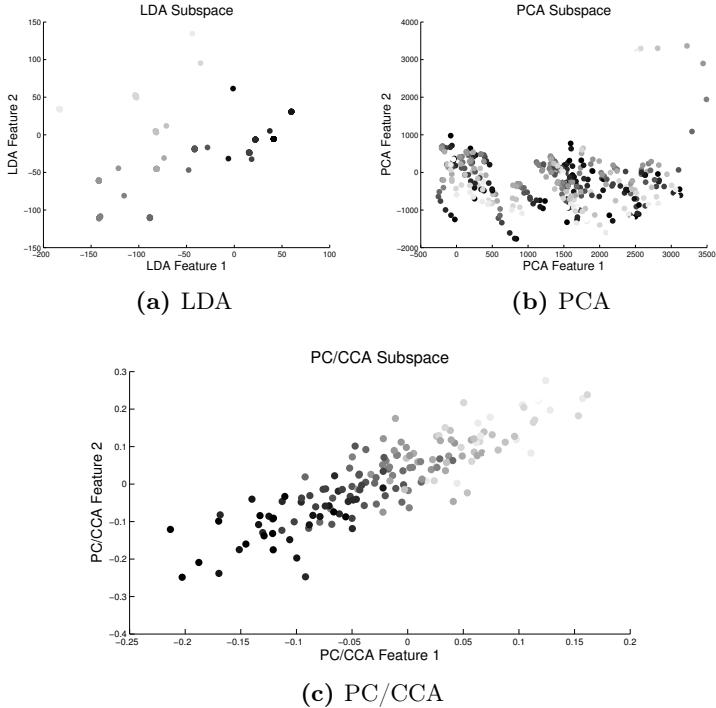


**Fig. 3.** Accuracy analysis of Phase Congruency features. MSE scores for each PC orientation ( $0, \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{2\pi}{3}$ , and  $\frac{5\pi}{6}$ ) as obtained from a leave-one-out test process.

error between the estimated and ground truth pose was used for comparing the systems and identifying the most significant PC features. Figure 3 shows the accuracies for the phase congruency orientations  $0, \frac{\pi}{6}, \frac{\pi}{3}, \frac{\pi}{2}, \frac{2\pi}{3}$ , and  $\frac{5\pi}{6}$  in radians. From the figure, we see that on average,  $\frac{\pi}{3}$  and  $\frac{\pi}{2}$  has the least MSE score and are selected for testing.

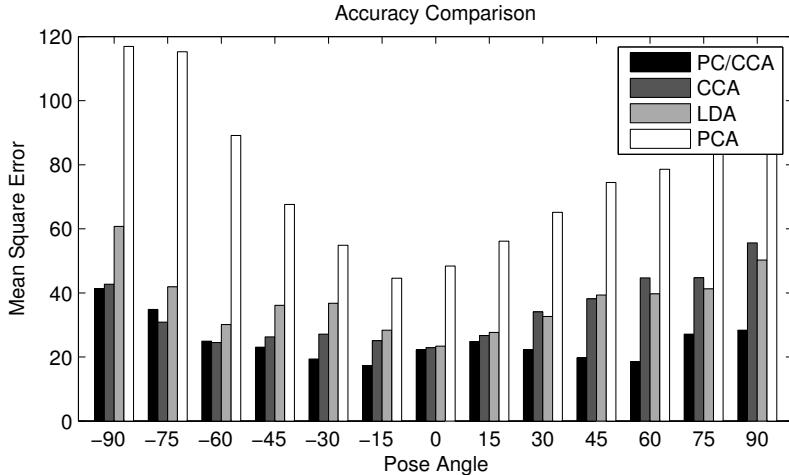
Figure 4 provides visual analysis of the effectiveness of each algorithm including LDA, PCA, and PC/CCA. The data shown was calculated from the leave-one-out process and shows how well the methods describe face pose changes. In all of the plots, each point indicates the position in feature space of a face image, while the shading of the point (*dark to light*) represents the actual pose of the image. The darkest points correspond to faces of pose  $-90^\circ$ , while the lightest points are faces of pose  $90^\circ$ . From figure 4a, we see that LDA provides distinct separation for each pose due to the discriminant process, however its usability is limited to a purely class based estimation process. It is apparent from the subspace that LDA has no capability for estimating unseen views. The PCA subspace, shown in figure 4b, provides a continuous manifold but fails to identify the modes of variation corresponding to pose change. Figure 4c shows that the PC/CCA method creates a more effective subspace that essentially formulates a linear pose manifold that allows for continuous pose estimation rather than discrete class based estimation. The manifold highly correlates with the pose angle of the face images and provides better generalization across face identity.

Figure 5 shows the pose estimation accuracy of all the systems in terms of the mean square error for each discrete pose. In addition to the already discussed techniques, CCA was also performed using the image data and pose labels to define a single feature vector used for estimation. As the chart shows, the PC/CCA method outperforms the other methods for nearly every face pose. As expected, the PCA performed the worst due to its inability to distinguish between face identity and pose variation. The unsupervised nature of PCA makes it



**Fig. 4.** (a) Linear Discriminant Analysis feature subspace (b) Principal Component Analysis feature subspace (c) Phase Congruency / Canonical Correlation Analysis feature subspace created using the two most effective phase congruency orientations ( $\frac{\pi}{3}$  and  $\frac{\pi}{2}$ ), where the intensity gradient of the points from dark to light represents the ground truth pose variable.

unsuitable for the task of pose estimation. Though LDA performed nearly as well as the standard CCA method, it should be noted that LDA requires a strictly class based classifier and cannot perform generalization between the pose classes. This is observed in figure 4a, where we see no connectivity between classes. Finally, the results show that PC/CCA performs better than conventional CCA, which is likely due to the inability of CCA to extract more than one feature vector from the data set. Using the oriented phase congruency images allows for multiple correlation modes to be defined, providing greater accuracy. Additionally, the enhanced accuracy is attributed to the phase congruency features providing a representation of the face images that is more sensitive to pose changes. The performance advantage of PC/CCA is particularly evident for poses of  $-90^\circ$  and  $90^\circ$ , where it maintains a mean error of approximately  $20^\circ$  less than LDA. This advantage can be attributed to the underlying manifold generated by PC/CCA which represents the extreme pose angles as the end caps of the structure. Table II gives a summary of the test results for all systems. As observed in the table, the



**Fig. 5.** Pose estimation accuracy comparison. MSE scores per pose for each method including PC/CCA, CCA, LDA, and PCA.

**Table 1.** Pose estimation accuracy overview

Method	PCA	LDA	CCA	PC/CCA
Overall MSE	77.87	37.55	34.11	24.91

PC/CCA method performed class-based pose estimation with an overall error of 9.2° less than the other methods.

## 5 Conclusion

We have presented a new method for pose estimation that exploits oriented phase congruency images and canonical correlation analysis to define a linear pose manifold. The phase congruency features provide a pose-sensitive representation of the original face images by eliminating illumination variances, as well as mitigating identity variances. Additionally, the oriented features serve to synthesize multiple canonical correlation basis vectors, providing enhancements over the conventional CCA process which is limited by the input observation dimension.

The proposed face pose estimation method was tested using the *Pointing’04* face database and compared to similar subspace methods using a leave-one-out testing strategy. It is shown that the PC/CCA method more accurately defines the latent pose subspace by formulating a continuous linear manifold in the feature space that highly correlates with face pose angle. The results indicate that the PC/CCA system performed class based estimation with a lower error rate than conventional PCA, LDA, and CCA. It is expected that the system could perform better using a face data set with aligned/registered face images.

Additionally, the PC/CCA method is expected to perform well for estimating novel views unseen by the training system. Future work for this topic includes testing the system performance using other pose varying face databases, definitively measuring the invariance of the method to illumination, expression, and registration variances, as well as using non-linear methods to more effectively model the latent pose manifold.

## References

1. Murphy-Chutorian, E., Trivedi, M.M.: Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31, 607–626 (2008)
2. Sherrah, J., Gong, S., Ong, E.J.: Face distributions in similarity space under varying head pose. *Image and Vision Computing* 19, 807–819 (2001)
3. Zhu, Y., Fujimura, K.: Head pose estimation for driver monitoring. In: IEEE, 2004 Intelligent Vehicles Symposium, pp. 501–506 (2004)
4. Chen, L., Zhang, L., Hu, Y., Li, M., Zhang, H.: Head pose estimation using fisher manifold learning. In: AMFG 2003: Proceedings of the IEEE International Workshop on Analysis and Modeling of Faces and Gestures, Washington, DC, USA, p. 203. IEEE Computer Society, Los Alamitos (2003)
5. Kovesi, P.: Edges are not just steps. In: The 5th Asian Conference on Computer Vision (2004)
6. Hotelling, H.: Analysis of a complex of statistical variables into principal components. *Journal of Educational Psychology* 24, 498–520 (1933)
7. Gundimada, S., Asari, V.: A novel neighborhood defined feature selection on phase congruency images for face recognition in the presence of extreme variations. *International Journal of Information Technology* 3, 25–31 (2006)
8. Wilson, H.R., Wilkinson, F., Lin, L.M., Castillo, M.: Perception of head orientation. *Vision Research* 40, 459–472 (2000)
9. Kruger, V., Bruns, S., Sommer, G.: Efficient head pose estimation with gabor wavelet networks. In: Proc. British Machine Vision Conference, pp. 12–14 (2000)
10. Ma, B., Zhang, W., Shan, S., Chen, X., Gao, W.: Robust head pose estimation using lgbp. In: International Conference on Pattern Recognition, vol. 2, pp. 512–515 (2006)
11. Gundimada, S., Asari, V.K.: Facial recognition using multisensor images based on localized kernel eigen spaces. *Trans. Img. Proc.* 18, 1314–1325 (2009)
12. Gourier, N., Hall, D., Crowley, J.L.: Estimating face orientation from robust detection of salient facial features. In: Proceedings of Pointing 2004, ICPR, International Workshop on Visual Observation of Deictic Gestures, Cambridge, UK (2004)
13. Bishop, C.M.: *Pattern Recognition and Machine Learning* (Information Science and Statistics), 1st edn. Springer, Heidelberg (2007)
14. Melzer, T., Reiter, M., Bischof, H.: Appearance models based on kernel canonical correlation analysis. *Pattern Recognition* 36, 1961–1971 (2003)
15. Hotelling, H.: Relations between two sets of variates. *Biometrika* 28, 321–377 (1936)
16. Borga, M.: Learning multidimensional signal processing. PhD thesis (Department of Electrical Engineering, Linkoping University)

# Contour Based Shape Retrieval

Levente Kovács

Distributed Events Analysis Research Group

Computer and Automation Research Institute, Hungarian Academy of Sciences  
Kende u. 13-17, 1111 Budapest, Hungary

[levente.kovacs@sztaki.hu](mailto:levente.kovacs@sztaki.hu)

<http://web.eee.sztaki.hu>

**Abstract.** This paper presents a contour-based indexing and retrieval method for content-based image/video retrieval applications. It is based on extracting closed contours, smoothing the contour, indexing with a variation of BK-trees, and using a turning function metric for data comparison. The method is very lightweight, fast and robust - the goal being retaining close to realtime speeds for real applicability. We provide evaluation data showing that the method performs well and fast, and is suitable for inclusion into content based retrieval systems as a descriptor for recognition of in-frame objects and shapes.

## 1 Introduction

Content-based media retrieval deals with a broad area of indexing and searching among audio-visual data. This includes example-based retrievals of image and video data. The basis of high level, semantic searches is always a combination of low level features and descriptors, each of which concentrates on a representative local or global feature, on which higher level semantics can be built. One of such lower level descriptors is the representation of extracted object contours and/or shapes, aiding the indexing and retrieval of similar shapes and objects. Query formulations based on shape descriptors enable users to find specific objects, and also the possibility of shape-based classification, e.g. target recognition for tracking, signaling the appearance of certain objects on a scene, and so on.

In this paper we will present a very fast, lightweight, yet robust and easy-to-modularize shape retrieval approach, based on contour extraction, denoising/smoothing the contour-lines, indexing into a quickly searchable tree structure, and retrieval results showing high precision. The goal of the presented method is to be used as an additional low-level descriptor in a high level content-based retrieval engine [1]. The foremost purpose is to be able to automatically categorize newly extracted shapes into pre-existing classes, enabling the automatic recognition of scene elements, thus aiding the higher level understanding of image/scene contents.

Traditionally, contours/shape descriptors have been extracted and compared with a series of methods, including Hidden Markov Models [2][3], Scale Invariant Feature points (SIFT) [4], tangent/turning functions [5][6], curvature maps

[7], shock graphs [8], Fourier descriptors [9][10], and so on. They all have their benefits and drawbacks, regarding computational complexity, precision capabilities, implementation issues, robustness and scalability. See [11] for one of many comparisons performed between some of these methods.

The works in [2][3] curvature features of contour points are extracted and used to build Hidden Markov Models, and some weighted likelihood discriminator function is used to minimize classification errors between the different models, and good results (64–100% recognition rates) are presented achieved in the case of plane shape classification. In [7] curvature maps are used to compare 3D contours/shapes. In [9][10] Fourier descriptors are used, as probably the most traditional way of representing contour curves, for comparison purposes. In [10] Support Vector Machine based classification and self-organizing maps are both used for contour classification, which results in a robust, yet highly complex and computationally expensive method, resulting in recognition (precision) rates above 60%, and above 80% in most cases. Turning/tangent function based contour description and comparison [5][6] are also used, mostly for comparison purposes, for it being lightweight, fairly easy to implement, yet research seems to concentrate to continuously depart from it. These methods work by representing the contours as a function of the local directional angle of the contour points along the whole object, and comparing two such representations with different methods, most of them being also rotation invariant.

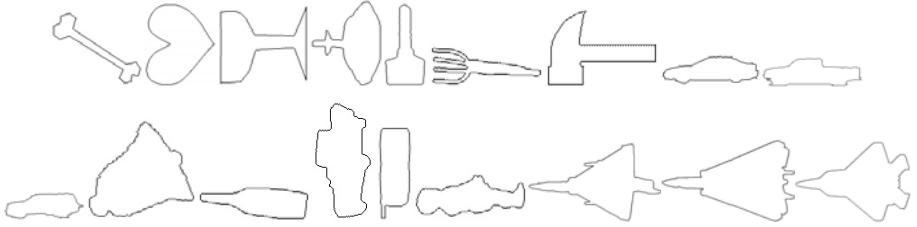
For the purposes of this paper we have chosen to use the turning function based contour representation. The main reason has been the goal of creating a very lightweight implementation of contour matching and classification, which is also robust, and easy to implement and modularize.

## 2 The Shape Retrieval

The dataset we use in this paper are those from [2] with minor additions. The set contains 590 individual shapes, grouped into 23 classes, each class containing versions (rotated, modified but similar, noisier, etc.) of the same shape. Some examples for the different classes are shown on Fig. 1. The data set is a collection of Matlab mat files, which we transformed into different size binary images, each containing the shape in the middle of the image, having a 5 pixel border around the shape. The reason why the method uses images instead of the available point lists is that the implemented method is intended to be a general contour descriptor and indexer module as a part of a content based retrieval engine (the one also used in [1]), thus being able to work with any type of input as long as it is a binary image containing a freeform, but closed contour, object shape.

Using the series of images generated, the steps of the retrieval the following:

1. Load image, extract contour points, drop reoccurring coordinates.
2. Build the index tree: the original point list is the stored data, but a denoised version is generated for the turning function based comparison, which is the heart of the indexing.
3. Classification/retrieval step: a query shape is the input, the results are the most similar shapes from the indexed data.



**Fig. 1.** Examples of shape classes

## 2.1 Contour Extraction

For contour representation, a list of coordinates is extracted from the input shape images. The point list is extracted by a bug follower contour tracking method with backtracking, able to start from any random object point and follow a contour of the object shape, having any free form.

The resulting point list contains hundreds to thousands of neighboring coordinate pairs, and sometimes points can be repeated in the case of certain small bulges present on the contour. Thus, a post-processing step eliminates coordinate re-occurrences. The resulting list will be finally stored.

During the indexing and retrieval process, a turning function based distance metric will be used - described below in section 2.2 - for which we also generate a denoised contour version, thus improving the indexing and retrieval times, and the also the recognition rate (detailed below in sections 2.3-5).

The smoothing procedure is also a lightweight approach. Let

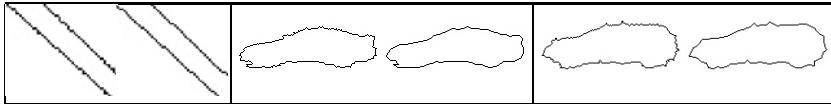
$$C = C_i = (x_i, y_i) | i = \overline{0, n} \quad (1)$$

be the contour with  $C_i$  as the contour points,  $n$  the number of point coordinates, starting from a random contour position  $(x_0, y_0)$ , and ending with a neighbor of the start position. Then, the smoothed contour will be

$$C' = C'_i = (x'_i, y'_i) | i = \overline{0, m}, m < n \quad (2)$$

$$(x_i, y_i) \in C' \text{ if } d((x_{i-k}, y_{i-k}), (x_{i+l}, y_{i+l})) < \varepsilon \quad (3)$$

where  $\varepsilon$  is a threshold (generally between 2.8 and 4.24 and  $k \geq 2, l \geq 2$ ). This is basically a very primitive version of an outlier detector, working well enough for most practical purposes (see e.g. Fig. 2), yet remaining lightweight and very fast. The resulting contours are only used in the comparison distance metric in the indexing and retrieval phase, the points of the original contour are not changed. This simple smoothing does not disturb the features of the contour, but results in considerable improvement in recognition rates and indexing/retrieval times.



**Fig. 2.** For each pair: original contour (left). smoothed (right).

## 2.2 Indexing

The indexing step of the presented method takes as input the contours extracted in the previous section, and produces a serialization of an index tree structure, which will be used in the retrieval step.

The trees we use are customized BK-trees [12], which we will call BK\*-trees. Traditionally BK-trees have been used for string matching algorithms. Essentially they are representations of point distributions in discrete metric spaces. That is, if we have feature points with an associated distance metric, then we can populate a BK\*-tree with these points in the following way:

1. Pick one of the points as the root node,  $R$ .
2. Each node will have a constant number of  $M$  child nodes.
3. A point  $P_j$  will be placed into the child node  $N_i$  ( $i = 0 \dots M - 1$ ), if

$$i \cdot \frac{d}{M} < d(P_i, P_j) < (i + 1) \cdot \frac{d}{M} \quad (4)$$

where  $d$  is the maximum distance that two points can have (respective the associated metric) and  $P_j$  is  $P_i$ 's parent node. Thus, a node will contain a point if its distance from the parent falls into the interval specified above; each node representing a difference interval  $[i \cdot d/M; (i + 1) \cdot d/M]$ .

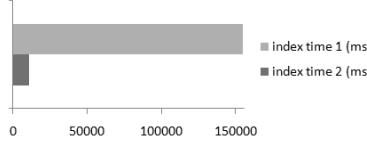
4. Continue recursively until there are no more points left to insert.

As it is, this structure can be used to build quickly searchable index trees for any descriptor which has a metric. We also use this indexing structure for content-based video retrieval (in the system also being part of [1]), as the base structure of indexed content descriptors, where multidimensional queries can be performed using the BK\*-tree indexes of different feature descriptors.

The performance of the indexer: it takes 10.8 seconds to index the 590 shape database we use for the purposes of this paper, on a 2.4GHz Core2 CPU core.

We compared the indexing of the raw contour point sets (in which only the reoccurring coordinate points have been eliminated) with the indexing where the smoothed contour versions were used as the input of the comparison function. Fig. 3 shows the difference, where we obtained a decrease of 15 times in the running time of the indexer, if the smoothing step was used. The effects of the smoothing on the recognition rates are detailed in section 2.3 and 2.4.

The distance metric of the indexing - also used in querying the index structure in the retrieval phase - is a comparison of the results of turning function over the shapes. The output of the turning function is a 2D function representing the directions of the shape points over its contour positions.



**Fig. 3.** Using the smoothed contour representations as the base of comparison increases the indexer's speed by as much as 15 times on average

The turning function  $\theta(s)$  is a standard way of representing a polygon. It measures the angle of tangent as a function of arc length  $s$ . The most important feature of this representation is that it is invariant for translation and scaling, but it is unstable when noise is present - this is why we implemented the smoothing step presented above. We measure the distance between two turning function representations with the following formula:

$$D(P_1, P_2) = \min_t \left[ \int |\theta_1(s+t) - \theta_2(s) + \theta| ds \right] \quad (5)$$

where  $\theta$  is a translation parameter, which makes the distance metric rotation invariant, by comparing the shifted versions of the first function to the second function. Thus the distance becomes translation, scaling and rotation invariant.

### 2.3 Retrieval and Classification

Given the index tree generated with the above described method, the retrieval of shapes can be performed. The query of the retrieval is a shape image similar in contents of the ones indexed, that is a black and white image containing the contour of the query object.

Given a content-based query ( $Q$ ), the index tree is searched for similar entries:

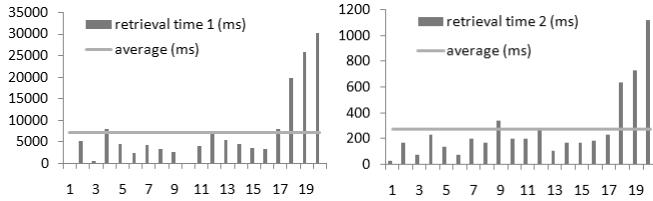
1. If  $d_0 = d(Q, R) < t$  ( $t$  is user-adjustable), the root  $R$  element is a result.
2. Let  $N_i$  be the children of node  $P_j$  ( $P_0 = R$ ), and let  $d_k = d(N_k, P_j)$  for the child  $N_k$  where

$$\begin{cases} k \cdot \frac{d}{M} \in [d_{j-1} - t, d_{j-1} + t] \\ (k+1) \cdot \frac{d}{M} \in [d_{j-1} - t, d_{j-1} + t] \\ k \cdot \frac{d}{M} \leq d_{j-1} - t \text{ and } (k+1) \cdot \frac{d}{M} \geq d_{j-1} + t \end{cases} \quad \text{or} \quad (6)$$

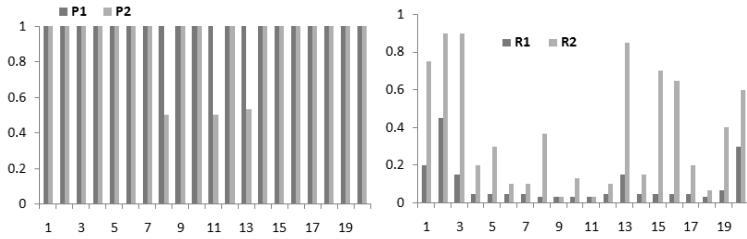
then if  $d_k < t$  the element from child  $N_k$  is a result.

3. Repeat step 2 recursively until the whole tree is visited.
4. Sort all the results in the increasing order of their  $d$  distances and return the ordered result list.

The performance of the retrieval is detailed in sections 2.4-5. Fig. 3 showed that the indexing speed is greatly increased by the smoothing, but that in itself brings nothing, unless the retrieval performance is conserved. We will show that not only does the smoothing conserve the recognition rate, but it also improves the retrieval speed more than 20 times over. Fig. 4 shows retrieval times without and



**Fig. 4.** Comparing retrieval times without (left) and with (right) the smoothing step in the distance metric



**Fig. 5.** Comparing recognition rates without (P1, R1) and with (P2, R2) the smoothing step in the distance metric. Left: precision values. Right: recall values.

with smoothing, containing times of 20 different queries (each query was picked from a different class), and their averages. The average run time for the raw retrieval over the 590 dataset is 7230 ms, which - by including the smoothing step - drops down to 275 ms.

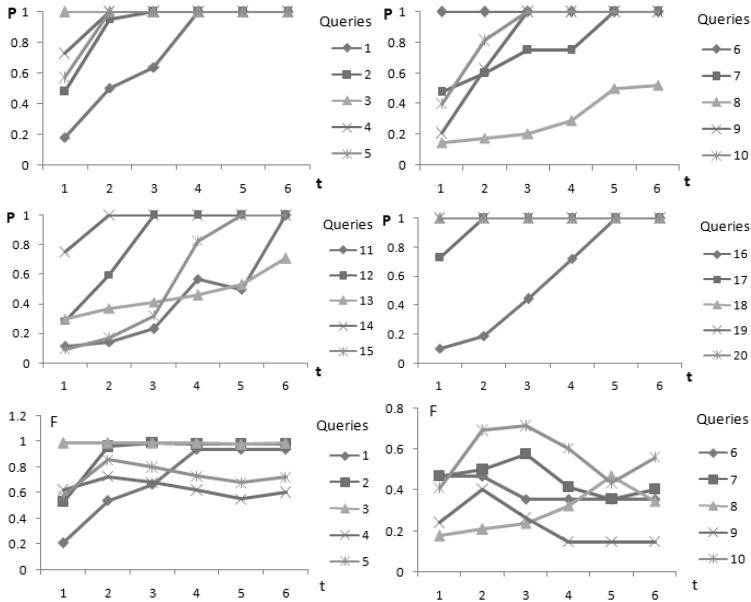
We also compared retrieval performance changes caused by the inclusion of the smoothing step. Fig. 5 shows precision and recall values for the two cases. As the experiments show, comparison with smoothing resulted in almost similar precision values, while it caused considerable improvement in recall rates. Precision and recall are defined by

$$P = \frac{\text{nr. of relevant retrieved items}}{\text{nr. retrieved items}}; \quad R = \frac{\text{nr. of relevant retrieved items}}{\text{nr. all relevant items}} \quad (7)$$

## 2.4 Evaluation on Synthetic Data

In this section we present some retrieval evaluation details, concerning recognition rate evaluation, and presenting precision-recall values for a series of retrieval experiments. The retrievals were performed on the above presented datapool of 590 shapes belonging to 23 classes. We used 20 different query shapes, each belonging to a different class, and for each query we ran 6 different retrievals with differing retrieval thresholds (influencing the number of retrieved results), leading to a total of 120 retrieval runs.

The shape retrieval literature - e.g. [29] - contains extensive recognition rate evaluation data, generally varying between 60-100% precision rate. Our



**Fig. 6.** Precision values of retrievals for 20 query shapes in 6 different retrieval threshold cases (top and middle row) and F0.5 score values for the first 10 queries (bottom row) showing combined P-R values

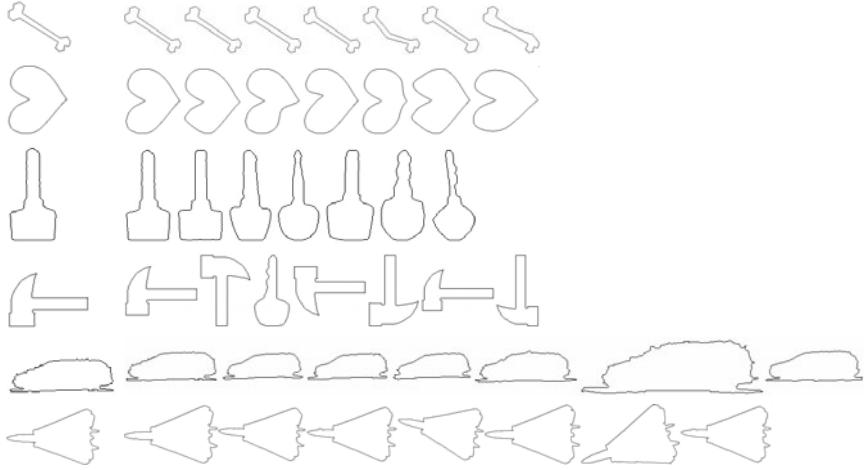
experiments showed that the retrieval method presented in this paper can produce 100% recognition rates in 90% of retrievals (to be precise: the method can produce these results with the presently used shape database).

Fig. 6 presents recognition rate performance data (precision and F score values) over the 20 queries with 6 retrieval thresholds (separated into 4 graphs for easier viewing). The graphs show that with the appropriate retrieval thresholds we can achieve 100% precision in 90% of the cases. Table 3a presents actual numerical values for the recognition rates for 20 queries, while Table 3b shows average recognition rate values from other works for a quick comparison.

Fig. 7 presents results for 7 sample queries (from the 20 query shapes) and the first 7 retrieved results. As this example also shows, even when recall values become low at certain retrievals, the recognition always remains stable, and classification of a shape into the different shape classes remains high in all cases.

## 2.5 Evaluation on Real Data

To evaluate the performance of the method in real circumstances, we performed recognition tests on a dataset of real plane shapes (over 8000 shapes in 24 classes), automatically segmented from real video footage - no synthetic data. Each plane type is treated as a separate class, and during recognition a retrieval only counts as positive when the correct plane type is recognized. Given the index tree generated with the described method, the retrieval of shapes can



**Fig. 7.** Example retrieval results for 6 queries (left column) and first 7 retrieved results (on the right)

**Table 1.** The used shape classes, the number of shape variations of the classes and a sample mask from them

class id.	1	2	3	4	5	6	7	8	9	10	11	12
shapes in class	221	207	665	836	544	597	300	572	87	211	79	184
sample												
class id.	13	14	15	16	17	18	19	20	21	22	23	24
shapes in class	104	224	445	124	11	340	638	392	90	74	285	501
sample												

**Table 2.** Recognition rate data for 120 queries, belonging to 7 different classes

test nr.	1	2	3	4	5	6	7	avg.
nr. of queries	12	7	10	10	40	25	16	120
recognition rate	1	0.86	0.89	1	0.93	0.65	0.59	0.85

be performed. The query of the retrieval is a shape image similar in content to the ones already indexed. Table 1 shows how many different shape variations each class had in itself.

Testing of the retrieval performance is done as follows: a video of a plane type that has a class in the dataset is taken as a test video, and for each frame of the video a query is performed against the indexed shape database. If plane type  $A$  was always recognized as belonging to class  $B$ , then according to the retrievals source  $A$  is 100% of class  $B$ . Table 2 contains some examples for average recognition rates of the retrieval (1 means 100%). Generally, the performance will

**Table 3.** (a) Recognition data for the 20 query shapes, belonging to different shape classes: set1 - Bicego et al. [3], set2 - cars [2], set3 - MPEG-7 shapes [2], set4 - plane shapes [2]. (b) Average recognition rates from other works.

		(a)																				
query nr.	sets	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	
best recog.		set1					set2					set3					set4					
avg. recog.		0.7	0.9	1	0.95	0.9	1	0.8	0.3	0.8	0.9	0.4	0.8	0.5	0.96	0.6	0.6	0.95	1	1	1	
(b)																						
[3] - avg. precisions for:						[10] - avg. precisions for:						[9] - avg. precisions for diff. features:										
different noise levels:						0.92	99 shapes					0.9	MI	0.69								
30 degree tilt for 9 slant levels:						0.8	216 shapes					0.97	FD	0.57								
60 degree tilt for 9 slant levels:						0.75	1045 shapes					0.84	UNL	0.71								
90 degree tilt for 9 slant levels:						0.69	24 shape classes					0.9	UNL-F	0.98								

usually get higher if the analysis is performed on a longer feed/image sequence (thus more queries are performed) since the inter-class variance of different shape types is very high, and also, a high number of plane shapes are very similar from a profile perspective (which can lower recognition rates).

### 3 Conclusions

In this paper we have presented an implementation of a shape indexing and retrieval method, with the main goals of being lightweight and fast, and with high precision. The final approach serves as one of the low level descriptors in content-based retrieval, which higher level semantic interpretations build upon. Future work will concentrate on applications for realtime recognition tasks, e.g. object recognition and tracking, identification tasks for unmanned aerial vehicles.

**Acknowledgments.** This work has been partially supported by the MEDUSA project of the EDA.

### References

1. Szlávík, Z., Kovács, L., Havasi, L., Benedek, C., Petrás, I., Utasi, A., Licsár, A., Czúni, L., Szirányi, T.: Behavior and event detection for annotation and surveillance. In: International Workshop on Content-Based Multimedia Indexing, pp. 117–124 (2008)
2. Thakoor, N., Gao, J., Jung, S.: Hidden markov model-based weighted likelihood discriminant for 2d shape classification. IEEE Tr. on Image Processing 16, 2707–2719 (2007)

3. Bicego, M., Murino, V.: Investigating hidden markov models' capabilities in 2d shape classification. *IEEE Tr. on Pattern Recognition and Machine Intelligence* 26, 281–286 (2004)
4. Lowe, D.G.: Object recognition from local scale-invariant features. In: *ICCV*, pp. 1150–1157 (1999)
5. Scassellati, B., Alexopoulos, S., Flickner, M.: Retrieving images by 2d shape: a comparison of computation methods with perceptual judgments. In: *SPIE Storage and Retrieval for Image and Video Databases II*, vol. 2185, pp. 2–14 (1994)
6. Latecki, L.J., Lakamper, R.: Application of planar shape comparison to object retrieval in image databases. *Pattern Recognition* 35, 15–29 (2002)
7. Gatzke, T., Garland, M.: Curvature maps for local shape comparison. In: *Shape Modeling and Applications*, pp. 244–253 (2005)
8. Sebastian, T., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing their shock graphs, vol. 26, pp. 550–571 (2004)
9. Frejlichowski, D.: An algorithm for binary contour objects representation and recognition. In: Campilho, A., Kamel, M.S. (eds.) *ICCIAR 2008. LNCS*, vol. 5112, pp. 537–546. Springer, Heidelberg (2008)
10. Wong, W.T., Shih, F.Y., Liu, J.: Shape-based image retrieval using support vector machines, fourier descriptors and self-organizing maps. *Intl. Journal of Information Sciences* 177, 1878–1891 (2007)
11. Rosenhahn, B., Brox, T., Cremers, D., Seidel, H.: A comparison of shape matching methods for contour based pose estimation. In: Reulke, R., Eckardt, U., Flach, B., Knauer, U., Polthier, K. (eds.) *IWCIA 2006. LNCS*, vol. 4040, pp. 263–276. Springer, Heidelberg (2006)
12. Burkhard, W., Keller, R.: Some approaches to best-match file searching. *Communications of the ACM* 16, 230–236 (1973)

# Illumination Normalization for Robust Face Recognition Using Discrete Wavelet Transform

Amnart Petpon and Sanun Srisuk

Department of Computer Engineering, Mahanakorn University of Technology  
51 Cheum-Sampan Rd., Nong Chok, Bangkok, Thailand 10530  
ta\_tee473@hotmail.com, sanun@mut.ac.th

**Abstract.** In this paper, we introduce an illumination normalization approach within frequency domain by utilizing Discrete Wavelet Transform (DWT) as a transformation function in order to suppress illumination variations and simultaneously amplify facial feature such as eyeball, eyebrow, nose, and mouth. The basic ideas are: 1) transform a face image from spatial domain into frequency domain and then obtain two major components, approximate coefficient (Low frequency) and detail coefficient (High frequency) separately 2) remove total variation in an image by adopting Total Variation Quotient Image (TVQI) or Logarithmic Total Variation (LTV) 3) amplify facial features, which are the significant key for face classification, by adopting Gaussian derivatives and Morphological operators respectively. The efficiency of our proposed approach is evaluated based on a public face database, Yale Face Database B, and its extend version, Extend Yale Face Database B. Our experimental results are demonstrated that the proposed approach archives high recognition rate even though only single image per person was used as the training set.

## 1 Introduction

The performance of face recognition is greatly degraded due to the intensity variation even though the original image and the test image belong to the same class. The variations between the images of the same face/class due to illumination and viewing direction are almost always larger than image variations due to the change in face identity. Numerous face recognition methods have been proposed for solving such major problems such as eigenface or Principal Component Analysis (PCA) [1], fisherface or Linear Discriminant Analysis (LDA) [2], Elastic Bunch Graph Matching (EBGM) [3] and Support Vector Machine (SVM) [4]. In order to recognize a face image more efficiently, a preprocessing technique is required for producing a normalized face image by which its image variations are reduced. Having surveyed the research literature, most objectives to deal with illumination variation are categorized: 1) extracting illumination invariant or illumination free [5] [6] [7]. 2) extracting face / image representation [8] [9]. 3) compensating dark region in the face image [10]. 4) enhancing global image contrast (e.g. histogram equalization [11] and wavelet based method [12]).

Since the accuracy of face recognition is completely dependent on the potential total variations in a face image, especially illumination variation. To attack such variation we then design an illumination normalization approach by utilizing Discrete Wavelet

Transform (DWT) as a transformation function in order to obtain illumination insensitive facial feature and simultaneously amplify facial feature such as eyeball, eyebrow, nose, and mouth.

The rest of this paper is organized as follows. In section 2, we give a brief overview of the reflectance model then the related methods, which are used in the experiment, are described in details. Finally, we introduce our proposed approach for face recognition under varying illumination. Experimental results on Extend Yale face database B will be given in section 3. Section 4 gives a conclusion.

## 2 Methodologies

In this section, we give an overview of reflectance model for image representation as well as a powerful illumination normalization model and also a brief of Discrete Wavelet Transform (DWT). We then present our proposed approach for illumination normalization in frequency domain. The approach is used to extract illumination invariant from face image under which the lighting condition is uncontrolled.

### 2.1 Illumination Normalization

A face may be classified as 3D objects that have the same shape but differ in the surface albedo function. In general, an image  $f(x, y)$  is characterized by two components: the amount of source illumination incident on the scene being viewed and the amount of illumination reflected by the objects in the scene. In both Retinex and Lambertian theories, they are based on the physical imaging model by which the intensity can be represented as the product of reflectance and illumination. In Retinex theory, the image  $f(x, y)$  is formed as:

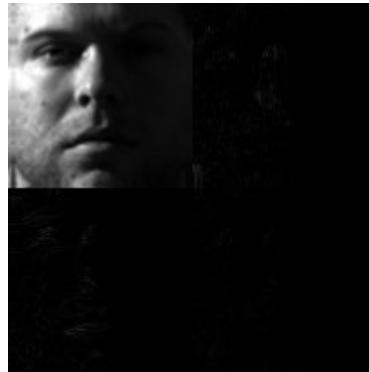
$$f(x, y) = r(x, y)i(x, y), \quad (1)$$

where  $r(x, y) \in (0, 1)$  is the reflectance determining by the characteristics of the imaged objects and  $i(x, y) \in (0, \infty)$  is the illumination source. In Lambertian reflectance function, the image can be described by the product of the albedo (texture) and the cosine angle between a point light source and the surface normal

$$f(x, y) = \rho(x, y)n(x, y)^T s, \quad (2)$$

where  $\rho(x, y)$  is the albedo (surface texture) associated with point  $x, y$  in the face object,  $n(x, y)$  is the surface normal (shape) of the face object, and  $s$  is the point light source direction whose magnitude is the light source intensity. In this paper, the surface normal of the object is assumed to be the same for all objects of the class. In fact, the reflectance and albedo are illumination free in quotient image. Final step is to estimate the illumination,  $S(x, y)$  where can be easily computed by weighted Gaussian filter as Self Quotient Image (SQI) does in [5].

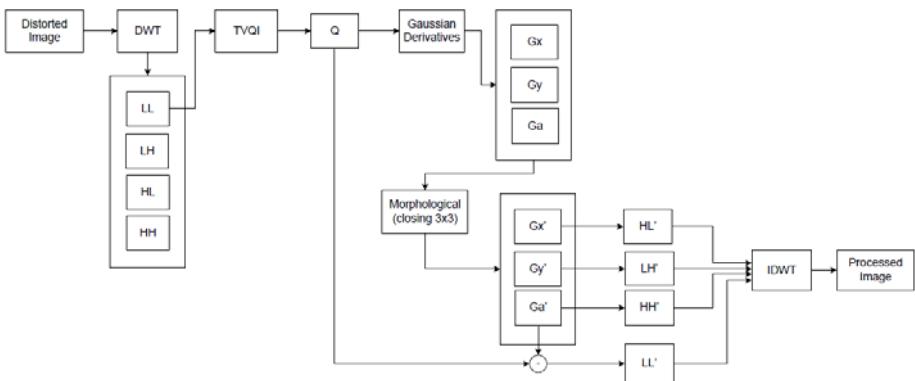
$$Q(x, y) = \frac{f(x, y)}{S(x, y)} \quad (3)$$



**Fig. 1.** Multi-resolution Wavelet Decomposition: Low-Low or LL(lef-top), Low-High or LH(left-bottom), High-Low or HL(right-top), and High-High or HH(right-bottom)

Similarly, Chen et al. has proposed a powerful illumination normalization model, called Total Variation Quotient Image (TVQI) [6], by decomposing a face image  $f$  into large-scale feature  $u$  (e.g. skin, background) and small-scale feature  $v$  (e.g. eyes, noses, mouths, eyebrows) separately, where  $f$ ,  $u$ , and  $v$  are functions of image intensity. Later Logarithmic Total Variation (LTV) [13] has also been proposed, it is almost totally same as TVQI does unless differ in taking logarithm on the input face image. The normalized face image was obtained by dividing the original image with the large-scale feature  $u$

$$\begin{aligned} u &= \min_u \int |\nabla u| + \lambda \|f - u\|_{L^1}, \\ v &= f - u, \\ TVQI &= \frac{f}{u}, \end{aligned} \quad (4)$$



**Fig. 2.** The Proposed Approach

## 2.2 Multi-level Discrete Wavelet Transform

The transformation of a 2D signal like images is just another form of representing the signal. It does not change the information content present in the signal. The Discrete Wavelet Transform (DWT) becomes a popular technique in image/video compression and image denoising [14]. As in the wavelet domain, the small coefficients are more likely due to unnecessary information, noise, and large coefficient due to important information. These small coefficients can be therefore thresholded without affecting the significant features of the image. Moreover DWT has been successfully used in image processing for features extraction [15].

However the DWT is just a sampled version of Continuous Wavelet Transform (CWT) and its computation may consume significant amount of time and resources, depending on the resolution required. The basic idea of multi-level DWT is to decompose the image into an approximate component (Low-Low or LL frequency) and detail components, which can be further divided into Low-High or LH, High-Low or HL, and High-High or HH frequency separately as depicted in Figure 1.

$$LL_i(r, c) = \left[ H_r * [H_c * L_{i-1}]_{\downarrow 2,1} \right]_{\downarrow 1,2} (r, c) \quad (5)$$

$$LH_i(r, c) = \left[ H_r * [G_c * L_{i-1}]_{\downarrow 2,1} \right]_{\downarrow 1,2} (r, c) \quad (6)$$

$$HL_i(r, c) = \left[ G_r * [H_c * L_{i-1}]_{\downarrow 2,1} \right]_{\downarrow 1,2} (r, c) \quad (7)$$

$$HH_i(r, c) = \left[ G_r * [G_c * L_{i-1}]_{\downarrow 2,1} \right]_{\downarrow 1,2} (r, c) \quad (8)$$

where  $i$  is decomposition level,  $*$  denotes the convolution operator,  $\downarrow 2, 1$  sub-sampling along the rows by considering only even row,  $\downarrow 1, 2$  sub-sampling along the columns by considering only even column,  $H$  and  $G$  are a Low pass and High pass filter, respectively. In DWT,  $H$  and  $G$  are called as mother wavelets such as Daubechies, Coiflets, Symlets, and Biorthogonal.

## 2.3 Proposed Approach

In our approach, as depicted in Figure 2, we first utilize DWT in order to decompose a distorted face image, caused by illumination variations, into different four sub-bands LL, LH, HL, and HH separately without sub-sampling. The main reason is due to most of illumination variations is mainly lie in the Low frequency band [10]. The sub-sampling procedure has been neglected so that all image features which might be lost in Wavelet domain caused by the sub-sampling can be retained and used in further processing. We then obtain the quotient image, which is illumination invariant, by computing only LL sub-band with TVQI or LTV and discard the rest of sub-bands. Since the rest of sub-bands are mostly containing high frequency such as edges and some edges caused by some casting shadows or specularities which might degrade the face recognition performance, hence, it can be discarded.

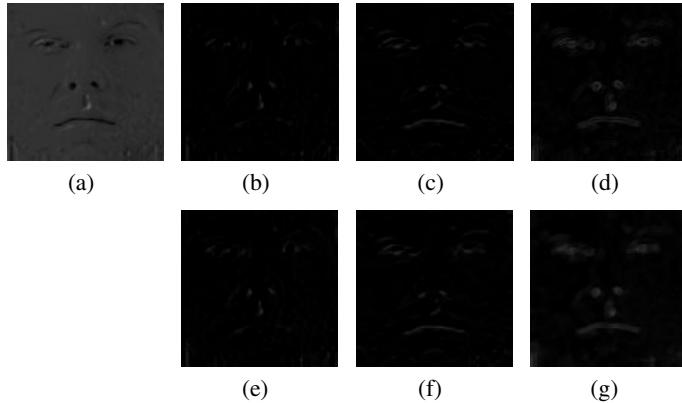
The remaining part of the approach is to construct new sub-bands replacing the discarded one. The quotient image ( $Q$ ) is then computed with 1D Gaussian derivatives,

as equation (10), along with horizontal and vertical image axis. We finally get three images, Gaussian derivative image convoluted along with horizontal axis ( $G_x$ ), Gaussian derivative image convoluted along with vertical axis ( $G_y$ ), and amplitude image ( $G_a = \sqrt{G_x^2 + G_y^2}$ ).

$$G(x, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} e^{\left(\frac{-x^2}{2\sigma^2}\right)}, \quad (9)$$

$$\dot{G}(x, \sigma) = \frac{-x}{\sigma^2} G(x, \sigma). \quad (10)$$

To obtain  $\dot{G}_x$ ,  $\dot{G}_y$ , and  $\dot{G}_a$ . Three Gaussian images  $G_x$ ,  $G_y$ , and  $G_a$  are then computed with Morphological closing operator (dilation and then erosion) by mask size as 3x3 pixels, some sample of processed images are illustrated in Figure 3 in order to fill all holes. This is to amplify facial feature (e.g. eyeball, eyebrow, nose, and mouth).



**Fig. 3.** Processed Images: (a) quotient image, (b)-(d) results of Gaussian Derivatives, and (e)-(g) results of Morphological closing operator

The rest of computation can be mathematically described as equation (14),

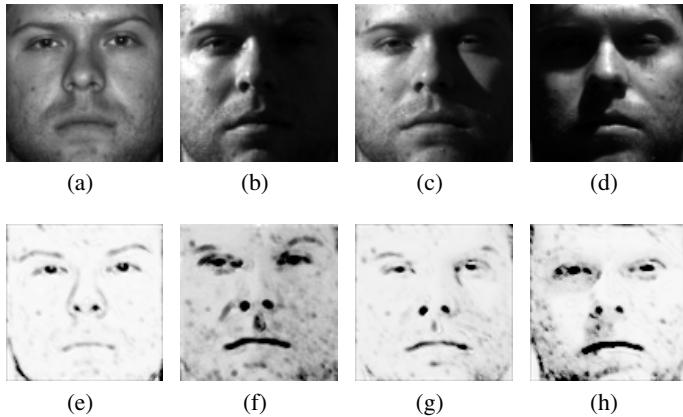
$$L'L = Q - \dot{G}_a, \quad (11)$$

$$L'H = \dot{G}_y, \quad (12)$$

$$H'L = \dot{G}_x, \quad (13)$$

$$H'H = \dot{G}_a. \quad (14)$$

Finally Inverse Discrete Wavelet Transform (IDWT) is used for reconstructing the processed image as shown in Figure 4. It is evident that some casting shadows and specularities on the face image can be greatly removed.



**Fig. 4.** Results of our proposed method: (a) original image with normal illumination variation, (b)-(d) images with extreme illumination variation, and (e)-(h) processed images with our proposed approach

### 3 Experiments

For the testing of our algorithm we evaluated the proposed method on a popular public face database, Yale Face Database B, and its extend version, Extend Yale Face Database B, [16]. The comparative methods are comprised of Histogram Equalization, Self-Quotient Image (SQI) [5], Total Variation Quotient Image (TVQI) [6], Logarithmic Total Variation (LTV) [13], Discrete Cosine Transform (DCT) [17], and Wavelet-based Illumination Normalization (DWT Histogram) [12].

#### 3.1 Data Preparation

The Yale Face Database B contains 5,760 images of 10 subjects under 9 poses x 64 illumination conditions and the Extend Yale Face Database B contains 16,128 images of 28 subjects under 9 poses x 64 illumination conditions. Since we are focusing in the illumination variation, thus only frontal face image for each subject with different 64 illumination was selected, 2,432 images in total from 38 subjects. In our experiment, these two databases will be combined as one single database. The database is regularly divided into 5 subsets according to the angle of the light source directions and the central camera axis as follows:

- Subset 1 ( $0^\circ$  to  $12^\circ$ ) - 7 face images under different illumination conditions, 3 corrupted image was discarded, 263 images in total,
- Subset 2 ( $13^\circ$  to  $25^\circ$ ) - 12 face images under different illumination conditions, 456 images in total,
- Subset 3 ( $26^\circ$  to  $50^\circ$ ) - 12 face images under different illumination conditions, 1 corrupted image was discarded, 455 images in total,

- Subset 4 ( $51^\circ$  to  $77^\circ$ ) - 14 face images under different illumination conditions, 8 corrupted image was discarded, 524 images in total,
- Subset 5 (above  $78^\circ$ ) - 19 face images under different illumination conditions, 15 corrupted image was discarded, 707 images in total.

All face images (subset 1 to 5) were manually rotated, resized and cropped to  $100 \times 100$  pixels with 256 gray levels according to the coordinates of two eyes. They were cropped so that the only face regions are considered. The default settings in our experiments are as follows: the mother wavelet function is Coiflet wavelet of order 1,  $\lambda$  for TVQI/LTV is 0.75 as suggested in [6],  $\sigma$  for Gaussian derivative is 0.5, and mask size for Morphological closing operator is 3x3. Finally, the template matching and the Linear Discriminant Analysis (LDA) were used in classification task.

### 3.2 First Experiment

In the first experiment, all face images from one of five subsets were used as a training set and the rest of subsets were used as a test set respectively. The results are given in Table 1. From the tables, one can observe that the average recognition rate of our proposed method (DWT+TVQI and DWT+LTV) are very high unless DWT+LTV. A reason is that the original LTV firstly takes logarithm on the image which might disturb some facial information.

**Table 1.** Recognition rates (%) when using images of subset 1 as training set

Methods	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5	Average
SQI	100	99.56	98.68	88.74	91.8	95.76
DCT	100	97.81	96.92	90.08	64.78	89.92
TVQI	100	100	96.48	91.41	93.21	96.22
LTV	100	100	99.78	82.25	94.06	95.22
DWT(Histogram)	100	99.56	77.36	20.23	20.93	63.62
DWT+TVQI	100	100	99.78	96.18	97.17	98.63
DWT+LTV	100	100	92.53	85.11	86	92.73

**Table 2.** Recognition rates (%) when using images of subset 2 as training set

Methods	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5	Average
SQI	96.58	100	95.6	94.85	90.66	95.54
DCT	96.58	100	90.11	91.22	68.74	89.33
TVQI	99.62	100	97.58	95.42	89.53	96.43
LTV	100	100	99.78	90.46	92.64	96.58
DWT(Histogram)	100	100	94.07	43.32	30.13	73.5
DWT+TVQI	98.86	100	99.78	97.71	96.75	98.62
DWT+LTV	99.24	100	96.26	94.85	92.5	96.57

**Table 3.** Recognition rates (%) when using images of subset 3 as training set

Methods	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5	Average
SQI	94.68	90.35	100	97.14	97.6	95.95
DCT	97.72	94.52	100	92.56	79.63	92.89
TVQI	99.24	97.81	100	96.18	94.06	97.46
LTV	97.72	98.46	100	95.8	97.03	97.8
DWT(Histogram)	92.78	96.05	100	92.56	62.8	88.84
DWT+TVQI	99.62	98.9	100	97.33	99.15	99
DWT+LTV	97.72	97.81	100	95.04	95.33	97.18

**Table 4.** Recognition rates (%) when using images of subset 4 as training set

Methods	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5	Average
SQI	56.65	79.61	96.04	100	98.87	86.23
DCT	95.44	96.27	96.48	100	92.64	96.17
TVQI	81.75	89.47	96.48	100	99.58	93.46
LTV	57.79	62.94	90.33	100	99.86	82.18
DWT(Histogram)	17.87	46.71	85.71	100	94.2	68.9
DWT+TVQI	98.48	98.9	99.56	100	99.72	99.33
DWT+LTV	83.65	89.47	96.48	100	99.72	93.86

**Table 5.** Recognition rates (%) when using images of subset 5 as training set

Methods	Subset 1	Subset 2	Subset 3	Subset 4	Subset 5	Average
SQI	63.5	70.39	94.29	98.28	100	85.29
DCT	24.33	24.56	55.82	94.27	100	59.8
TVQI	99.24	96.71	98.68	97.14	100	98.35
LTV	77.95	76.97	91.87	98.85	100	89.13
DWT(Histogram)	6.84	13.82	62.86	93.51	100	55.41
DWT+TVQI	98.1	98.03	99.56	99.81	100	99.1
DWT+LTV	80.23	86.18	95.38	99.24	100	92.21

However most of face recognition application usually has a limitation about the insufficient number of the training set. In order to evaluate our proposed method with the limitation, only single image per subject with normal illumination condition were used as a training set and the rest of subset were used as test set. The result were given in Table 6.

**Table 6.** Average Recognition Rates (%) when using a single image per person as training set

Methods	SQI	DCT	TVQI	LTV	DWT(Histogram)	DWT+TVQI	DWT+LTV
Average	87.72	86.37	84.62	83.25	54.24	94.98	82.29

**Table 7.** Average Error Rates (%) when randomly selecting 5 images from subset 1-5 as training set

Methods	SQI	DCT	TVQI	LTV	DWT(Histogram)	DWT+TVQI	DWT+LTV
Average ERR	3.46	9.37	10.96	5.54	19.73	1.44	6.49

As a consequence, our proposed approach can reach very high average recognition rate, 94.98%, even though a single image per person was used as a training set and we can conclude that 1) DWT is suitable with TVQI rather than LTV due to Logarithm function gives high contrast in the image intensity. 2) the number of images in training set is direct proportional to recognition rate. The more images we constructed in training set, the more accuracy we are given.

### 3.3 Second Experiment

In this experiment, a random subset with 5 images (one image from each of subset 1-5) per subject is chosen as training set, and the rest of images as testing set. Since the above training set are randomly selected, we ran the simulation 3 times and average the results over them. The average recognition error rate of various methods are listed in Table 7. From the table, it can be seen that our proposed approach outperformed other methods because the smallest average error rates, 1.44%.

## 4 Conclusions

We have presented illumination normalization approach within frequency domain by utilizing Discrete Wavelet Transform (DWT) for robust face recognition under varying illumination. Our motivation is to first decompose a face image from spatial domain into frequency domain with Low and High pass filter in order to separate illumination variation and facial structure. Then eliminate illumination variation and simultaneously amplify facial structure. We adopt DWT for image decomposition and then adopt TVQI/LTV for eliminating illumination effects at Low-Low frequency sub-band (illumination variation lies at low frequency). The rest of sub-bands (Low-High, High-Low, and High frequencies sub-bands) can be further computed from the processed quotient image by using Gaussian derivatives and Morphological closing operator. Reconstructed image is subsequently obtained by IDWT. The advantages of our approach are only single image per person is required and no need a prior alignment. Comparative experimental results can be obviously seen that our proposed approach outperformed all other methods under extreme varying illumination condition.

## References

1. Turk, M., Pentland, A.: Eigenfaces for recognition. *J. Cognitive Neuroscience* 3, 71–86 (1991)
2. Belhumeur, P.N., Hespanha, J.P., Kriegman, D.J.: Eigenfaces vs. fisherfaces: Recognition using class specific linear projection. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 19, 711–720 (1997)
3. Wiskott, L., Fellous, J.M., Kruger, N., von der Malsburg, C.: Face recognition by elastic bunch graph matching. In: *International Conference on Image Processing*, vol. 1, p. 129 (1997)
4. Guo, G., Li, S.Z., Chan, K.: Face recognition by support vector machines. In: *IEEE International Conference on Automatic Face and Gesture Recognition*, p. 196 (2000)
5. Wang, H., Li, S.Z., Wang, Y.: Face recognition under varying lighting conditions using self quotient image. In: *IEEE International Conference on Automatic Face and Gesture Recognition*, p. 819 (2004)
6. Chen, T., Yin, W., Zhou, X.S., Comaniciu, D., Huang, T.S.: Illumination normalization for face recognition and uneven background correction using total variation based image models. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, pp. 532–539 (2005)
7. Wang, J., Wu, L., He, X., Tian, J.: A new method of illumination invariant face recognition. In: *International Conference on Innovative Computing, Information and Control*, p. 139 (2007)
8. Ojala, T., Pietikäinen, M., Harwood, D.: A comparative study of texture measures with classification based on featured distributions. *Pattern Recognition* 29, 51–59 (1996)
9. Tao, Q., Veldhuis, R.N.J.: Illumination normalization based on simplified local binary patterns for a face verification system. In: *Biometrics Symposium 2007 at The Biometrics Consortium Conference*, Baltimore, Maryland, USA, September 2007, pp. 1–7. IEEE Computational Intelligence Society, Los Alamitos (2007)
10. Choi, S.I., Kim, C., Choi, C.H.: Shadow compensation in 2d images for face recognition. *Pattern Recogn.* 40, 2118–2125 (2007)
11. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*. Addison-Wesley Longman Publishing Co., Inc., Boston (2001)
12. Du, S., Ward, R.K.: Wavelet-based illumination normalization for face recognition. In: *ICIP*, vol. (2), pp. 954–957 (2005)
13. Chen, T., Yin, W., Zhou, X.S., Comaniciu, D., Huang, T.S.: Total variation models for variable lighting face recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 1519–1524 (2006)
14. Chang, S.G., Yu, B., Vetterli, M.: Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing* 9, 1532–1546 (2000)
15. Teoh, A.B.J., Goh, Y.Z., Ong, M.G.K.: Illuminated face normalization technique by using wavelet fusion and local binary patterns. In: *ICARCV*, pp. 422–427. IEEE, Los Alamitos (2008)
16. Georgiades, A.S., Belhumeur, P.N., Kriegman, D.J.: From few to many: Illumination cone models for face recognition under variable lighting and pose. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 643–660 (2001)
17. Chen, W., Er, M.J., Wu, S.: Illumination compensation and normalization for robust face recognition using discrete cosine transform in logarithm domain. *IEEE Transactions on Systems, Man, and Cybernetics, Part B* 36, 458–466 (2006)

# Feature-Based Lung Nodule Classification

Amal Farag<sup>1</sup>, Asem Ali<sup>1</sup>, James Graham<sup>1</sup>,  
Shireen Elhabian<sup>1</sup>, Aly Farag<sup>1</sup>, and Robert Falk<sup>2</sup>

<sup>1</sup> Computer Vision and Image Processing Laboratory  
University of Louisville, Louisville, KY

<sup>2</sup> Medical Imaging Division, Jewish Hospital, Louisville, KY  
aafara02@louisville.edu  
www.cvip.uofl.edu

**Abstract.** Model-based detection and classification of nodules are two major steps in CAD systems design and evaluation. This paper examines feature-based nodule description for the purpose of classification in low dose CT scanning. After candidate nodules are detected, a process of classification of these nodules into types is needed. The SURF and the LBP descriptors are used to generate the features that describe the texture of common lung nodules. These features were optimized and the resultant set was used for classification of lung nodules into four categories: juxta-pleural, well-circumscribed, vascularized and pleural-tail, based on the extracted information. Experimental results illustrate the efficiency of using multi-resolution feature descriptors, such as the SURF and LBP algorithms, in lung nodule classification.

**Keywords:** Lung nodule classification, LDCT scans, SURF, multi-resolution LBP.

## 1 Introduction

In the past two decades numerous screening studies in the US, Europe and Japan have been conducted for studying the enhancements of early detection of lung cancer using CT vs. X-ray, and the correlation of early detection to enhancement in lung cancer related mortality. Globally, lung cancer remains the most common malignancy with an estimated 1.5 million newly diagnosed cases in 2007 and 1.35 million deaths occurring that same year. Of the 1.35 million deaths 975,000 cases were men and 376,000 cases were female. The highest recorded 5-year patient survival rates of 14% are observed in the United States while the 5 year survival rate is 8% in Europe [1]. CAD systems have been sought for facilitating large-scale screening and as helping tools for radiologists.

A CAD system involves image filtering, isolation of lung tissues by segmentation, nodule detection and classification (e.g., [2][3]). We have embarked on examination of feature-based algorithms, commonly used in the computer vision literature, as an attempt to find discriminatory features that may be used to classify detected nodule candidates into categories; this will enhance the specificity of the nodule detection process and will be a step forward towards image-based tissue classification. In [4] we used an adaptation of the Daugman Iris Recognition Algorithm and the SIFT

algorithm (e.g., [5]) to generate features for the classification of detected nodules. In this paper we investigate two non-linear approaches to extract texture information from lung nodules to automatically classify each nodule into one of four pre-defined categories identified by [6]. The first is the Speeded-Up Robust Features (SURF) algorithm and the second is the Local Binary pattern (LBP) algorithm. The use of these approaches for lung nodule feature extraction will be described in details.

The closest related works to our application are the following: local texture analysis was used in [7] to identify and classify lung abnormalities such as tuberculosis. Instead of a binary normal/abnormal decision when classifying feature vectors, a weighted voting among the k-NNs was used to compute a probability measure that a region is abnormal. Keramidas et al. in [8] used the Local Binary Pattern for the classification step to extract feature vectors from the area between the thyroid boundaries to then detect thyroid nodules in ultrasound images. Local-Binary pattern has been used for tracheal texture analysis in CT images to help in the classification of Ground Glass nodules as seen in [9]. These features in [8] and [9] are extracted from a cubic sub-volume of interest of size  $7 \times 7 \times 7$  voxels. LBP is used as one of these feature descriptors.

A nodule is defined as a small mass or lump of irregular or rounded shape which is ambiguous when trying to apply the definition to the computer vision and machine learning fields. For example, according to Samala et al. [10], nine feature descriptors that define the nodule characteristics used by radiologists are: 1. internal structure; 2. calcification; 3. subtlety; 4. malignancy; 5. texture; 6. speculation; 7. lobulation; 8. margin and 9. sphericity.

In our work, we allow  $I(x)$  to represent a CT slice where  $x = \{(x, y) : 1 \leq x \leq N_x, 1 \leq y \leq N_y\}$  is a finite spatial grid supporting the slice and  $x_0 = (x_0, y_0)$  is the centroid of a detected nodule region, using this information we can formulate the assignment of various nodule types. The main goal of our framework is to assign a nodule type “c” to a given nodule region using texture-based descriptor,  $T(x_0)$ , where  $c \in \{J, W, V, P\}$  and corresponds to juxta, well-circumscribed, vascular and pleural-trail respectively. Two main stages are involved: first, the detection of potential nodules for the given CT slice(s), and secondly, the building of a nodule descriptor for each nodule type classification/assignment. In this paper we focus on the second stage.

This paper is organized as follows: section 2 describes the feature descriptor algorithms used in the classification analysis, section 3 discusses performance evaluation and section 4 concludes the paper.

## 2 Feature Descriptors

The success of object description hinges on two main conditions: distinction and invariance. The methodology needs to be robust to accommodate for variations in imaging conditions and at the same time produce a distinctive characterization of the desired object. Within the four nodule categories we recognize inter-variations in shape and texture among the Juxta-Pleural, Pleural-Tail, Vascularized and Well-Circumscribed nodules.

## 2.1 Speeded-Up Robust Features (SURF)

The Speeded-Up Robust Features (SURF) algorithm (e.g., [13][14]) may be advantageous over the SIFT algorithm and the Harris-Laplace feature detector [12]. This algorithm was implemented to improve execution time without compromising feature detection efficiency. The detector is based on the Hessian matrix and relies on integral images to reduce computation time. The descriptor is a distribution of Haar-wavelet responses within the neighborhood of interest. The Hessian matrix is defined as:

$$H(\mathbf{x}, \sigma) = \begin{bmatrix} L_{xx}(\mathbf{x}, \sigma) & L_{xy}(\mathbf{x}, \sigma) \\ L_{xy}(\mathbf{x}, \sigma) & L_{yy}(\mathbf{x}, \sigma) \end{bmatrix}$$

where  $\mathbf{x} = (x, y)$  represents a point in the image  $I$ , and  $\sigma$  is a scalar value.  $L_{xx}(\mathbf{x}, \sigma)$  is the convolution of the second order Gaussian derivative with the image in point  $\mathbf{x}$ . The SURF descriptor consists of several steps. A square region is constructed around the interest point and oriented either in a rotation invariant method, where the Haar-wavelet response in the  $x$  – and  $y$ – directions are computed and weighted with a Gaussian centered at the interest point, or a non rotation invariant method. The wavelet responses in both directions are then summed-up over each sub-region. The total number of descriptors for each point is 64.

The nodules are the input images, in our case, and the region of interest is mainly the texture information concentrated around an area where texture information is not sparsely found since the spatial support of lung nodules are relatively small in size. The “blob-response threshold” was reduced to 240; the number of octaves desired is only 1; and the step-response is 1 instead of 2.

## 2.2 Multi-resolution Local-Binary Pattern (LBP)

The LBP operator, first introduced in [15], is a power texture descriptor. The original operator labels the pixels of an image by thresholding the  $3 \times 3$  neighborhood of each pixel with the center value and considering the result as a binary number. At a given pixel position  $(x_c, y_c)$ , the decimal form of the resulting 8-bit word is

$$LBP(x_c, y_c) = \sum_{i=0}^7 s(I_i - I_c)2^i,$$

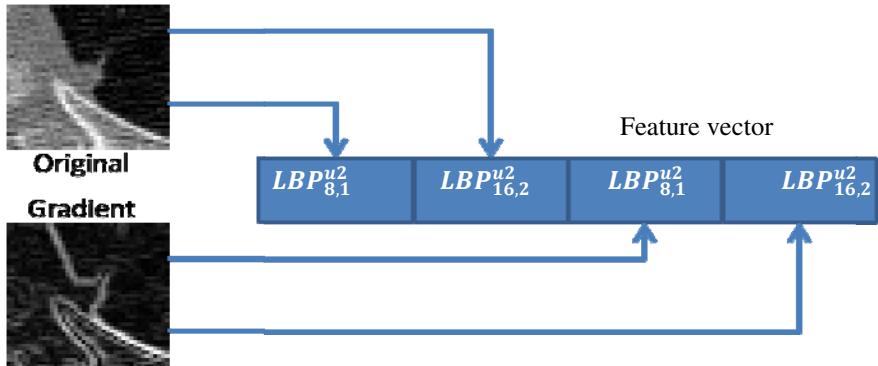
where,  $I_c$  corresponds to the center pixel  $(x_c, y_c)$ ,  $I_i$  to gray level values of the 8 surrounding pixels, and function  $s(o)$  is a unit-step function. The operator is invariant to monotonic changes in grayscale and can resist illumination variations as long as the absolute gray-level value differences are not badly affected. In [16] the LBP operator was extended to a circular neighborhood of different radius size to overcome the limitation of the small original  $3 \times 3$  neighborhood size failing to capture large-scale structures. Each instance is denoted as  $(P, R)$ , where  $P$  refers to the equally spaced pixels on a circle of radius  $R$ . The next extension is the use of uniform patterns [16]. An LBP pattern is considered uniform if it contains at most two bitwise transitions from 0 to 1 and vice-versa, when the binary string is circular. The reason for using uniform patterns is that they contain most of the texture

information and mainly represent texture primitives. The notation  $LBP_{PR}^{u2}$  used in this paper refers to the extended LBP operator in a  $(P, R)$  neighborhood, with only uniform patterns considered. We use the LBP to generate a feature vector which describes the nodule region of interest in a LDCT slice. To generate a feature vector, we use three different scenarios: 1. LBP of the original nodule images, 2. LBP of the gradient image of the nodule or 3. LBP of both the original and gradient nodule images (see Fig. 1). Where, the gradient magnitude image is generated using Sobel filters ( $h_x$  and  $h_y$ ), i.e.

$$|\nabla I| = \sqrt{(h_x \otimes I)^2 + (h_y \otimes I)^2}.$$

A similarity measure is then used to classify these nodules to one of the four classes: juxta, well-circumscribed, pleural tail and vascularized. Principle component analysis (PCA) [11] and linear discriminant analysis (LDA) [18] are used to project the extracted LBP descriptors to a low-dimensional subspace where noise is filtered out.

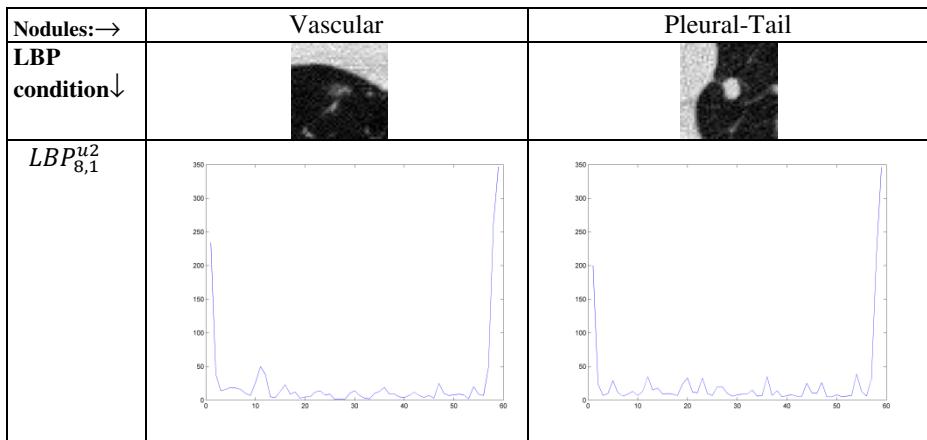
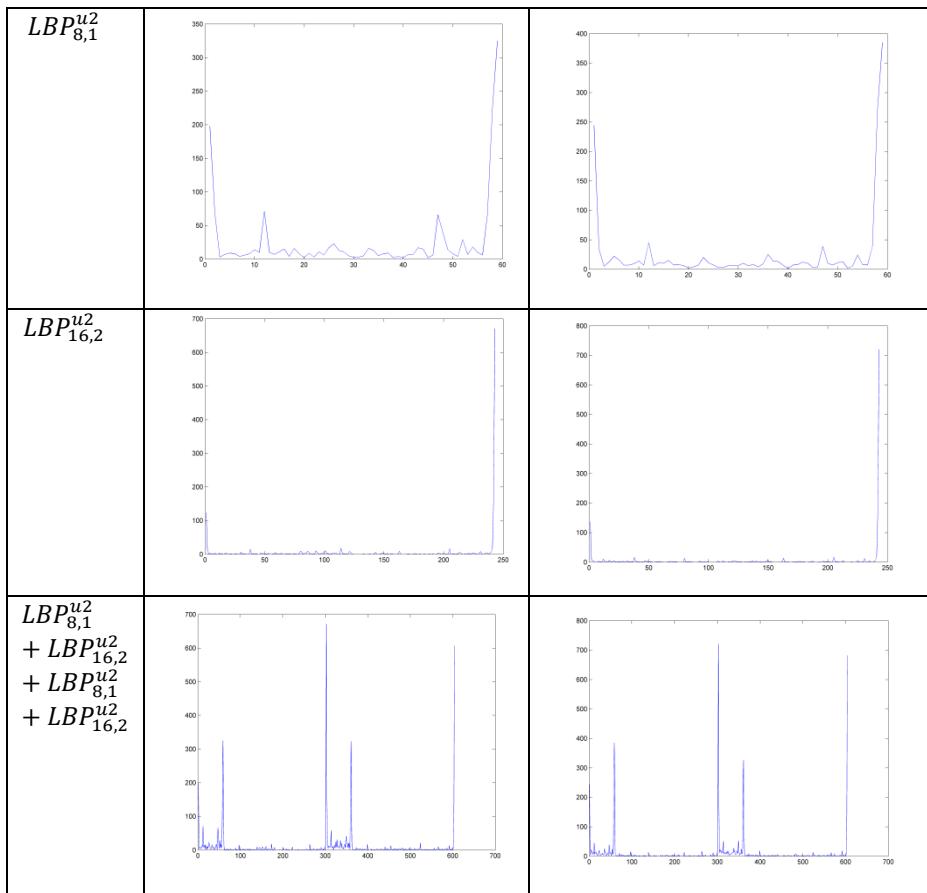
Table 1 depicts some sample output feature information obtained from the LBP algorithm for the four classes. The first results depicted are obtained from applying the LBP algorithm onto the original nodule images only. The third example is an addition of the LBP applied to the original nodule images and gradient nodule images for each feature type (i.e.  $LBP_{8,1}^{u2}$  is found using the gradient and original nodule image, same for the  $LBP_{16,2}^{u2}$ .)

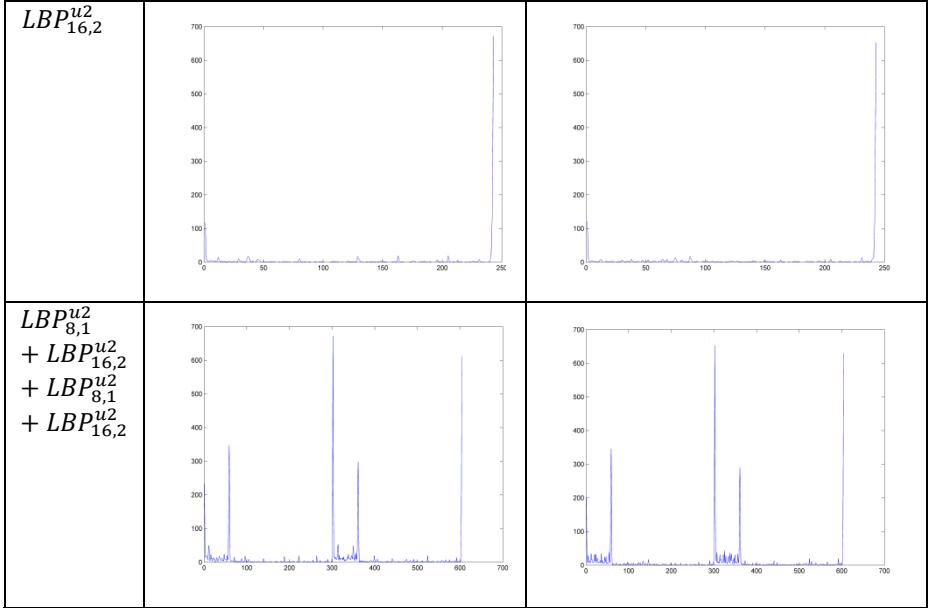


**Fig. 1.** LBP feature vector generation using original and gradient nodule images

**Table 1.** Visualization of LBP Recognition results for a sample of the Juxta, Well-Circumscribed, Pleural-Tail and Vascularized Nodules

Nodules:→	Juxta		Well	
LBP condition↓				





### 3 Experimental Results

This work is based on the ELCAP public database [17]. Classification was performed using the k nearest-neighbor, leave-one out classifier with Euclidean distance as the similarity measure. Classification results are obtained using the nodules extracted from the 50 sets of low-dose CT lung scans taken at a single breath-hold with slice thickness 1.25 mm, of the ELCAP database. The locations of the 397 nodules were provided by the radiologists, where 39.12% are juxta-pleural nodules, 13.95% are vascularized nodules, 31.29% well-circumscribed nodules and 15.65% pleural-tail nodules. In this paper, we created a subset database containing 294 nodules of the original 397. Since the assumption that the nodule region has been already detected, we use the groundtruth marked nodules by the radiologists to avoid sources of errors due to automated detection. Given a nodule's centroid, we extract texture descriptor information using two main techniques: SURF and LBP descriptors. The database resolution is 0.5x0.5 mm and scan parameters approximately 40 mAs [17].

Training in this paper was performed using a one-time random sampling approach. To quantify nodule type classification performance, we measure true positives rates. A classification result is considered a true positive if a sample from class  $w_i$  is classified as belonging to the same class.

Table 2 shows the classification results for the nodule SURF descriptor for different percentages of training data. A projection of the SURF on a PCA-based subspace is trained by the descriptors of each nodule type are also shown. This projection reduces noise information or sources of uncertainty that can arise when producing the feature descriptors. The overall results of the SURF shows that less training data was required to be able to classify the nodules.

**Table 2.** SURF Results obtained from the original SURF, and results after applying the PCA method to the SURF output

Training Percentage	Raw SURF			
	Juxta	Well	Vascular	Pleural Tail
25%	56%	45%	41%	11%
50%	57%	42%	34%	22%
75%	59%	48%	29%	22%
100%	57%	43%	32%	22%

Training Percentage	PCA SURF			
	Juxta	Well	Vascular	Pleural Tail
25%	75%	76%	73%	69%
50%	60%	73%	71%	60%
75%	66%	82%	68%	36%
100%	64%	73%	63%	51%

**Table 3.** LBP,  $LBP_{8,1}^{u^2}$ , results obtained from applying LBP to the original image only. The raw LBP, PCA and LDA projected LBP results are shown.

Training Percentage	Raw LBP (8 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	49%	46%	15%	15%
50%	49%	38%	27%	17%
75%	52%	39%	24%	22%
100%	50%	41%	24%	22%

Training Percentage	PCA LBP (8 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	65%	77%	51%	43%
50%	69%	65%	29%	46%
75%	69%	65%	32%	35%
100%	65%	68%	27%	24%

Training Percentage	LDA LBP (8 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	59%	45%	22%	28%
50%	46%	37%	17%	13%
75%	59%	47%	24%	30%
100%	64%	46%	41%	48%

Tables 3 thru 6 depict the classification results for various multi-resolution LBP algorithm experiments conducted for different percentages of training data. Results of the LBP descriptors projected onto the PCA- and LDA-based subspaces are also shown. LDA-LBP results provided overall enhanced results when comparing to its

**Table 4.** LBP,  $\text{LBP}_{16,2}^{u2}$ , results obtained from applying LBP to the original image only. The raw LBP, PCA and LDA projected LBP results are shown.

Training Percentage	Raw LBP (16 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	50%	25%	15%	17%
50%	47%	43%	37%	17%
75%	48%	43%	37%	24%
100%	43%	41%	46%	17%
Training Percentage	PCA LBP (16 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	59%	64%	56%	48%
50%	64%	60%	51%	39%
75%	64%	58%	34%	22%
100%	57%	59%	22%	26%
Training Percentage	LDA LBP (16 Features)			
	Juxta	Well	Vascular	Pleural tail
25%	52%	49%	39%	41%
50%	74%	68%	56%	52%
75%	81%	84%	80%	76%
100%	100%	99%	95%	100%

**Table 5.** LBP,  $\text{LBP}_{16,2}^{u2} + \text{LBP}_{16,2}^{u2}$ , results obtained from applying the first LBP to the original image and the second LBP to the gradient image. The raw LBP, PCA and LDA projected LBP results are shown.

Training Percentage	Raw LBP			
	Juxta	Well	Vascular	Pleural tail
25%	44%	36%	22%	22%
50%	46%	49%	44%	11%
75%	45%	46%	32%	11%
100%	48%	47%	37%	11%
Training Percentage	PCA LBP			
	Juxta	Well	Vascular	Pleural tail
25%	61%	64%	54%	50%
50%	55%	63%	41%	30%
75%	67%	59%	20%	20%
100%	56%	61%	15%	9%
Training Percentage	LDA LBP			
	Juxta	Well	Vascular	Pleural tail
25%	48%	45%	34%	33%
50%	70%	68%	54%	61%
75%	85%	85%	80%	76%
100%	100%	77%	34%	98%

raw and PCA-LBP counterparts. Table 4,  $\text{LBP}_{16,2}^{u2}$  with LDA projection yielded the best results as training increased for all nodule types. When comparing the PCA-SURF and PCA- $\text{LBP}_{16,2}^{u2}$ , the SURF algorithm provided overall better results for all

**Table 6.** LBP,  $LBP_{8,1}^{u2} + LBP_{16,2}^{u2} + LBP_{8,1}^{u2} + LBP_{16,2}^{u2}$ , results obtained from applying the LBP to the original image for the first two terms. The remaining terms LBP was applied to their gradient images. The raw LBP, PCA and LDA projected LBP results are shown.

Training Percentage	Raw LBP			
	Juxta	Well	Vascular	Pleural tail
25%	50%	37%	12%	13%
50%	54%	45%	34%	17%
75%	54%	51%	32%	20%
100%	57%	51%	24%	24%
Training Percentage	PCA LBP			
	Juxta	Well	Vascular	Pleural tail
25%	69%	82%	56%	48%
50%	64%	67%	39%	33%
75%	66%	60%	29%	17%
100%	71%	64%	20%	33%
Training Percentage	LDA LBP			
	Juxta	Well	Vascular	Pleural tail
25%	57%	51%	27%	35%
50%	71%	62%	51%	59%
75%	86%	83%	76%	76%
100%	100%	93%	85%	100%

nodule types and training, if compare the PCA-Surf with other instances of LBP where 8 features where computed the results where comparable.

## 4 Conclusion

In this paper we investigated the effects of texture analysis using two non-linear feature descriptor algorithms; the SURF descriptor, which is a variation of the SIFT algorithm, with lower-dimensional subspace projection using PCA and the multi-resolution LBP feature descriptor with lower-dimensional subspace projections using PCA and LDA. The results from the descriptors were used to classify the nodules into their corresponding classes as defined by [6]. The results revealed that the PCA-SURF method provided improved results over the PCA-LBP classification results. Overall the LDA-LBP classification resulted in the best overall classification results.

Future directions are geared towards generating a larger nodule database from other clinical data to expand our work. Further experimentations with the approaches described in this paper will be conducted as well as examination of other feature descriptor approaches to compare with the results obtained in this paper.

## References

- United States National Institute of Health, [www.nih.gov](http://www.nih.gov)
- Amal Farag, Lung Cancer: Nodule detection and False Positive Reduction from LDCT Scans, Master of Engineering Thesis, University of Louisville (May 2009)

3. Farag, A., Elhabian, S., Elshazly, S., Farag, A.: Quantification of Nodule Detection in Chest CT: A Clinical Investigation Based on the ELCAP Study. In: Proc. of 2nd Int. Workshop on Pulmonary Image Proc. in conjunction with MICCAI 2009, London, pp. 149–160 (September 2009)
4. Farag, A., Elhabian, S., Graham, J., Farag, A., Falk, R.: Toward Precise Pulmonary Nodule Type Classification. In: Proc. of the Int. Conference on Medical Image Computing and Computer-Assisted intervention (MICCAI 2010), Beijing, China, September 20–25 (2010)
5. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
6. Kostis, W.J., et al.: Small pulmonary nodules: reproducibility of three-dimensional volumetric measurement and estimation of time to follow-up. Radiology 231, 446–452 (2004)
7. van Ginneken, B., Katsuragwa, S., Romney, B., Doi, K., Viergever, M.: Automatic Detection of Abnormalities in Chest Radiographs Using Local Texture Analysis. IEEE Transactions on Medical Imaging 21(2) (2002)
8. Keramidas, E.G., Iakovidis, D.K., Maroulis, D., Karkanis, S.: Efficient and effective ultrasound image analysis scheme for thyroid nodule detection. In: Kamel, M.S., Campilho, A. (eds.) ICIAR 2007. LNCS, vol. 4633, pp. 1052–1060. Springer, Heidelberg (2007)
9. Tao, Y., Dewan, M., Chen, A., Corso, J., Xuan, J., Salganicoff, M., Krishnan, A.: Multi-level Ground Glass Nodule Detection and Segmentation in CT Lung Images. In: Yang, G.-Z., Hawkes, D., Rueckert, D., Noble, A., Taylor, C. (eds.) MICCAI 2009. LNCS, vol. 5762, pp. 715–723. Springer, Heidelberg (2009)
10. Samala, R., et al.: A Novel Approach to Nodule Feature Optimization on Thin Section Thoracic CT. Acad. Radiology. 15, 1181–1197 (2009)
11. Shlens, J.: A Tutorial on Principal Component Analysis. Institute for Nonlinear Science. UCSD (2005)
12. Mikolajczyk, K.: —Detection of Local Features Invariant to Affine Transformations Application to Matching and Recognition, PhD Thesis, Ecole Doctorale Mathématiques, Sciences et Technologies de l'Information, Informatique (MSTII), Institut National Polytechnique de Grenoble (INPG), France (2002)
13. Bay, H., Ess, A., Tuytelaars, T., et al.: Speeded-Up Robust Features (SURF). Computer Vision and Image Understanding 110(3), 346–359 (2008)
14. Bay, H.: Original SURF Code (2006),  
[http://www.vision.ee.ethz.ch/~surf/download\\_ac.html](http://www.vision.ee.ethz.ch/~surf/download_ac.html)
15. Ojala, T., Pietikainen, M., Harwood, D.: A comparative study of texture measures with classification based on feature distributions. Pattern Recognition 29, 51–59 (1996)
16. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. IEEE Transactions on Pattern Analysis and Machine Intelligence PAMI-24, 971–987 (2002)
17. ELCAP lung image database,  
<http://www.via.cornell.edu/databases/lungdb.html>
18. Balakrishnama, S., Ganapathiraju, A.: Linear Discriminate Analysis- A brief tutorial,  
[http://www.isip.msstate.edu/publications/reports/isip\\_internal/1998/linearppublications/linear\\_discrim\\_analysis/](http://www.isip.msstate.edu/publications/reports/isip_internal/1998/linearppublications/linear_discrim_analysis/)

# Multiple-object Tracking in Cluttered and Crowded Public Spaces

Rhys Martin and Ognjen Arandjelović

University of Cambridge  
Department of Engineering  
Cambridge CB2 1TQ, UK

**Abstract.** This paper addresses the problem of tracking moving objects of variable appearance in challenging scenes rich with features and texture. Reliable tracking is of pivotal importance in surveillance applications. It is made particularly difficult by the nature of objects encountered in such scenes: these too change in appearance and scale, and are often articulated (e.g. humans). We propose a method which uses fast motion detection and segmentation as a constraint for both building appearance models and their robust propagation (matching) in time. The appearance model is based on sets of local appearances automatically clustered using spatio-kinetic similarity, and is updated with each new appearance seen. This integration of all seen appearances of a tracked object makes it extremely resilient to errors caused by occlusion and the lack of permanence of due to low data quality, appearance change or background clutter. These theoretical strengths of our algorithm are empirically demonstrated on two hour long video footage of a busy city marketplace.

## 1 Introduction

In recent years the question of security in public spaces has been attracting an increasing amount of attention. While the number of surveillance cameras has steadily increased so have the problems associated with the way vast amounts of collected data are used. The inspection of recordings by humans is laborious and slow, and as a result most surveillance footage is used not preventatively but rather *post hoc*. Work on automating this process by means of computer vision algorithms has the potential to be of great public benefit and could radically change how surveillance is conducted.

Most objects of interest in surveillance footage move at some point in time. Tracking them reliably is a difficult but necessary step that needs to be performed before any inference at a higher level of abstraction is done. This is the problem we address in this paper.

### 1.1 Problem Difficulties

Public spaces are uncontrolled and extremely challenging environments for computer vision-based inference. Not only is their appearance rich in features, texture and motion – as exemplified in Figures 1(a) and 1(b) – but it is also continuously exhibiting variation of both high and low frequency in time: shopping windows change as stores



**Fig. 1.** (a) A typical frame extracted from the video footage used in the evaluation of this paper, showing a busy city marketplace, (b) a magnified image region containing examples of occlusion of objects of interests, as well as a cluttered and feature rich background, (c) the difference between two successive frames and (d) the inferred motion regions

open and close, shadows cast by buildings and other landmarks move, delivery lorries get parked intermittently etc. This is a major obstacle to methods based on learning the appearance of the background, e.g. [1–4].

Indeed, little related previous research addressed the exact problem we consider in this paper, instead concentrating on simpler recognition and tracking environments. A popular group of methods is based on grouping low-level features, for example by detecting common motion patterns [5, 6] or using cascaded appearance classifiers [7, 8]. While these tend to perform well in uncrowded scenes, they have difficulties coping with occlusions. This is particularly the case with mutual occlusions, involving multiple tracked objects. Both methods of Rabaud and Belongie [5], and Brostow and Cipolla [6], share some similarity with the method proposed in this paper, in that they consider the coherence of feature motion. However, unlike our method, their approaches rely on having long, reliable tracks of interest points. Our experiments suggests that this is not a realistic assumption for uncontrolled crowded scenes – local features are difficult to detect reliably in videos of the kind of quality which is found in practice. These usually have poor resolution and are often compressed, making most local features very short lived. This is complicated further by frequent occlusion and object articulation.

In contrast, template-based methods which employ holistic appearance struggle with the issue of variability in appearance of tracked objects and their scale [9–12], and generally have high computational demands. Zhao and Nevatia [3], for example, employ more restrictive object and scene models, in the form of human shape and ground plane calibration and assume a bird’s eye view of the scene. This approach is thus fundamentally more restrictive than ours in several important aspects. Additionally, unlike ours, models of this kind struggle with the problem of initialization which is usually manual. This is a major limitation in a practical application which involves a great number of moving entities which uncontrollably enter and leave the scene.

In summary, an algorithm successful at tracking moving entities in a crowded public space, has to, on the one hand, learn a model sufficiently persistent and discriminative to correctly track in the presence of occlusion and distinguish between potentially similar entities, yet flexible enough to allow for appearance changes due to articulation, pose and scale change. In the next section we describe the details of our approach at achieving this, followed by a section in which its performance is illustrated on real-world footage of a public square.

## 2 Algorithm Details

Our algorithm employs multiple (and in a sense complementary) representations as a means of capturing a suitably strong model that allows for reliable tracking and track continuity following partial or full occlusion, while at the same time exhibiting sufficient flexibility in changing appearance and computational efficiency. We first give an overview of the approach, followed by a detailed description of each of the steps.

### 2.1 Overview

The proposed method consists of an interlaced application of the following key algorithmic elements:

- Detection of motion regions (in all frames, across the entire frame area)
- Spatial grouping of motion regions
- Interest point detection (within motion regions only)
- Appearance model building by spatio-kinetic clustering of interest points (newly detected ones only)
- Correspondence matching of feature clusters between successive frames

We build appearance models from bottom up, grouping local features within motion regions into clusters, each cluster representing a moving object, according to the coherence of their motion and taking into account perspective effects of the scene. Permanence of appearance models is achieved by retaining all features added to a cluster even after their disappearance (which often happens, due to occlusion, articulation, or image noise, for example). Robustness in searching for feature and object correspondence between frames is gained by using constraints derived from detected motion regions, allowing us to account for occlusion or transiently common motion of two objects.

## 2.2 Detecting and Grouping Motion Regions

An important part of the proposed method lies in the use of motion regions. These are used to dramatically improve computational efficiency, reducing the image area which is processed further by focusing only on its “interesting” parts, as well as to constrain the feature correspondence search – described in Section 2.4 – which is crucial for reliable matching of appearance models of moving entities between subsequent frames.

Let  $I_t \in \mathbb{R}^{H \times W}$  be the frame (as a  $H \times W$  pixel image) at the  $t$ -th time step in the input video. At each time step, our algorithm performs simple motion detection by pixel-wise subtraction of two frames  $k$  steps apart:

$$\Delta I_t(x, y) = I_t(x, y) - I_{t-k}(x, y). \quad (1)$$

Typical output is illustrated in Figure 1(c) which shows rather noisy regions of appearance change. Note that many locations which correspond to moving entities by coincidence do not necessarily significantly change in appearance. To account for this, we employ the observation that the objects of interest have some expected spatial extent. Thus, we apply a linear smoothing operator on the frame difference  $\Delta I_t(x, y)$ :

$$C_t(x, y) = \int_{u,v} \Delta I_t(x+u, y+v) G(u, v, y) \quad (2)$$

where  $G(u, v, y)$  is an *adaptive* Gaussian filter. Specifically, the variances of the axis-aligned kernel are made dependent on the location of its application:

$$G(u, v, y | \sigma_u, \sigma_v) = \frac{1}{2\pi \sigma_u \sigma_v} \exp \left\{ -0.5 u^2 / \sigma_u(y) - 0.5 v^2 / \sigma_v(y) \right\}. \quad (3)$$

The variation of  $\sigma_u(y)$  and  $\sigma_v(y)$  is dependent on the scene perspective and the loose shape of the objects of interest. We learn them in the form  $\sigma_u(y) = c_1 y + c_2$  and  $\sigma_v(y) = c_3 y + c_2$ . As our appearance model (described next) is top-down, that is, initial hypotheses for coherently moving entities are broken down, rather than connected up, we purposefully choose relatively large  $c_1$  and  $c_3$  (0.045 and 0.25, respectively). The remaining constant is inferred through minimal user input: the user is asked to select two pairs of points such that the points in each pair are at the same distance from the camera and at the same distance from each other, and that each pair is at a different distance from the camera.

Finally, we threshold the result and find all connected components consisting of positively classified pixels (those exceeding the threshold) which we shall for brevity refer to as motion regions. On our data set, on average they occupy approximately 8% of the total frame area. Examples are shown in Figure 1(d).

## 2.3 Building Appearance Models using Spatio-Kinetic Clustering of Interest Points

Having identified regions of interest in the scene, we extract interest points in them as scale-space maxima [13]. While motion regions are used to constrain their matching

and clustering, descriptors of local appearance at interest points are collectively used to represent the appearance of tracked objects.

Each interest point's circular neighbourhood is represented by the corresponding 128-dimensional SIFT descriptor [13]. These are then grouped according to the likelihood that they belong to the same object. Exploiting the observation that objects have limited spatial extent, as well as that their constituent parts tend to move coherently, we cluster features using both spatial and motion cues, while accounting for the scene geometry.

The spatial constraint is applied by virtue of hierachial clustering – only the  $K$  nearest neighbours of each interest point are considered in trying to associate it with an existing cluster. Using a limited velocity model, an interest point and its neighbour are tracked  $N$  frames forwards and backwards in time to extract the corresponding motion trajectories. Let the motion of a tracked interest point be described by a track of its location through time  $\{(x_t, y_t)\} = \{(x_{t_1}, y_{t_1}), (x_{t_1+1}, y_{t_1+1}), \dots, (x_{t_2}, y_{t_2})\}$  and that of its  $i$ -th of  $K$  nearest neighbours  $\{(x_t^i, y_t^i)\} = \{(x_{t_1}^i, y_{t_1}^i), (x_{t_1+1}^i, y_{t_1+1}^i), \dots, (x_{t_2}^i, y_{t_2}^i)\}$ , where the interval  $[t_1, t_2]$  is determined by the features' maximal past and future co-occurrence. The two interest points are associated with the same appearance cluster – a cluster being the current best hypothesis of a single moving entity – if they have not been already associated with separate clusters and the motion incoherence of the corresponding trajectories does not exceed a threshold  $t_{\text{coherence}}$ :

$$\sum_{t=t_1}^{t_2} \left\| \frac{(x_t, y_t) - (x_t^i, y_t^i)}{(y_t + y_t^i) / 2 + c_2} \right\|^2 - \left( \sum_{t=t_1}^{t_2} \left\| \frac{(x_t, y_t) - (x_t^i, y_t^i)}{(y_t + y_t^i) / 2 + c_2} \right\| \right)^2 < t_{\text{coherence}}. \quad (4)$$

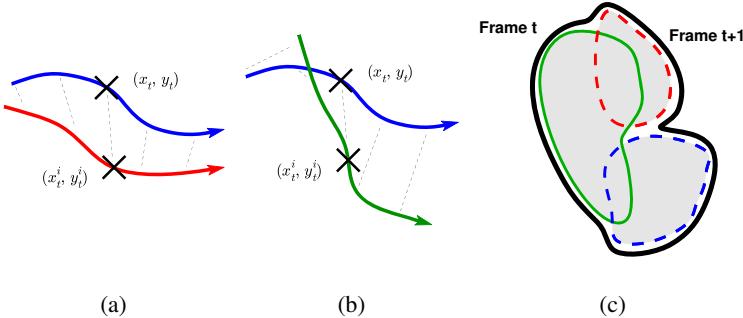
as conceptually illustrated in Figures 2(a) and 2(b). The coherence measure in Equation 4 accounts for the previously learnt perspective of the scene by inversely weighting the distance between two features by their distance from the horizon. Note that we make the implicit assumption that the vertical position of the camera is significantly greater than the height of tracked objects (if this assumption is invalidated, the denominators in Equation 4 can reach a small value without the objects being near the horizon).

The result of the described spatio-kinetic clustering is a set of clusters per each motion region. These are associated with the region only temporarily and it is not assumed that they correspond to the same object (indeed, in most cases they do not due to different motion characteristics).

## 2.4 Model Propagation through Time

Having learnt quasi-permanent appearance models of objects (as their constituent features are being detected using the approach described in Section 2.3), we turn our attention to the question of tracking these through time.

Consider a cluster of features in a particular motion region and the problem of localizing this cluster in the subsequent frame. We know that the features which belong in it move coherently and we know that this motion is limited in velocity. However, the corresponding motion region may no longer exist: the features may have temporarily ceased moving, or the objects which comprised a single motion region may have parted (e.g. two people separating, or after temporary occlusion), or it may have joined



**Fig. 2.** Kinetic (a) coherence and (b) discrepancy result in two features with spatial proximity getting assigned to respectively the same and different clusters. (c) A feature located in a specific motion region in frame  $I_t$  is searched for in the subsequent frame  $I_{t+1}$  in the area occupied by the initial motion region (green, solid) and all motion regions that intersect it in  $I_{t+1}$  (blue and red, dashed).

another (e.g. two people meeting or occluding each other). To account for all of these possibilities, each feature is searched for in the area occupied by the original region it was detected in and all the regions in the subsequent frame which intersect it, as shown in Figure 2(c).

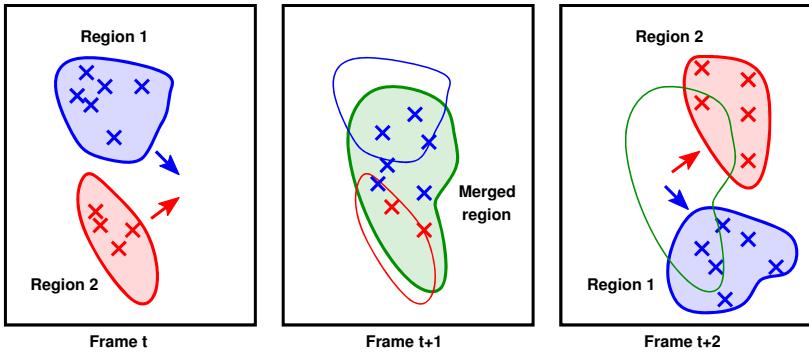
Consider an interest point with the appearance at the time step  $t$  captured by the corresponding SIFT descriptor  $\mathbf{d}_t \in \mathbb{R}^{128}$ . It is matched to that in the next frame  $t+1$  and within the search area, which has the most similar appearance,  $\mathbf{d}_{t+1}^k$ , provided that their similarity exceeds a set threshold according to the following criterion:

$$\mathbf{d}_t \xrightarrow{\text{match}} \mathbf{d}_{t+1}^k \quad (5)$$

where

$$k = \begin{cases} \arg \min_i \rho(i) & \rho(k) \leq t_{\text{feature}} \\ \text{new feature} & \rho(k) > t_{\text{feature}} \end{cases} \quad \text{and} \quad \rho(i) = \frac{\mathbf{d}_t^T \mathbf{d}_{t+1}^i}{\|\mathbf{d}_t\| \|\mathbf{d}_{t+1}^i\|} \quad (6)$$

Features from the same cluster which are localized within the same motion region in the new frame are associated with it, much like when the model is first built, as described in Section 2.3. However, the cluster can also split when its constituent features are localized in different motion regions (e.g. when two people who walked together separate). Cluster splitting is effected by splitting the appearance model and associating each new cluster with the corresponding motion region. On the other hand, notice that clusters are never joined even if their motion regions merge, as illustrated in Figure 3. This is because it is the clusters themselves which represent the best current estimate of individual moving objects in the scene, whereas motion regions merely represent the image plane uncertainty in temporal correspondence, caused by motion of independent entities.



**Fig. 3.** A conceptual illustration showing the robustness of our appearance model in coping with partial or full occlusion regardless of its duration

## 2.5 Review of Handling of the Key Tracking Problems

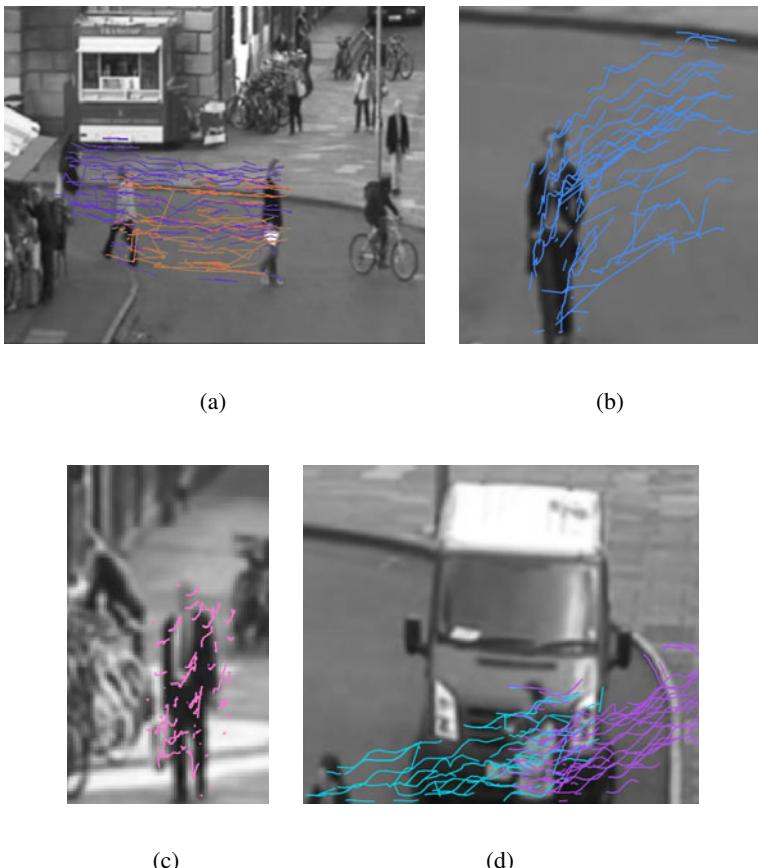
As a conclusion to the theoretical section of the paper, let us consider how our algorithm copes with some of the most important tracking scenarios.

- **Case 1, Independently moving entity:** An appearance model is built from the appearances of clustered local features. As the cluster is tracked by matching those features which are at the time reliably matched, the model in the form of a set of appearance features (many of which are *not* visible at any point in time) is constantly enriched as new appearances are observed.
- **Case 2, Coherently moving entities, which separate:** Motion incoherence after separation is used to infer separate entities, which are back-tracked to the time of common motion when the corresponding clusters (as their feature sets) are associated with the same moving region. Novel appearance, in the form of new local features is added to the correct appearance cluster using spatial constraints.
- **Case 3, Separately moving entities, which join in their motion:** This situation is handled in the same manner as that described previously as Case 2, but with tracking proceeding forwards, not backwards in time.
- **Case 4, Partial occlusion of a tracked entity:** The proposed appearance model in the form of a set of appearances of local features, is inherently robust to partial occlusion – correct correspondence between clusters is achieved by matching reliably tracked, visible features.
- **Case 5, Full occlusion of a tracked entity:** When a tracked entity is occluded by another, both of their clusters are associated with the same motion region. This association continues until sufficient evidence for the occluded entity re-emerges and a new motion region is detected. At that point the overlap of regions of interest is used to correctly match appearance models, separating them and re-assigning feature clusters with the correct moving regions.

An empirical demonstration of these theoretical arguments is presented next.

### 3 Empirical Analysis

To evaluate the effectiveness of the proposed method we acquired a data set fitting the problem addressed in this paper and containing all of the challenging aspects described in Section 1. Using a stationary camera placed on top of a small building overlooking a busy city marketplace we recorded a continuous video stream of duration 1h:59m:40s and having the spatial resolution of  $720 \times 576$  pixels. A typical frame is shown in Figure 1(a) while Figure 1(b) exemplifies some of the aforementioned difficulties on a magnified subregion.



**Fig. 4.** Experimental results on a nearly two hour long video footage of a busy marketplace confirm the advantages of our method predicted from theory. Feature tracks corresponding to each person's model are shown in different colours. Illustrated is our method's robustness to (a) mutual occlusion of tracked objects, (b,c) successful tracking of an object in the presence of scale change, and unstable and changing set of detected local features associated with the object's appearance model, and (d) a combination of mutual occlusion, cluttered and texture-rich background, scale change and gradual disappearance of a tracked object from the scene.

Experimental results we obtained corroborate previously stated strengths of our method expected from theory. The permanence of the proposed model which captures all seen appearances of a tracked object, coupled with a robust frame-to-frame feature matching, makes it particularly resilient to errors caused by occlusion. An example of this can be seen in Figure 4(a). It shows feature tracks associated with automatically learnt appearance models corresponding to two people (shown in different colours – green and purple), which are then successfully tracked even following their mutual occlusion, that is, after one passes in front of the other.

A magnification of a isolated person being tracked in Figure 4(b) and another at an approximate 50% smaller scale in Figure 4(c), serve to illustrate the role of several building elements of our algorithm. Specifically, it can be observed that few features last for more than 0.5s in the former example and more than 0.1s in the latter. This is a consequence of appearance change due to motion and articulation, as well as image and spatial discretization noise. It is the incremental nature of our algorithm, whereby novel features are added to the existing model, and the use of spatio-kinetic clusters, which allows all of the shown tracks to be associated with the same moving object. These examples should not be correctly tracked by such previously proposed method as those of Rabaud and Belongie [5], and Brostow and Cipolla[6].

Finally, Figure 4(d) shows successful tracking in the presence of several simultaneous difficulties: the two tracked people cross paths, mutually occluding, in front of a feature-rich object, one of them progressively disappearing from the scene and both of them changing in scale due to the movement direction. As before, many of the associated features are short lived, disappearing and re-appearing erratically.

## 4 Summary and Conclusions

In this paper we described a novel method capable of automatically detecting moving objects in complex cluttered scenes, building their appearance models and tracking them in the presence of partial and full occlusions, change in appearance (e.g. due to articulation or pose changes) and scale. The proposed algorithm was empirically evaluated on a two hour long video footage of a busy city marketplace and the claimed theoretical properties of the approach substantiated by through successful performance on several difficult examples involving the aforementioned challenges.

## References

1. Wren, C., Azarbayejani, A., Darrell, T., Pentland, A.: Pfnder:real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 19, 780–785 (1997)
2. Haritaoglu, I., Harwood, D., David, L.: W4:real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)* 22, 809–830 (2000)
3. Zhao, T., Nevatia, R.: Tracking multiple humans in crowded environment. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 406–413 (2004)

4. Isard, M., MacCormick, J.: Bramble: a Bayesian multiple-blob tracker. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 34–41 (2001)
5. Rabaud, V., Belongie, S.: Counting crowded moving objects. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition, CVPR (2006)
6. Brostow, G.J., Cipolla, R.: Unsupervised Bayesian detection of independent motion in crowds. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 1, pp. 594–601 (2006)
7. Gavrila, D.M.: Pedestrian detection from a moving vehicle. In: Proc. European Conference on Computer Vision (ECCV), vol. 2, pp. 37–49 (2000)
8. Viola, P., Jones, M., Snow, D.: Detecting pedestrians using patterns of motion appearance. In: Proc. IEEE International Conference on Computer Vision (ICCV), pp. 734–741 (2003)
9. Tu, P., Sebastian, T., Doretto, G., Krahnstoever, N., Rittscher, J., Yu, T.: Unified crowd segmentation. In: Proc. European Conference on Computer Vision (ECCV), vol. 4, pp. 691–704 (2008)
10. Zhao, T., Nevatia, R., Lv, F.: Segmentation and tracking of multiple humans in complex situations. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), vol. 2, pp. 194–201 (2001)
11. Lipton, A., Fujiyoshi, H., Patil, R.: Moving target classification and tracking from real-time video. In: Proc. DARPA Image Understanding Workshop (IUW), pp. 8–14 (1998)
12. Matthews, I., Ishikawa, T., Baker, S.: The template update problem. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) 26, 810–815 (2004)
13. Lowe, D.G.: Local feature view clustering for 3D object recognition. In: Proc. IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pp. 682–688 (2001)

# Compliant Interframe Coding for Motion-JPEG2000

René Rosenbaum<sup>1,\*</sup> and Heidrun Schumann<sup>2</sup>

<sup>1</sup> Institute of Data Analysis and Visualization (IDA)

Department of Computer Science

University of California, Davis, CA 95616, U.S.A.

<sup>2</sup> Visual Computing and Computer Graphics

Institute of Computer Science

University of Rostock, 18059 Rostock, Germany

**Abstract.** This publication proposes a strategy to enhance the compression performance of Motion-JPEG2000 by the reduction of interframe redundancies. This is achieved by a new frame structure allowing to remove redundancies and to reconstruct the original stream without loss. The introduced advancement is able to increase the compression efficiency and keeps other beneficial property of the traditional codec, as, high-quality frame-wise access and scalability. All of the required operations are accomplished in JPEG2000 domain by examining and handling still encoded contents. Thus, the approach requires little computing power. Our results demonstrate that the strategy is at least as efficient as the original codec even for less suited video contents.

## 1 Introduction

Motion-JPEG2000 (M-JPEG2000) is a modern video compression standard [1] combining low complexity processing and high quality content reconstruction. This makes it an excellent and distinct solution for applications areas that are characterized by resource-limited hardware and the need for high-precision viewing. However, the compression performance is not as high as provided by other state-of-art video codecs [2]. This is a significant drawback which hinders the migration of the codec in areas also requiring permanent storage, as digital cinema and archiving, or transmission, as video surveillance and remote sensing [3,4]. A solution for this particular problem not constraining other features of the codec has still not been found.

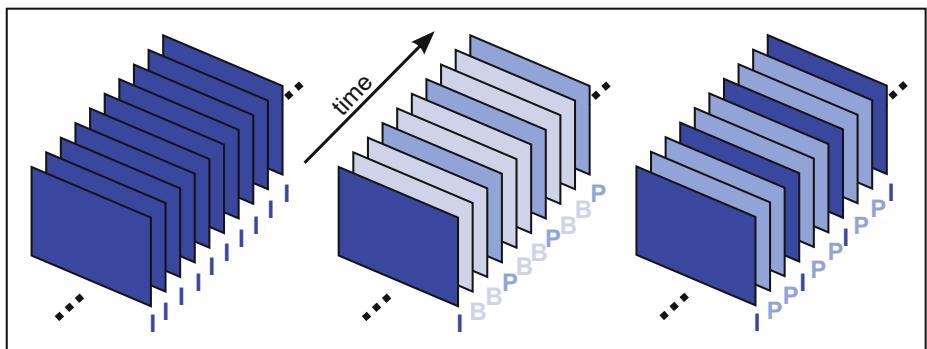
This publication introduces such a solution based on the reduction of inter-frame redundancies within the video-stream. Instead of encoding and keeping all contents of a single frame, the proposed approach is founded on a frame-wise detection and removal of image regions which have not changed. A new frame

---

\* The author gratefully acknowledges the support of Deutsche Forschungsgemeinschaft (DFG) for funding this research (#RO3755/1-1) and Bernd Hamann for providing valuable feedback.

structure thereby ensures unique identification of non-redundant data pieces and lossless reconstruction of the original stream. The fact that all operations are applied in JPEG2000 domain allows for the application of the approach to uncompressed or already M-JPEG2000-compressed video contents at low complexity. As the reduced stream is still compliant to the M-JPEG2000 standard, it can be stored or streamed with any technology developed for this codec. Thereby, it does not interfere with other benefits and features of the codec which significantly simplifies the migration of the proposed technology into already existing system. However, a modification of the decoder is required in order to restore all video contents.

M-JPEG2000 has a distinct application area. Thus, the proposed approach is not intended to replace other video codecs, but to significantly enhance the efficiency of M-JPEG2000. Section 2 reviews the codec and derives the reasons for its worse compression performance. Founded on this, a new strategy to enhance the compression efficiency for storage (Section 3) and transmission (Section 4) is proposed. The achieved results are discussed in Section 5. Conclusions and future work close this contribution in Section 6.



**Fig. 1.** Structure of a data-stream encoded with M-JPEG2000 (left), a codec taking advantage of prediction (center), and the proposed advancement (right)

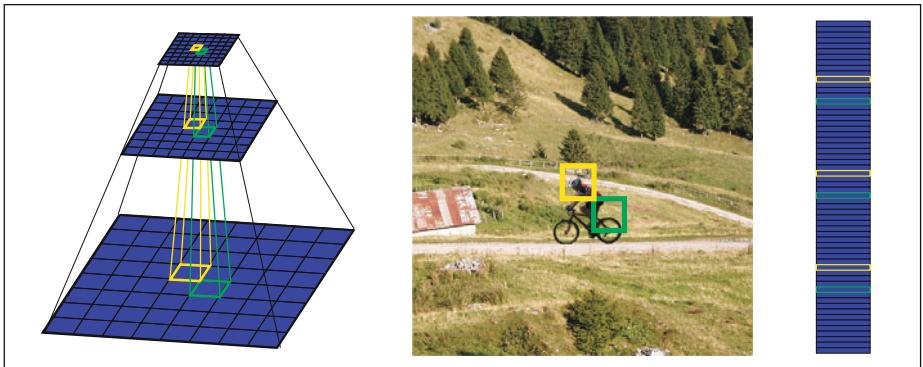
## 2 Motion-JPEG2000 and Related Work

M-JPEG2000 is the successor of a variety of non-standard Motion-JPEG methods used to create and access consecutive JPEG frames at reasonable quality [5]. M-JPEG2000 collects all frames of the sequence in a standardized data format, which may also include audio data and means to time the media. There may be multiple *tracks* each consisting of multiple frames. The frame sequence of each track has the structure as depicted in Figure 1/left. A significant advantage of M-JPEG2000 and distinct feature with regard to related video codecs is that all frames of the sequence are individually *intraframe* encoded using the JPEG2000 baseline technology. Thus, the codec benefits from all the advantageous properties of JPEG2000, as high compression efficiency of each single frame, inherent

scalability, and error resilience support. As the proposed approach is mainly founded on the independent access and modification of single frames, the remainder of this section deals with relevant parts of the JPEG2000 codec only. For more details on M-JPEG2000, the interested reader is referred to [6] and [1].

JPEG2000 is based on the DWT and Embedded Block Coding with Optimized Truncation (EBCOT) [7]. In  $k$  DWT - decomposition stages,  $d = 1, 2, \dots, k$ , the image is transformed into  $3k + 1$  subbands,  $\text{LL}_k$  and  $\text{LH}_d$ ,  $\text{HL}_d$  and  $\text{HH}_d$ . Due to the underlying dyadic decomposition subbands dimensions at stage  $d$  are half the size of stage  $d-1$ . The DWT leads to an inherent multi-resolution representation of the image. Data required to reconstruct a certain spatial region is spread over different levels (cf. Figure 2/left).

Each subband is further partitioned into rectangular blocks, known as *code-blocks*, each of which is independently coded into a finely embedded bit-stream. Due to the overlapping property of the wavelet synthesis operations, code-blocks associated with one region in the image have a slight influence over neighboring spatial regions.



**Fig. 2.** Multiresolution representation of a single video frame in JPEG2000 domain (left). The pyramids mark those parts that must be considered to reconstruct the corresponding region in pixel domain (center). The respective contributions are scattered over the whole sequential data-stream (right).

Although each code-block is coded independently, their bit-streams are not explicitly identified within a JPEG2000 data-stream. Instead, code-blocks are collected into larger groupings known as *precincts*. This partition has no impact on the transformation or coding of image samples; it serves only to organize code-blocks. Precinct dimensions can be selected in such a way as to control the spatial contribution of code-blocks. Thus, it is possible to form pyramids of elements, whereby each pyramid covers a different spatial region (cf. Figure 2/left and center). This feature has been labeled *Limited Spatial Access* (LSA) [8] and is crucial for content manipulation in JPEG2000 domain.

The precinct partition has also great influence on the appearance of the final data-stream. Each precinct is represented as a collection of packets with one packet including an incremental contribution from each of its code-blocks on a

certain *quality layer*. As the identification of each packet is derived from their position within the data-stream, an *empty header packet* must be included if there is no contribution of a precinct to a certain layer. A complete JPEG2000 data-stream consists of header information and a concatenated list of packets.

Despite the sophisticated intraframe encoding, M-JPEG2000 lacks of compression performance compared to other standards, as H.264/AVC [2] or the widely accepted MPEG family ([9][10][11]). The performance gain of these codecs is mostly founded on the removal of existing interframe redundancies. A common strategy is the use of a more complex frame structure (cf. Figure 1/center) and the application of prediction for *motion compensation*. It is based on the fact that many contents do not substantially change but move their spatial positions. By appropriate detection and encoding of these movements, the belonging regions can be efficiently encoded and reconstructed. A frame either describes all contents (*I-frame*) or content which is predicted by using knowledge from prior frames (*P-frame*), or prior and later frames (*B-frame*). However, motion compensation is complex and reduces the options for accurate temporal access. Due to the application of prediction, P- and B-frames are usually reconstructed in low quality. Our approach adopts ideas from prediction-based codecs, but requires much less computing power, provides accurate frame-wise access and lossless encoding, and can be efficiently applied to already encoded data sources.

### 3 Introducing Interframe Coding to Motion-JPEG2000

The primary goal of the proposed advancement is to increase the compression performance of M-JPEG2000 by removing interframe redundancies. To achieve this, we propose a regions-wise *detection* and *encoding* of changes between frames. Thereby, we consider the constraints of resource-limited hardware and keep the required consumption of computing power very low. This can be accomplished by processing the frame content directly in JPEG2000 domain instead of its pixel representation. Here, the options provided by JPEG2000 for random frame- and region-wise access are of crucial advantage. The removal of interframe redundancies is accomplished by the omission of those image parts which do not change with regard to the previous frame. The result is a compliant M-JPEG2000 video, whereby selected frames are condensed to non-redundant data. To avoid the complexity of a complete motion compensation, the proposed codec neglects any content movement and considers spatially corresponding image regions only. The original content of an arbitrary frame can easily be reconstructed based on its ancestor. Thereby, all benefits of M-JPEG2000 are kept. The following sections explain the respective stages in more detail.

#### 3.1 Detection of Regions with Changed Contents

The detection of changed image regions requires additional consideration in case the sequence has already been encoded. To avoid the strong computing power required to decode each single frame, we propose a strategy based on the comparison of single precinct contributions in JPEG2000 domain. The main problem

here is the prevention of *false positives* (elements considered as modified, but they are not) and *false negatives* (elements considered as identical, but they contain changed content).

**The main idea.** As each precinct of a frame has a counterpart at the previous frame, the main idea of our detection strategy is a comparison in the length of the associated bit streams. As the length of the encoded data strongly depends on the content, it can be assumed that temporal changes alter this property. Thus, a comparison of all corresponding elements within the previous and current frame clearly reveals the modified regions and avoids false negatives.

Due to the spatial overlap between adjacent precincts, their bit-streams may also slightly vary if content has changed in neighboring precincts. As these elements do not belong to the modified region in pixel domain, the proposed strategy might be vulnerable to false positives. However, such precincts might also be considered as non-redundant in order to avoid reconstruction artifacts.

**Detection strategies.** The detection procedure should be of low complexity. Although access to precincts can be accomplished quickly [8], comparison and further processing of all corresponding elements for every frame pair is costly. To overcome this, we propose three strategies of varying complexity taking advantage of the multiresolution property of the data to derive information for precincts contributing to the same region. Although this reduces complexity, it might also lead to wrong decisions, and thus, to a reduction in quality.

**Mode 0:** All precincts are compared. If one precinct differs from its counterpart, all precincts belonging to the same region-pyramid are replaced.

This mode achieves the best visual quality, but also has highest complexity.

**Mode 1:** Starting at decomposition level 1, all precincts are compared. If one differs from its counterpart, only this and the remaining precincts from the same region-pyramid residing at higher levels are replaced.

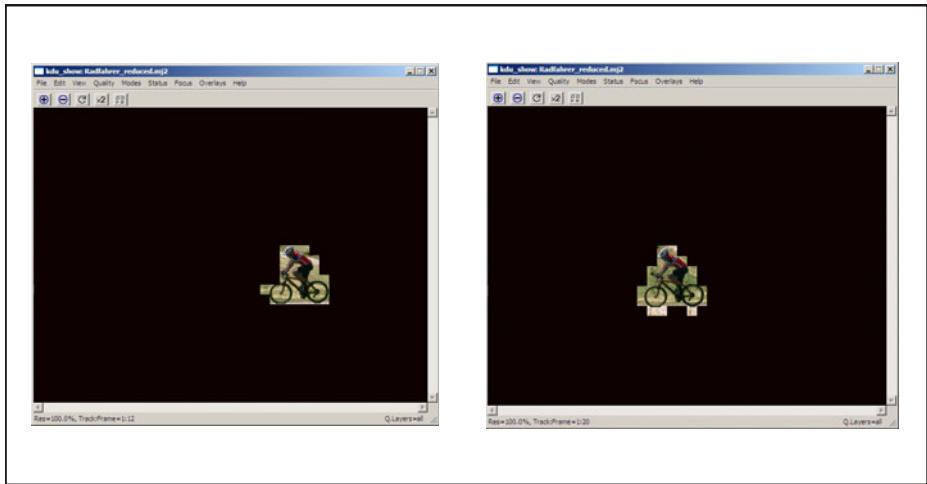
This mode is based on the fact that the spatial overlap of elements increases at higher pyramid levels. It is considered to achieve the best trade-off between the number of replaced precincts and the resulting quality.

**Mode 2:** Only precincts belonging to decomposition level 1 are compared.

If one precinct differs from its counterpart, all precincts of the the same region-pyramid are replaced.

As for these precincts the spatial influence is smallest, the number of false positives can be reduced. This, however, might also lead to an increased number of false negatives, and thus to a worse quality. As only a small number of precincts is compared, complexity is low.

In case the differences between two elements are small, e.g., caused by noise, they might also be negligible. To handle such events, all introduced modi may be combined with a *threshold* stating the allowed relative difference between elements. As this difference is founded on the length of the encoded data, reasonable values vary depending on the respective contents, and thus cannot be



**Fig. 3.** Two P-frames taken from the static test sequence used in our experiments. Each frame is compliant to M-JPEG2000 and contains only contents from image regions that have changed to the previous frame.

applied generally as a mean to control the quality of the resulting image. High threshold values might lead to artifacts or false negatives.

### 3.2 Encoding of the Changed Regions

Once the relevant image regions have been identified, the actual interframe coding can take place. This is accomplished by (1) a novel frame structure of the data-stream and (2) data reduction.

**A novel frame structure.** Contrary to the homogenous structure of an original M-JPEG2000 data-stream, a second frame type is introduced for content-reduced frames (cf. Figure 10/right). As within the strategies founded on motion compensation, *P-frames* are used to contain non-redundant content with regard to the last I- or P-frame that comes temporally before it. A reasonable sequence starts with an I-frame followed by a certain number of P-frames. The number of subsequent P-frames influences the resulting compression performance and the granularity of the frame access. The total number of frames within the original and modified data-stream is identical.

**Data reduction.** Data reduction is applied to the introduced P-frames only. They are considered to contain only data from regions which have been detected as changed with regard to the last frame. Contents which have been identified as identical are removed.

In order to ensure low complexity, we propose to take advantage of LSA and accomplish the removal directly in JPEG2000 domain. This stage may be part of the encoding process (uncompressed contents) or implemented by a transcoder

(M-JPEG2000-compressed contents). By removing all precinct contributions of a region-pyramid, an image region can be completely removed without significantly influencing adjacent areas. In order to allow for identification of the remaining precinct contributions and to keep compliance, however, each removed data packet must be replaced by an empty header packet. This leads to a compliant frame with reduced contents (cf. Figure 3). The reduction can be implemented by applying strategies for content exchange in JPEG2000 domain, as proposed in [12] or [8], or by using appropriate JPEG2000 toolkits [13].

Although, data-streams modified this way can be decoded by any M-JPEG2000-compliant decoder, the interframe coding must be undone to reconstruct the original content of each P-frame. To achieve this, the same principle as proposed for data reduction can be applied. Contrary to data reduction, they are included from the previous frame. Non-redundant elements can easily be identified by their non-empty data contributions and are not replaced. An incremental frame-wise reconstruction leads to the original M-JPEG2000 stream. Fast frame-wise access can be achieved by limiting the reconstruction to corresponding I- and P-frames.

## 4 Content Streaming

The introduced procedure for interframe coding in M-JPEG2000 sequences described in the last section leads to a reduced and compliant M-JPEG2000 data-stream that may be submitted and reconstructed without further modification. By applying modern image transmission technology, as the *JPEG2000 Interactive Protocol* (JPIP) [14], however, further reduction of resource consumption can be achieved. To accomplish this, we take advantage of two basic mechanisms: (1) *external signalization*, and (2) *granular caching* of the encoded contents. The first is used to reduce the transmitted data, the second to decrease the complexity of content reconstruction. Both strategies are mutually independent and can be applied exclusively or in combination.

**External signalization** means that to each transmitted data chunk is assigned a unique ID to flexibly overcome transmission errors or delays. As this feature inherently provides the identification of non-redundant precinct contributions, the introduction of empty header packets is not longer required and less data must be transferred.

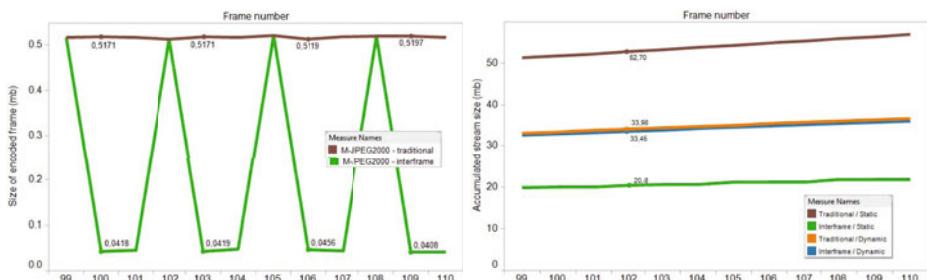
**Granular caching** allows for a highly selective access to already received data. We propose to organize the cache precinct-wise [14][13]. If it is assumed that all contents belonging to the previous frame are fully cached and have already been displayed, the next frame can be easily reconstructed by overwriting and decoding those entries for which content is received. All remaining precincts represent static content with regard to the last frame. Decoding of the cache always restores the content of the last received frame.

## 5 Results

This section presents the results we achieved for the proposed approach. Thereby, we focus on the main objective of this publication – increased *compression performance* –, but also consider the required *computing power* and reconstruction *quality*. We also discuss the *drawbacks* of the approach. As the proposed approach is primarily intended to enhance the efficiency of M-JPEG2000, we do not address comparison of our methodology to related video codecs.

**Compression performance.** The proposed enhancement is based on the removal of precinct contributions belonging to regions which have been detected as identical. The measures depicted in Figure 4 allow for comparison between the performance achieved with and without applying the advancement. As the achieved performance gain depends on similarities between compared frames, we present results for two sequences with a distinct characteristic. Within the first sequence, many image regions are *static* and do not vary between adjacent frames (cf. Figure 3). Contrary, most of the content within the second sequence is *dynamic*. In order to keep accurate temporal access and strong error resilience support, we used a low P- to I-frame ratio of 2:1 within the reduced stream.

Figure 4 left allows for visual comparison in the compression efficiency for individual frames. While performance is nearly constant for the traditional codec, a significant reduction is achieved by the proposed approach. The upward deflections clearly indicate the unchanged I-frames that are left unchanged with regard to the traditional encoding. Much less volume is required for P-frames. Figure 4/right illustrates the overall compression gain and demonstrates that best performance is achieved if the enhancement is applied to a mostly static scene. As much of the content is removed from the P-frames, the stream is approximately *2.8 times smaller* than the M-JPEG2000 source stream. The performance gain is much smaller for the considered dynamic sequence (approximately 1%) as the proposed approach does not apply full motion compensation. In such cases, we propose to apply a reasonable detection threshold in order to increase performance.



**Fig. 4.** Applied to mostly static contents, there is a significant difference in the volume of the compressed frames (left). This leads to a much better overall performance (right). For dynamic scenes the benefit is small.

**Computing power.** The proposed technique consists of two stages: detection and encoding. Due to the low complexity of data processing in JPEG2000 domain [12][8], the additional computing power required by the proposed advancement is little. This also applies for the reconstruction of the original data-stream. Thereby, the increase depends on the volume of the contents that must be removed (encoding) or restored (reconstruction). For more detailed statements to the resource requirements and exact complexity values for different exchange setups, the reader is referred to [8]. In video transmission, the proposed cache-based reconstruction requires even less computing power and memory than standard M-JPEG2000 decoding. This is due to the fact that less data is received, processed, cached, and decoded.

**Quality.** The quality of the reconstructed image sequence is influenced by the detection strategy for changed contents only. There is no degradation in quality in pixel domain. This does not apply for already encoded contents. Depending on the chosen mode, the original data-stream is restored losslessly (Mode 0, no threshold) or at some degree of quality loss (Mode 2, high threshold). *Quality correlates with the need for bandwidth and computing power.*

The proposed approach increases compression performance, exhibits low complexity, and can be applied broadly. However, there are two general **drawbacks**: (1) Coarse spatial access to the encoded data and (2) constrained error resilience support. The first point is imposed by the alignment restraint to the contents in JPEG2000 domain. This reduces spatial access granularity, and thus, the probability that identical elements can be detected. This can be reducing the sizes of the elements, but not without loss in compression efficiency. Dimensions of  $32 \times 32$  for LSA have proved to perform well for the majority of the tested sequences. The second drawback is caused by the use of P-frames that reuse encoded contents from previous frames. This leads to the propagation of potential detection or transmission errors across frames. This, however, is a general disadvantage of prediction-based codecs and can be reduced by a low I- to P-frame ratio.

## 6 Conclusions

We proposed a strategy to increase the compression performance of Motion-JPEG2000 by introducing interframe coding into the codec. The approach takes advantage of a new frame structure and is founded on the removal of data belonging to image regions which have not changed with regard to the previous frame. All required operations are accomplished in M-JPEG2000 domain leading to a *low complexity* codec. It may be applied to plain or already encoded imagery in desktop or transmission setups and leads to M-JPEG2000-compliant data-streams. The *highest compression gain* can be achieved for static setups as available during video surveillance or remote sensing. In the given example an already compressed video could be further *reduced by factor 2.8*. For highly dynamic contents, the gain is small.

Future work will focus on a more complex transcoding of the encoded data in order to increase spatial access and thus compression performance.

## References

1. ISO/IEC: JPEG 2000 image coding system, part3, cor. 1. Final Draft International Standard part 3, ISO/IEC JTC 1/SC 29/WG 1 N2705 (2002)
2. Marpe, D., George, V., Cycon, H.L., Barthel, K.U.: Performance evaluation of motion-jpeg2000 in comparison with h.264/avc operated in pure intra coding mode. In: Proceedings of SPIE - Conference on Wavelet Applications in Industrial Processing (2003)
3. Dufaux, F., Ebrahimi, T.: Motion jpeg2000 for wireless applications. In: Proceedings of First International JPEG2000 Workshop (2003)
4. Pearson, G., Gill, M.: An evaluation of motion jpeg 2000 for video archiving. In: Proceedings of Archiving 2005, pp. 237–243 (2005)
5. Shi, Y.Q., Sun, H.: Image and Video Compression for Multimedia Engineering. CRC Press, Inc., Boca Raton (1999)
6. Taubman, D., Marcellin, M.: Jpeg2000: Standard for interactive imaging. Proceedings of the IEEE 90, 1336–1357 (2002)
7. Taubmann, D.: High performance scaleable image compression with ebcot. In: Proceedings of IEEE International Conference On Image Processing, vol. 3, pp. 344–348 (1999)
8. Rosenbaum, R., Schumann, H.: Limited spatial access in jpeg2000 for remote image editing. In: Proceedings of VII 2004 (2004)
9. ISO/IEC: Information technology: Coding of moving pictures and associated audio for digital storage media at up to about 1.5mbits/s, international standard, part 2: Video. Final, ISO/IEC 11172-2 (1993)
10. ISO/IEC: MPEG-2 information technology - generic coding of moving pictures and associated audio information: Video. Final, ISO/IEC JTC 1 13818-2 (1995)
11. ISO/IEC: MPEG-4 overview (V.21 Jeju Version). Final, ISO/IEC JTC 1/SC 29/WG 11 N4668 (2002)
12. Rosenbaum, R., Taubman, D.S.: Merging images in the jpeg2000 domain. In: Proceedings of VII 2003 (2003)
13. KAKADU: A comprehensive, heavily optimized, fully compliant software toolkit for jpeg2000 developers. Version 4.5 (2010), <http://www.kakadusoftware.com/>
14. ISO/IEC: JPEG 2000 image coding system, part9. Final Draft International Standard part 9, ISO/IEC JTC 1/SC 29/WG 1 N3052R (2004)

# EVP-Based Multiple-View Triangulation

G. Chesi and Y.S. Hung

Department of Electrical and Electronic Engineering

University of Hong Kong

Pokfulam Road, Hong Kong

Tel.: +852-22194362 (G. Chesi), +852-28592675 (Y.S. Hung)

Fax: +852-25598738

{chesi,yshung}@eee.hku.hk

<http://www.eee.hku.hk/~chesi>, <http://www.eee.hku.hk/~yshung>

**Abstract.** This paper addresses multiple-view  $L_2$  triangulation by proposing a new method based on eigenvalue problems (EVPs), which belong to the class of convex programming. The proposed method provides a candidate of the sought 3D point and a straightforward condition for establishing its optimality, which also yields a guaranteed range for the optimal cost of the triangulation problem in case of non-optimality. The proposed method is illustrated through some well-known examples with real data, for which the provided candidate 3D point is always optimal. These examples also show that the computational time of the proposed method is indeed small and competitive with existing approaches.

## 1 Introduction

Multiple-view triangulation is a key problem in computer vision with numerous applications, such as 3D object reconstruction, map estimation, robotic path-planning, etc, see e.g. [1][2]. Triangulation can be performed by minimizing different norms of the reprojection error, and typically  $L_2$  triangulation is preferred.

Several contributions have been proposed in the literature for  $L_2$  triangulation. In [3][4] the authors show how the exact solution of triangulation with two-views can be obtained by computing the roots of a one-variable polynomial of degree six. For triangulation with three-views, the exact solution is obtained in [5] by solving a system of polynomial equations through methods from computational commutative algebra, and in [6] through Groebner basis techniques. Multiple-view triangulations is considered for example in [7] via branch-and-bound algorithms. Before proceeding it is also useful mentioning some contributions for triangulation with different norms, such as [8][9].

This paper proposes a new method for multiple-view  $L_2$  triangulation. This method exploits the fundamental matrices among the views and provides a candidate of the sought 3D point by solving an eigenvalue problem (EVP), which belongs to the class of convex programming. A straightforward condition for establishing the optimality of the found candidate is provided, which also yields a guaranteed range for the optimal cost of the triangulation problem in case of

non-optimality. The proposed method is illustrated through some well-known examples with real data, for which the provided candidate 3D point is always optimal. These examples also show that the computational time of the proposed method is indeed small and competitive with existing approaches.

The paper is organized as follows. Section 2 provides some preliminaries and the problem formulation. Section 3 describes the proposed method. Section 4 shows the results. Lastly, Section 5 concludes the paper with some final remarks.

## 2 Preliminaries

The notation adopted throughout the paper is as follows:

- $\mathbf{M}^T$ : transpose of matrix  $\mathbf{M} \in \mathbb{R}^{m \times n}$ ;
- $\mathbf{I}_n$ :  $n \times n$  identity matrix;
- $\mathbf{0}_n$ :  $n \times 1$  null vector;
- $\mathbf{e}_i$ :  $i$ -th column of  $\mathbf{I}_3$ ;
- $SO(3)$ : set of all  $3 \times 3$  rotation matrices;
- $SE(3)$ :  $SO(3) \times \mathbb{R}^3$ ;
- $\|\mathbf{v}\|$ : 2-norm of  $\mathbf{v} \in \mathbb{R}^n$ ;
- s.t.: subject to.

Let  $F_i = (\mathbf{R}_i, \mathbf{t}_i) \in SE(3)$  denote the coordinate frame of the  $i$ -th camera, where  $\mathbf{R}_i \in SO(3)$  is the rotation matrix and  $\mathbf{t}_i \in \mathbb{R}^3$  is the translation vector expressed with respect to a common reference coordinate frame  $F^{ref} \in SE(3)$ . A generic 3D point of the scene is expressed in homogeneous coordinates as

$$\mathbf{X} = (x, y, z, 1)^T \quad (1)$$

where  $x, y, z \in \mathbb{R}$  are expressed with respect to  $F^{ref}$ . The projection of  $\mathbf{X}$  onto the image plane of the  $i$ -th camera is given by

$$\mathbf{x}_i = (u_i, v_i, 1)^T \quad (2)$$

where  $u_i, v_i \in \mathbb{R}$  are the screen coordinates along the horizontal and vertical directions. The relation between  $\mathbf{X}$  and  $\mathbf{x}_i$  is expressed by the projective law

$$d_i \mathbf{x}_i = \mathbf{P}_i \mathbf{X} \quad (3)$$

where  $d_i \in \mathbb{R}$  and  $\mathbf{P}_i \in \mathbb{R}^{3 \times 4}$  is the projection matrix given by

$$\mathbf{P}_i = \mathbf{K}_i [\mathbf{R}_i, \mathbf{t}_i] \quad (4)$$

with  $\mathbf{K}_i \in \mathbb{R}^{3 \times 3}$  being the upper triangular matrix containing the intrinsic camera parameters of the  $i$ -th camera. The solution for  $\mathbf{x}_i$  in (3) is denoted by  $\Phi(\mathbf{X}, \mathbf{P}_i)$  and has the expression

$$\Phi(\mathbf{X}, \mathbf{P}_i) = \frac{\mathbf{P}_i \mathbf{X}}{\mathbf{e}_3^T \mathbf{P}_i \mathbf{X}}. \quad (5)$$

The multiple-view  $L_2$  triangulation problem is to determine the 3D point that minimizes the mean square re-projection error in  $L_2$  norm for given estimates of the image points and projection matrices, i.e.

$$\mathbf{X}^* = \arg \min_{\mathbf{X}} \sqrt{\frac{\sum_{i=1}^N \|\Phi(\mathbf{X}, \hat{\mathbf{P}}_i) - \hat{\mathbf{x}}_i\|^2}{2N}} \quad (6)$$

where  $N$  is the number of cameras,  $\hat{\mathbf{P}}_1, \dots, \hat{\mathbf{P}}_N$  are estimates of the projection matrices  $\mathbf{P}_1, \dots, \mathbf{P}_N$ , and  $\hat{\mathbf{x}}_1, \dots, \hat{\mathbf{x}}_N$  are estimates of the image points  $\mathbf{x}_1, \dots, \mathbf{x}_N$  corresponding to a common (unknown) 3D point.

### 3 Proposed Strategy

Let us define the optimal cost of the multiple-view  $L_2$  triangulation problem as

$$\mu^* = \sqrt{\frac{\sum_{i=1}^N \|\Phi(\mathbf{X}^*, \hat{\mathbf{P}}_i) - \hat{\mathbf{x}}_i\|^2}{2N}} \quad (7)$$

where  $\mathbf{X}^*$  is the solution of (6). The first step consists of rewriting  $\mu^*$  by using variables in the image domain rather than in the 3D space. We hence obtain

$$\begin{aligned} \mu^* &= \min_{\mathbf{x}_1, \dots, \mathbf{x}_N} \sqrt{\frac{\sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2}{2N}} \\ \text{s.t. } (\mathbf{x}_1, \dots, \mathbf{x}_N) &\in \left\{ \left( \Phi(\mathbf{X}, \hat{\mathbf{P}}_i), \dots, \Phi(\mathbf{X}, \hat{\mathbf{P}}_N) \right) \text{ for some } \mathbf{X} \right\} \end{aligned} \quad (8)$$

where the constraint denotes the set of admissible values for the variables  $\mathbf{x}_1, \dots, \mathbf{x}_N$ . Let us gather all scalar unknowns of the variables  $\mathbf{x}_1, \dots, \mathbf{x}_N$  into the vector

$$\mathbf{w} = (1, u_1, v_1, \dots, u_N, v_N)^T. \quad (9)$$

It follows that (8) can be rewritten as

$$\begin{aligned} \mu^* &= \min_{\mathbf{w}} \sqrt{\frac{\mathbf{w}^T \mathbf{G} \mathbf{w}}{2N}} \\ \text{s.t. } (\mathbf{x}_1, \dots, \mathbf{x}_N) &\in \left\{ \left( \Phi(\mathbf{X}, \hat{\mathbf{P}}_i), \dots, \Phi(\mathbf{X}, \hat{\mathbf{P}}_N) \right) \text{ for some } \mathbf{X} \right\} \end{aligned} \quad (10)$$

where  $\mathbf{G}$  is the matrix given by

$$\mathbf{G} = \sum_{i=1}^N \mathbf{G}_i^T \mathbf{G}_i \quad (11)$$

and  $\mathbf{G}_i$  is defined as

$$\mathbf{G}_i = \left( -\begin{pmatrix} \hat{u}_i \\ \hat{v}_i \end{pmatrix}, \mathbf{f}_i^T \otimes \mathbf{I}_2 \right) \quad (12)$$

being  $\mathbf{f}_i$  the  $i$ -th column of  $\mathbf{I}_N$ .

In order to solve (10), we introduce the following modified problem:

$$\begin{aligned} \mu_F^* &= \min_{\mathbf{w}} \sqrt{\frac{\mathbf{w}^T \mathbf{G} \mathbf{w}}{2N}} \\ \text{s.t. } & \mathbf{x}_i^T \hat{\mathbf{F}}_{i,j} \mathbf{x}_j = 0 \quad \forall i = 1, \dots, N-1, \quad j = i+1, \dots, N \end{aligned} \quad (13)$$

where  $\hat{\mathbf{F}}_{i,j} \in \mathbb{R}^{3 \times 3}$  is the estimate of the fundamental matrix between the  $i$ -th and the  $j$ -th view. This estimate can be simply computed from the estimates of the projection matrices  $\hat{\mathbf{P}}_i$  and  $\hat{\mathbf{P}}_j$  as described for example in [4]. Let us observe that the constraint of problem (13) is satisfied whenever the one of the original problem (10) is, hence implying that

$$\mu_F^* \leq \mu^*. \quad (14)$$

Therefore, the solution of the modified problem (13) may be different from the solution of (10) since the variables are allowed to vary in a possibly larger set. Nevertheless, we will see in Section 4 that such a case seldom occurs in practice.

Let us describe now the proposed strategy for solving (13). We introduce the quantity

$$\mu_- = \sqrt{\frac{\gamma_-}{2N}} \quad (15)$$

where  $\gamma_-$  is the solution of the optimization problem

$$\begin{aligned} \gamma_- &= \max_{\mathbf{Z}} -\text{tr}(\mathbf{Z} \mathbf{G}) \\ \text{s.t. } & \begin{cases} \mathbf{Z} \geq 0 \\ \mathbf{Z}_{1,1} = 1 \\ \text{tr}(\mathbf{Z} \mathbf{A}_k) = 0 \quad \forall k = 1, \dots, M \end{cases} \end{aligned} \quad (16)$$

where  $\mathbf{Z} \in \mathbb{R}^{2N+1, 2N+1}$  is a variable symmetric matrix, and  $\mathbf{A}_1, \dots, \mathbf{A}_M \in \mathbb{R}^{2N+1, 2N+1}$  are symmetric matrices such that

$$\mathbf{w}^T \mathbf{A}_k \mathbf{w} = \mathbf{x}_i^T \hat{\mathbf{F}}_{i,j} \mathbf{x}_j, \quad i = 1, \dots, N-1, \quad j = i+1, \dots, N. \quad (17)$$

This optimization problem provides a lower bound of  $\mu_F^*$ , indeed as shown in the Appendix one has that

$$\mu_- \leq \mu_F^*. \quad (18)$$

The optimization problem (16) belongs to the class of eigenvalue problems (EVPs), which are convex optimizations with linear matrix inequality constraints and linear matrix equality constraints. See e.g. [10] about EVPs. And see e.g. [11, 12, 13] for the use of EVPs in other areas of computer vision, such as the estimation of the fundamental matrix, the computation of worst-case positioning errors induced by image noise, and the estimation of the camera pose.

Once (16) is solved, one builds a candidate solution for the original problem (6) as follows. Let  $\bar{\mathbf{w}}$  be a vector with form as in (9) satisfying

$$\bar{\mathbf{Z}} \bar{\mathbf{w}} = \lambda_{\max}(\bar{\mathbf{Z}}) \bar{\mathbf{w}} \quad (19)$$

where  $\bar{\mathbf{Z}}$  is the matrix  $\mathbf{Z}$  evaluated at the optimum of (6). Let us extract  $\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_N$  from  $\bar{\mathbf{w}}$ . The candidate solution of the multiple-view  $L_2$  triangulation problem (6) is computed as

$$\hat{\mathbf{X}} = \min_{\mathbf{X}} \sum_{i=1}^N \left\| \hat{\mathbf{P}}_i \mathbf{X} - \bar{\mathbf{x}}_i \mathbf{e}_3^T \hat{\mathbf{P}}_i \mathbf{X} \right\|^2 \quad (20)$$

which is a linear least-squares minimization and can be simply solved. Indeed, one has that

$$\hat{\mathbf{X}} = \begin{pmatrix} \hat{\mathbf{Y}} \\ 1 \end{pmatrix} \quad (21)$$

where

$$\hat{\mathbf{Y}} = - \left( \sum_{i=1}^N \mathbf{Q}_i^T \mathbf{Q}_i \right)^{-1} \sum_{i=1}^N \mathbf{Q}_i \mathbf{r}_i \quad (22)$$

and  $\mathbf{Q}_i \in \mathbb{R}^{3 \times 3}$  and  $\mathbf{r}_i \in \mathbb{R}^3$  are such that

$$(\mathbf{I}_3 - \bar{\mathbf{x}}_i \mathbf{e}_3^T) \hat{\mathbf{P}}_i = (\mathbf{Q}_i, \mathbf{r}_i). \quad (23)$$

Finally, let us define

$$\mu_+ = \sqrt{\frac{\sum_{i=1}^N \|\Phi(\hat{\mathbf{X}}, \hat{\mathbf{P}}_i) - \hat{\mathbf{x}}_i\|^2}{2N}}. \quad (24)$$

Clearly,  $\mu_+$  is an upper bound of the solution of the original problem (6) since it is the cost function evaluated in a feasible point. Hence, at this point one has a lower and an upper bound of  $\mu^*$ :

$$\mu^* \in [\mu_-, \mu_+]. \quad (25)$$

From this, it is clearly possible to derive an immediate test for establishing whether  $\hat{\mathbf{X}}$  is the optimal solution of (6). Indeed:

$$\begin{aligned} \mu_- &= \mu_+ \\ &\Downarrow \\ \mu^* &= \mu_- \text{ and } \hat{\mathbf{X}} \text{ is optimal (i.e., solution of (6)).} \end{aligned} \quad (26)$$

As it will become clear in the next section, the cases where  $\hat{\mathbf{X}}$  is not optimal seem to be rare in practice. Moreover, when  $\hat{\mathbf{X}}$  is not optimal, the proposed strategy provides a lower and an upper bound of the solution of the original problem (6) as expressed by (25).

## 4 Results

In this section we present an illustrative example with synthetic data randomly generated, and two examples with real data available at the webpage of the Visual Geometry Group of Oxford University, <http://www.robots.ox.ac.uk/~vgg/data/data-mview.html>.

## 4.1 An Illustrative Example

Let us first consider an illustrative example with  $N = 5$  by randomly selecting the projection matrices and the image points as follows:

$$\hat{\mathbf{P}}_1 = \begin{pmatrix} 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 1 \\ 0 & 0 & 1 & 1 \end{pmatrix}, \quad \hat{\mathbf{P}}_2 = \begin{pmatrix} 1 & 1 & 1 & 0 \\ -1 & 1 & 1 & 0 \\ 1 & 0 & 0 & 1 \end{pmatrix}, \quad \hat{\mathbf{P}}_3 = \begin{pmatrix} 0 & 1 & -1 & 1 \\ 1 & 0 & 1 & 0 \\ -1 & 1 & 0 & 1 \end{pmatrix}$$

$$\hat{\mathbf{P}}_4 = \begin{pmatrix} 0 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 \\ 1 & 0 & 1 & 1 \end{pmatrix}, \quad \hat{\mathbf{P}}_5 = \begin{pmatrix} -1 & -1 & 0 & 0 \\ 1 & 0 & 0 & 1 \\ 0 & 1 & 1 & 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_i = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} \quad \forall i = 1, \dots, 5.$$

By solving the EVP (16) we find  $\mu_- = 0.452$ . From the vector  $\bar{\mathbf{w}}$  in (19) we extract the image points

$$\bar{\mathbf{x}}_1 = \begin{pmatrix} 0.499 \\ 0.389 \\ 1 \end{pmatrix}, \quad \bar{\mathbf{x}}_2 = \begin{pmatrix} -0.155 \\ 0.472 \\ 1 \end{pmatrix}, \quad \bar{\mathbf{x}}_3 = \begin{pmatrix} 0.081 \\ 0.346 \\ 1 \end{pmatrix}$$

$$\bar{\mathbf{x}}_4 = \begin{pmatrix} 0.094 \\ 0.685 \\ 1 \end{pmatrix}, \quad \bar{\mathbf{x}}_5 = \begin{pmatrix} 0.575 \\ 0.679 \\ 1 \end{pmatrix}.$$

We hence compute  $\hat{\mathbf{X}}$  in (21), finding  $\hat{\mathbf{X}} = (-0.239, -0.406, 0.527, 1)^T$ . The depths of  $\hat{\mathbf{X}}$  with respect to the five cameras are 1.527, 0.761, 0.833, 1.288 and 1.121. Then, we compute  $\mu_+$  in (21), obtaining  $\mu_+ = 0.452$ . Therefore, from (26) we conclude that the found solution is optimal, and

$$\mu^* = 0.452, \quad \mathbf{X}^* = (-0.239, -0.406, 0.527, 1)^T.$$

## 4.2 Examples with Real Data

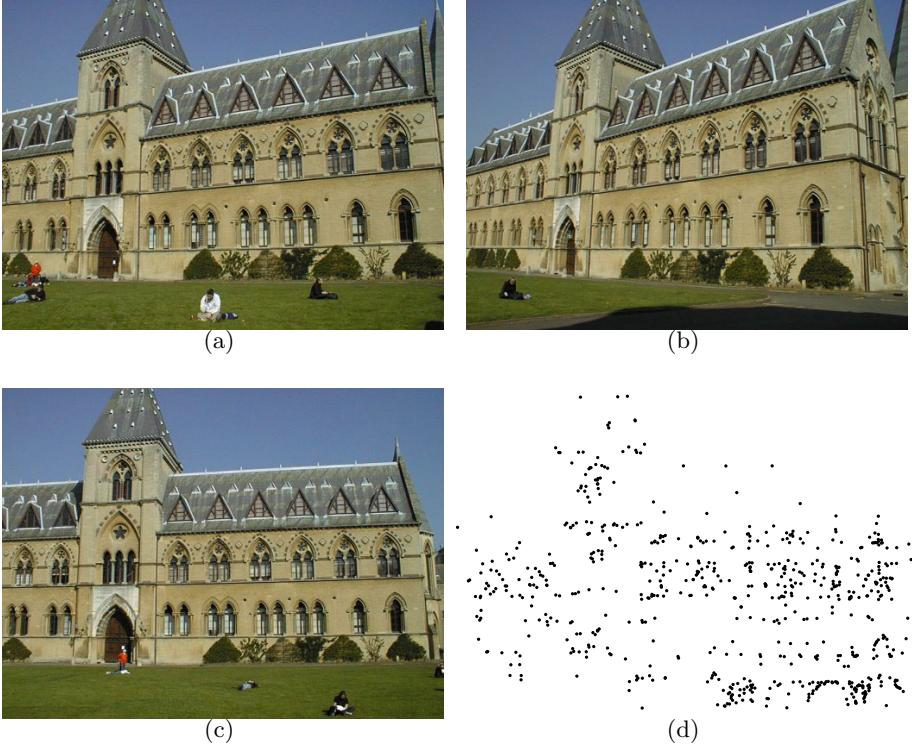
Here we consider two well-known examples with real data, in particular:

- the university library example, which is a sequence of 3 views with 667 3D points (Figure 1);
- the house example, which is a sequence of 10 views with 672 3D points (Figure 2).

Each 3D point is estimated by solving the triangulation problem with all the available views. The number of such views is shown in Tables 1–2. The solution provided by the proposed method for these examples is always optimal.

In order to provide some more details, let us consider a 3D point in the house example that is visible in  $N = 9$  views, e.g. with available image points given by

$$\hat{\mathbf{x}}_1 = \begin{pmatrix} 516.435 \\ 167.624 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_2 = \begin{pmatrix} 527.205 \\ 174.126 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_3 = \begin{pmatrix} 532.609 \\ 180.280 \\ 1 \end{pmatrix}$$



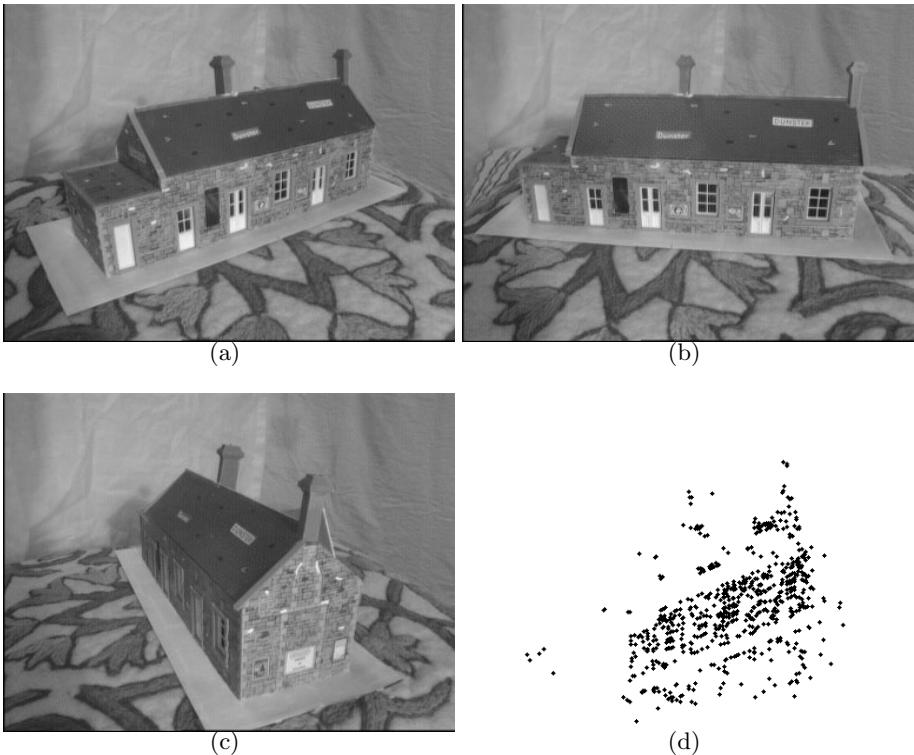
**Fig. 1.** University library example: images of the sequence and the reconstructed model by using for each 3D point all the available views

$$\begin{aligned}\hat{\mathbf{x}}_4 &= \begin{pmatrix} 533.905 \\ 186.198 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_5 = \begin{pmatrix} 529.289 \\ 193.379 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_6 = \begin{pmatrix} 519.414 \\ 199.970 \\ 1 \end{pmatrix} \\ \hat{\mathbf{x}}_7 &= \begin{pmatrix} 498.802 \\ 208.174 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_8 = \begin{pmatrix} 463.735 \\ 215.630 \\ 1 \end{pmatrix}, \quad \hat{\mathbf{x}}_9 = \begin{pmatrix} 429.465 \\ 219.676 \\ 1 \end{pmatrix}\end{aligned}$$

(the projection matrices are omitted for conciseness). By solving the EVP (16) we find  $\mu_- = 0.210$ . From the vector  $\bar{\mathbf{w}}$  in (19) we extract the image points  $\bar{\mathbf{x}}_1, \dots, \bar{\mathbf{x}}_9$ , from which we compute  $\hat{\mathbf{X}}$  in (21). We find  $\hat{\mathbf{X}} = (-1.110, 1.016, -5.653, 1)^T$ , whose depths with respect to the nine cameras range from 0.280 to 5.708. Then, we compute  $\mu_+$  in (24), obtaining  $\mu_+ = 0.210$ . Therefore, from (26) we conclude that the found solution is optimal, i.e.

$$\mu^* = 0.210, \quad \mathbf{X}^* = (-1.110, 1.016, -5.653, 1)^T.$$

The average computational time per point in these examples ranges from 0.02 seconds (case of 2 views) to 0.30 seconds (case of 10 views). This time is relative to an implementation of the proposed method in Matlab on a standard



**Fig. 2.** House example: some images of the sequence and the reconstructed model by using for each 3D point all the available views

**Table 1.** University library example: number of views  $N$  and number of 3D points with  $N$  views

	$N$	2	3
number of 3D points with $N$ views		585	82

**Table 2.** House example: number of views  $N$  and number of 3D points with  $N$  views

	$N$	3	4	5	6	7	8	9	10
number of 3D points with $N$ views		382	19	158	3	90	1	12	7

personal computer with Intel Pentium 4, 3 GHz, 2 GB RAM, and Windows XP. This computational time appears reasonably small and competitive with existing approaches, in particular with [7] which reports one of the fastest existing method with an average computational time of 0.02 seconds (time relative to an implementation in C language).

## 5 Conclusion

This paper has proposed a new method for multiple-view  $L_2$  triangulation which requires to solve an EVP and which provides a candidate of the sought 3D point. The proposed method has been illustrated through some well-known examples with real data, for which the provided candidate 3D point is always optimal. These examples have also shown that the computational time of the proposed method is indeed small and competitive with existing approaches. Future work will be devoted to investigate further properties of the proposed method.

## Acknowledgement

The work in this paper was supported in part by the CRCG of the University of Hong Kong (Project 200907176048) and the Research Grants Council of Hong Kong (GRF Projects HKU 712808E and HKU 711208E).

## References

1. Faugeras, O., Luong, Q.T.: *The Geometry of Multiple Images*. MIT Press, Cambridge (2001)
2. Chesi, G., Hashimoto, K. (eds.): *Visual Servoing via Advanced Numerical Methods*. Springer, Heidelberg (2010)
3. Hartley, R., Sturm, P.: Triangulation. *Computer Vision and Image Understanding* 68, 146–157 (1997)
4. Hartley, R., Zisserman, A.: *Multiple view in computer vision*. Cambridge University Press, Cambridge (2000)
5. Stewenius, H., Schaffalitzky, F., Nister, D.: How hard is 3-view triangulation really? In: Int. Conf. on Computer Vision, pp. 686–693 (2005)
6. Byrod, M., Josephson, K., Astrom, K.: Fast optimal three view triangulation. In: Asian Conf. on Computer Vision, Tokyo, Japan (2007)
7. Lu, F., Hartley, R.: A fast optimal algorithm for  $l_2$  triangulation. In: Asian Conf. on Computer Vision, Tokyo, Japan (2007)
8. Hartley, R., Schaffalitzky, F.:  $l_\infty$  minimization in geometric reconstruction problems. In: IEEE Conf. on Computer Vision and Pattern Recognition, pp. 504–509 (2004)
9. Ke, Q., Kanade, T.: Uncertainty models in quasiconvex optimization for geometric reconstruction. In: IEEE Conf. on Computer Vision and Pattern Recognition (2006)
10. Boyd, S., El Ghaoui, L., Feron, E., Balakrishnan, V.: *Linear Matrix Inequalities in System and Control Theory*. SIAM, Philadelphia (1994)
11. Chesi, G., Garulli, A., Vicino, A., Cipolla, R.: Estimating the fundamental matrix via constrained least-squares: a convex approach. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 24, 397–401 (2002)
12. Chesi, G., Hung, Y.S.: Image noise induced errors in camera positioning. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 29, 1476–1480 (2007)
13. Chesi, G.: Camera displacement via constrained minimization of the algebraic error. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 31, 370–375 (2009)

## Appendix

Here we prove that (18) holds. Suppose that the constraints in (16) are satisfied, and let us define

$$\mathbf{B}(c, \mathbf{d}) = c\mathbf{E} + \sum_{k=1}^M d_k \mathbf{A}_k \quad (27)$$

where  $c \in \mathbb{R}$  and  $\mathbf{d} \in \mathbb{R}^M$  are variables, and  $\mathbf{E} \in \mathbb{R}^{2N+1, 2N+1}$  is defined as

$$(\mathbf{E})_{i,j} = \begin{cases} -1 & \text{if } i = j = 1 \\ 0 & \text{otherwise.} \end{cases} \quad (28)$$

We have that

$$\text{tr}(\mathbf{Z}\mathbf{B}(c, \mathbf{d})) + \text{tr}(\mathbf{Z}\mathbf{G}) \geq 0 \quad (29)$$

for all  $c, \mathbf{d}$  such that

$$\mathbf{B}(c, \mathbf{d}) + \mathbf{G} \geq 0. \quad (30)$$

Consider now any  $c, \mathbf{d}$  such that (30) holds (clearly, such  $c, \mathbf{d}$  exist, e.g. given by  $c = 0$  and  $\mathbf{d} = \mathbf{0}_M$ ). Consider any  $\mathbf{w} \in \mathbb{R}^{2N+1}$  with the form in (9) and such that

$$\mathbf{x}_i^T \hat{\mathbf{F}}_{i,j} \mathbf{x}_j = 0 \quad \forall i = 1, \dots, N-1, \quad j = i+1, \dots, N. \quad (31)$$

Let us pre- and post-multiply the LHS of (30) times  $\mathbf{w}^T$  and  $\mathbf{w}$ , respectively. We obtain that

$$-c + \mathbf{w}^T \mathbf{G} \mathbf{w} \geq 0 \quad (32)$$

hence implying that

$$c \leq \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2. \quad (33)$$

Moreover, from (29) one has

$$c + \text{tr}(\mathbf{Z}\mathbf{G}) \geq 0 \quad (34)$$

which implies that

$$-\text{tr}(\mathbf{Z}\mathbf{G}) \leq c. \quad (35)$$

Therefore,

$$-\text{tr}(\mathbf{Z}\mathbf{G}) \leq \sum_{i=1}^N \|\mathbf{x}_i - \hat{\mathbf{x}}_i\|^2 \quad (36)$$

and we finally conclude that (18) holds.

# An Improved Shape Matching Algorithm for Deformable Objects Using a Global Image Feature

Jibum Kim and Suzanne M. Shontz

Department of Computer Science and Engineering  
The Pennsylvania State University  
University Park, PA 16802  
[{jzk164, shontz}@cse.psu.edu](mailto:{jzk164, shontz}@cse.psu.edu)

**Abstract.** We propose an improved shape matching algorithm that extends the work of Felzenszwalb [3]. In this approach, we use triangular meshes to represent deformable objects and use dynamic programming to find the optimal mapping from the source image to the target image which minimizes a new energy function. Our energy function includes a new cost term that takes into account the center of mass of an image. This term is invariant to translation, rotation, and uniform scaling. We also improve the dynamic programming method proposed in [3] using the center of mass of an image. Experimental results on the Brown dataset show a 7.8% higher recognition rate when compared with Felzenszwalb's algorithm.

## 1 Introduction

Shape matching is an important problem in many computer vision applications such as object tracking and image-based searches [12]. The goal of shape matching is to match the source image to the target image, i.e., the deformed image. Felzenszwalb proposed a shape matching algorithm for deformable objects using triangular meshes and dynamic programming in [3]. Felzenszwalb's algorithm was novel in that it does not require any initialization to detect the target image unlike previous algorithms (e.g., [4]). A modification of Felzenszwalb's algorithm using flexible shape priors was proposed in [5]. In [5], large deformations on flexible regions were allowed through the use of the shape priors. However, it is hard to know which parts should be made flexible in advance, and thus significant user knowledge is required. Moreover, [3,5] do not use global image features to detect deformable objects.

Recently, several papers have used global image features for shape matching. For example, hierarchy-based shape matching algorithms for deformable objects were proposed in [6,7]. In [6], the authors identified shape parts, composed of multiple salient points, and used many-to-many matching for shape matching. The authors in [7] used a shape tree to capture both local and global shape information and employed dynamic programming to perform shape matching.

In this paper, we use local and global shape information to perform shape matching in a complementary method compared with that of [7]. In particular, we use triangular meshes and dynamic programming for shape matching, as in [3][5], and extend the algorithm in [3] through the use of an added global image feature. The rest of this paper is organized as follows. In Section 2, we describe the three-step shape matching process. In Section 3, we present experimental results, and compare our experimental results with those of algorithm in [3] (i.e., the algorithm to which ours is the most similar). Finally, we give some conclusions and plans for future work in Section 4.

## 2 Shape Matching Process

Our shape matching process is composed of three steps. The first step is to determine boundary vertices which approximate the source image boundary. The second step is to generate a triangular mesh using the constrained Delaunay triangulation method to represent the deformable object. The third step is to find the optimal mapping from the source image to the target image which minimizes our energy function. We now describe each of the three steps in more detail.

### 2.1 Determination of the Boundary Vertices Approximating the Source Image Boundary

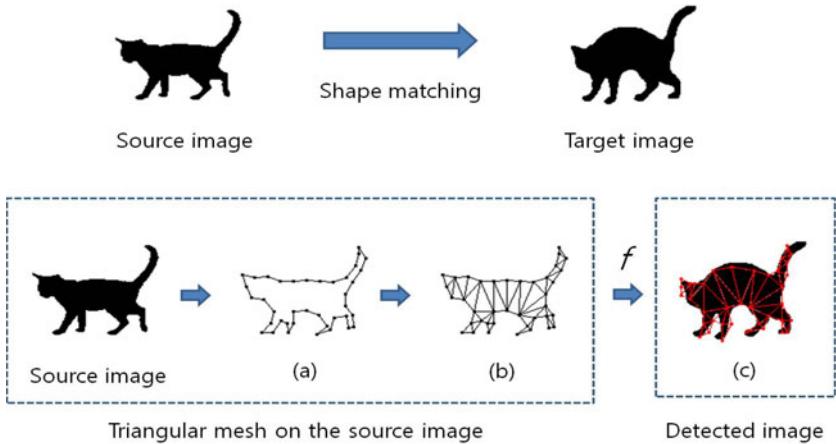
This step determines the number of boundary vertices to use in the source image. In order to find a polygonal approximation to a boundary curve in the source image,  $S$ , we create nearly equally-spaced vertices on the boundary of  $S$ . We select vertices such that the distance between them is as close to an ideal distance parameter as possible, and the boundary curve is close to the original curve. This step is illustrated in Figure II(a).

### 2.2 Generation of the Triangular Mesh on the Source Image Using the Constrained Delaunay Triangulation Method

This step combines the boundary vertices into triangles to represent the non-rigid objects. The constrained Delaunay triangulation method [8] implemented in Triangle [9] is used to generate a triangular mesh,  $M$ , that respects the boundary of  $S$ , without adding any interior vertices, to represent the deformable parts of the image. This step is shown in Figure II(b). Note that a triangular mesh without interior vertices can be represented using a dual graph. The vertices in the dual graph of  $M$  can be ordered and eliminated using a perfect elimination scheme [10].

### 2.3 Solution of the Shape Matching Problem

In order to determine the optimal placement of the boundary vertices on a target image,  $T$ , we formulate and solve an optimization problem with the goal of determining the mapping,  $f$ , from the triangles in  $S$  to the triangles in  $T$ , which



**Fig. 1.** Overview of the shape matching process. The function  $f$  maps triangles in the triangular mesh on the source image to a triangular mesh on the target image. (a) Equally-spaced boundary vertices are generated. (b) The triangular mesh is created. (c) The detected image is illustrated on the target image.

has the lowest energy. Unlike the energy (cost) functions in [35], our energy function is composed of three terms: an edge cost, a triangular deformation cost, and a triangular edge length distortion cost. The edge cost of the  $i^{th}$  triangle,  $E_{edge,i}$ , corresponds to a simple edge detector by assigning high costs to a low image gradient magnitude in  $T$ . Thus,  $E_{edge,i}$  increases if edges detected on  $T$  are not placed on the boundary. The edge cost of the  $i^{th}$  triangle is given by

$$E_{edge,i} = \frac{1}{\lambda + |\nabla I|},$$

where  $\lambda$  is a constant.

The triangular deformation cost,  $E_{def,i}$ , represents how far the original triangle is transformed from a similarity transform [11] when the mapping from the source image to the target image occurs. The affine transformation of each triangle from  $S$  to  $T$  takes a unit circle to an ellipse with major and minor axes of lengths,  $\alpha$  and  $\beta$ , respectively. Here,  $\alpha$  and  $\beta$  are the singular values of the matrix associated with the affine transformation. The squared log-anisotropy of  $\alpha$  and  $\beta$  is used to measure the triangular deformation as in [3]. The triangular deformation cost of the  $i^{th}$  triangle is given by

$$E_{def,i} = \log^2 \left( \frac{\alpha}{\beta} \right).$$

Unlike [35], we introduce a new cost, the triangular edge length distortion cost,  $E_{dis,i}$ , which penalizes the sum of the edge length distortions of each triangle.

Let  $l_{j,s}$  and  $l_{j,t}$  be the  $j^{th}$  edge lengths of a triangle in  $S$  and  $T$ , respectively. Then, the sum of the three edge lengths of the triangle in  $S$  and  $T$ , respectively, are given by

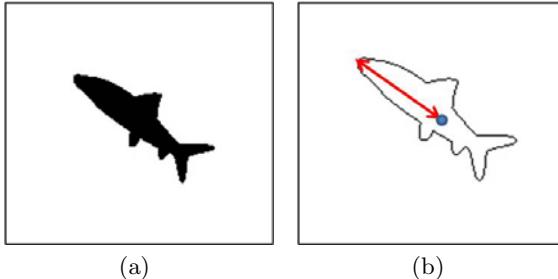
$$l_{sum,s} = \sum_{j=1}^3 l_{j,s}, \quad l_{sum,t} = \sum_{j=1}^3 l_{j,t}.$$

Then,  $\gamma$  is defined as

$$\gamma = \begin{cases} l_{sum,s}/l_{sum,t} & \text{if } l_{sum,s} > l_{sum,t} \\ l_{sum,t}/l_{sum,s} & \text{if } l_{sum,t} > l_{sum,s}. \end{cases}$$

To make the triangular edge length distortion cost invariant to uniform scaling from the source image to the target image, we use the center of mass of an image. Let  $d_{max,s}$  and  $d_{max,t}$  be the maximum distances from the center of masses to the boundary vertices in  $S$  and  $T$ , respectively. Figure 2 illustrates  $d_{max,s}$  and  $d_{max,t}$  for a fish shape in the Brown dataset. Then,  $\delta$  is defined as follows:

$$\delta = \begin{cases} d_{max,s}/d_{max,t} & \text{if } d_{max,s} > d_{max,t} \\ d_{max,t}/d_{max,s} & \text{if } d_{max,s} < d_{max,t}. \end{cases}$$



**Fig. 2.** (a) A sample image shape from the Brown dataset [2]. (b) The dot in the middle represents the center of mass for the image (this is denoted as  $C$  in the target image), and the arrow represents the maximum distance from the center of mass in the image to the boundary vertices (i.e.,  $d_{max,s}$  or  $d_{max,t}$ ).

The triangular edge length distortion cost,  $E_{dis,i}$ , is large when  $\delta$  and  $\gamma$  are different. This occurs when the transformation of the triangles from the source image to the target image does not correspond to uniform scaling. Finally,  $E_{dis,i}$  is defined as follows:

$$E_{dis,i} = \begin{cases} \log(\gamma/\delta) & \text{if } \gamma > \delta \\ \log(\delta/\gamma) & \text{if } \gamma < \delta. \end{cases}$$

The energy function,  $E(f)$ , is defined as a weighted sum of the three cost terms over the  $N$  triangles and is given by

$$E(f) = \sum_{i=1}^N E_i = \sum_{i=1}^N (E_{edge,i} + a * E_{def,i} + b * E_{dis,i}), \quad (1)$$

where,  $a$  and  $b$  are scalars. Note that small values of  $a$  and  $b$  are desirable for highly deformable objects. Let  $n$  be the number of vertices in  $M$ . Our goal is to determine the  $f$  that minimizes  $E(f)$ . To solve the optimization problem, the shape matching algorithm uses a dynamic programming method given in Algorithm 1. For computational efficiency, a discrete set of grid locations,  $G$ , in  $T$  is assumed. In line 3 of Algorithm 1, vertices  $i$  and  $j$  are the parents of the  $k^{th}$  vertex. The vertices in  $M$  are eliminated in order using a perfect elimination scheme [10].

The optimal matching image (detected mesh) with the lowest possible energy is found using the dynamic programming method shown in Algorithm 1. Note that, for some cases, matching based on the smallest energy does not guarantee a good match. For those cases, we choose the matching image with the next smallest energy. In order to choose the criteria for the sub-optimal cases, we employ the center of mass of an image, which is a global feature of an image. Let the center of mass in  $T$  be  $C$  and the center of the matched vertices in  $T$  be  $\hat{C}$ . We define  $D$  as follows:

$$D = \|C - \hat{C}\|_2.$$

---

**Algorithm 1.** *Shape Matching Algorithm : Dynamic Programming*


---

```

for  $k = 1$  to  $n - 2$  do
    Eliminate the  $k^{th}$  vertex in  $M$ 
     $\{i, j\} \leftarrow \text{parent}(k)$ 
    for  $p, q \in G$  do
         $V[i, j](p, q) \leftarrow \min_{r \in G} E(i, j, k, p, q, r) + V[i, k](p, q) + V[j, k](p, q)$ 
        Choose  $p, q \in G$  minimizing  $V[n - 1, n](p, q)$ . Trace back to find the optimal
        mapping for other vertices and compute  $\hat{C}$ .
    end for
end for
while  $D > \theta$  do
    Choose new  $p, q \in G$  with the next smallest  $V[n - 1, n](p, q)$ . Trace back to find
    the optimal mapping for other vertices and compute  $\hat{C}$ .
end while

```

---

If  $D$  is greater than a threshold value of  $\theta$ , it is likely that the detected mesh on  $T$  is poor as shown in Figure 3. The threshold value,  $\theta$ , can be defined using  $d_{max,t}$ , the maximum distance from  $C$  to the boundary vertices in  $T$ . In Figure 3,  $\theta$  is set to  $d_{max,t}/3$ . In this case, we select instead the next smallest energy, and find a new optimal mapping (i.e., by executing the while statement in Algorithm 1) until  $\hat{C}$  is such that

$$D \leq \theta. \quad (2)$$

Source image	Detected mesh on the target image	
	Felzenszwalb's algorithm [3]	Our algorithm
		

**Fig. 3.** Poor shape matching result of the algorithm in [3] (left) and improved matching result using the center of mass in the target image (right). For these figures, the detected mesh is illustrated on the target image. For this experiment,  $\theta = d_{max,t}/3$ .

### 3 Experiments

This section describes the experimental evaluation of our shape matching algorithm and provides a comparison of our results with those in [3], i.e., the shape matching algorithm which is the most similar to ours. We use the well-known Brown [12] dataset to test our algorithm. The Brown dataset has 99 images divided into 9 categories (each of which has 11 images). The Brown dataset is challenging because it includes both occluded images and images with missing parts. Sample images from the Brown dataset are illustrated in Figure 4. We use the standard evaluation method to test our algorithm. For each image, we count how many of the 10 best matches (i.e., as defined by the smallest cost) belong to the same category. For our algorithm, the detected mesh with the smallest cost must also satisfy (2). We use 25 pixels for the distance between successive boundary vertices. The grid size in the target images is set to 30x30. We use  $a=4b$  in (1). The threshold value  $\theta$  is set to  $d_{max,t}/3$ . Our experimental results show that choosing a value of  $\theta$  between  $d_{max,t}/3$  and  $d_{max,t}/2$  gives good matching results.



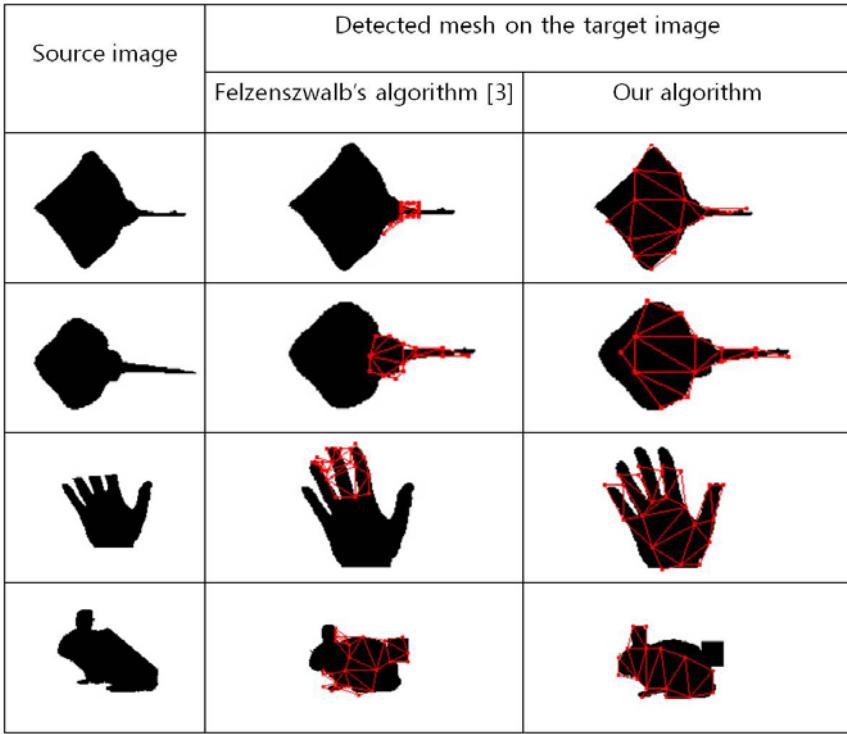
**Fig. 4.** Sample images in the Brown dataset [12]. Three sample images are shown per category.

Source image	Matching Results									
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th

**Fig. 5.** Good shape matching results for the Brown dataset on three source (query) images and comparison with [3]. For each source image, the 10 best matching results are shown with the smallest (left) to the largest (right) energy. The top figures in each group represent the matching results obtained from our algorithm, whereas the bottom figures in each group represent matching results using the algorithm in [3]. For these experimental sets, only two matching results of [3] (i.e., the bottom right images) fail to match.

Source image	Matching Results									
	1st	2nd	3rd	4th	5th	6th	7th	8th	9th	10th

**Fig. 6.** Poor shape matching results for the Brown dataset on three source (query) images and comparison with [3]. For each source image, the 10 best matching results are shown with the smallest (left) to the largest (right) energy. The top figures in each group represent the matching results obtained from our algorithm, and the bottom figures in each group represent matching results using the algorithm in [3]. For this experimental data set, both our algorithm and [3] show poor matching results. However, our algorithm shows better matching results than does the method in [3].



**Fig. 7.** Example matching results including detected meshes on the target image using images from the Brown dataset [12]. For this experiment, Felzenszwalb's algorithm shows poor matching results because triangles are placed at poor positions.

**Table 1.** Recognition rate results on the Brown dataset

Method	Recognition Rate
Felzenszwalb's algorithm [3]	64.4 %
Our algorithm	72.2 %

Figure 5 shows some of the good matching results obtained with our algorithm. For these source images, both our algorithm and Felzenszwalb's algorithm show good matching results. However, our algorithm shows slightly better matching results. Felzenszwalb's algorithm shows incorrect matching results for some cases (e.g., the bottom right images). Interestingly, we see that our algorithm and Felzenszwalb's algorithm show different rank orderings among the 10 best matching results. Figure 6 shows some poor matching results for our matching algorithm on the Brown data set. For these images, Felzenszwalb's algorithm fails in most cases and succeeds in only a few instances.

The recognition rates comparing our algorithm with Felzenszwalb's algorithm are shown in Table 1. The recognition rate is defined as the ratio of the total number of correct hits to the total number of correct hits possible [7]. For the

Brown data set, the total number of correct hits possible is 99\*10 since there are 99 images, and we find the 10 best matches. Our algorithm yields a 7.8% higher recognition rate when compared with Felzenszwalb's algorithm on the Brown data set. Several matching results, which include detected meshes on the target image, are shown in Figure 7. We observe that the algorithm in [3] produces poor matching results when detected meshes (triangles) are placed in poor positions within a target image. This often results in a large  $D$  value. This occurs because [3] matches the target image only by considering the local properties of each triangle such as shape similarity and the edge boundary. However, our algorithm produces better matching results by using global image features as shown in Figure 7.

## 4 Conclusions and Future Work

We have proposed a shape matching algorithm for deformable objects using both local and global shape information. In particular, we employ the center of mass of an image as a global image feature. Similar to the algorithm in [3], our algorithm does not need initialization to detect deformable objects. Experimental results show that Felzenszwalb's algorithm [3] sometimes produces poor matching results because it only considers the local properties of each triangle such as shape similarity and the edge boundary. However, our method produces improved matching results using a new energy function term and improved dynamic programming based upon global image features. Experimental results on the Brown dataset show a 7.8% higher recognition rate when compared to Felzenszwalb's algorithm. To further improve the recognition rate, we plan to use the symmetry of an image for deformable shape matching as in [13].

## Acknowledgements

The authors have benefitted from helpful conversations with Robert Collins of The Pennsylvania State University. The work of the second author was funded in part by NSF grant CNS 0720749 and an Institute for CyberScience grant from The Pennsylvania State University.

## References

- Yilmaz, A., Javed, O., Shah, M.: Object Tracking: A Survey. *ACM Comput. Surv.* 38(4), 1–45 (2006)
- Yeh, T., Tollmar, K., Darrell, T.: Searching the Web with Mobile Images for Location Recognition. In: IEEE CVPR, pp. 76–81 (2004)
- Felzenszwalb, P.F.: Representation and Detection of Deformable Shapes. *IEEE PAMI* 27(2), 208–220 (2005)
- Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active Contour Models. *Int. J. Comput. Vision* 1(4), 321–331 (1988)

5. Chang, T., Liu, T.: Detecting Deformable Objects with Flexible Shape Priors. In: IEEE ICPR, pp. 155–158 (2004)
6. Payet, N., Todorovic, S.: Matching hierarchies of deformable shapes. In: GBR, pp. 1–10 (2009)
7. Felzenszwalb, P.F., Schwartz, J.D.: Hierarchical matching of deformable shapes. In: IEEE CVPR, pp. 1–8 (2007)
8. Seidel, R.: Constrained Delaunay Triangulations and Voronoi Diagrams with Obstacles, Technical Report 260, Inst. for Information Processing, Graz, Austria, pp. 178–191 (1988)
9. Shewchuk, J.R.: Triangle: Engineering a 2D Quality Mesh Generator and Delaunay Triangulator. In: Lin, M.C., Manocha, D. (eds.) FCRC-WS 1996 and WACG 1996. LNCS, vol. 1148, pp. 203–222. Springer, Heidelberg (1996)
10. Golumbic, M.C.: Algorithmic Graph Theory and Perfect Graphs. Academic Press, New York (1980)
11. Widrow, B.: The rubber mask technique. Pattern Recognition 5(3), 174–211 (1973)
12. Sebastian, T.B., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing their shock graphs. IEEE PAMI 26(5), 550–557 (2004)
13. Lee, S., Liu, Y.: Symmetry-Driven Shape Matching. Penn State Technical Report CSE 9(11), 1–22 (2009)

# Multi-scale Topo-morphometric Opening of Arteries and Veins: An Evaluative Study via Pulmonary CT Imaging

Zhiyun Gao<sup>1</sup>, Colin Holtze<sup>2</sup>, Randall Grout<sup>2</sup>, Milan Sonka<sup>1</sup>,  
Eric Hoffman<sup>2</sup>, and Punam K. Saha<sup>1,2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering

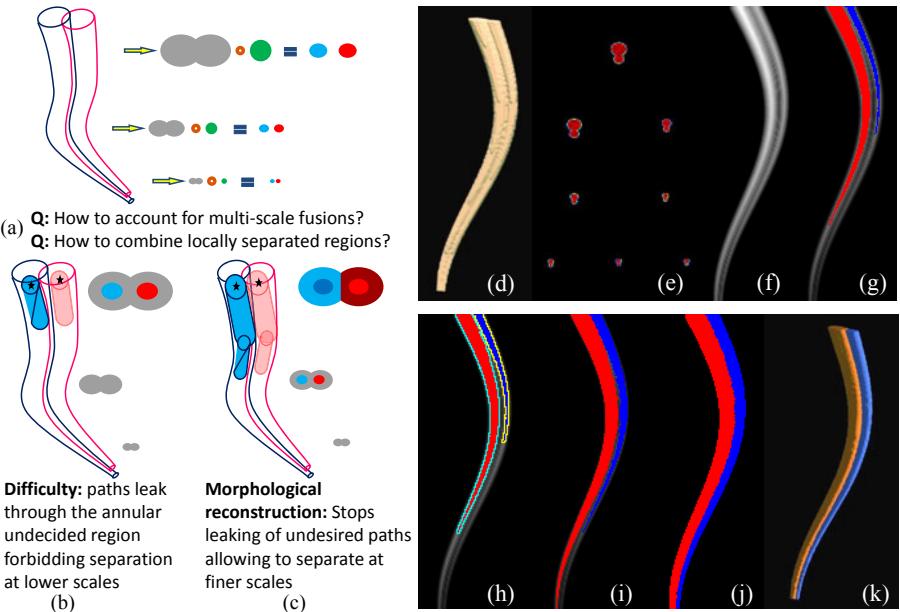
<sup>2</sup> Department of Radiology

The University of Iowa, Iowa City, Iowa 52242

**Abstract.** Distinguishing pulmonary arterial and venous (A/V) trees via *in vivo* imaging is essential for quantification of vascular geometry useful to diagnose several pulmonary diseases. A multi-scale topo-morphologic opening algorithm has recently been introduced separating A/V trees via non-contrast CT imaging. The method starts with two sets of seeds — one for each of A/V trees and combines fuzzy distance transform, fuzzy connectivity, and morphologic reconstruction leading to locally-adaptive multi-scale opening of two mutually fused structures. Here, we present results of a comprehensive validation study assessing both reproducibility and accuracy of the method. Accuracy of the method is examined using both mathematical phantoms and CT images of contrast-separated pulmonary A/V casting of a pig’s lung. Reproducibility of the method is evaluated using multi-user A/V separations of patients’s CT pulmonary data and contrast-enhanced CT data of a pig’s lung at different volumes. The qualitative and quantitative results are very promising.

## 1 Introduction

Over the last few decades, availability of a wide spectrum of medical imaging techniques [1] including MR, ultrasound, CT, PET, X- and  $\gamma$ -rays, has rapidly grown enhancing needs for computerized algorithms to extract quantitative measures from acquired images. Segmentation is a critical task in many quantitative medical imaging applications and, often, the segmentation problem of an individual application is unique in nature. Here, we address the problems related to separating pulmonary artery/vein (A/V) trees via CT imaging with no contrast. Challenges in A/V separation are multi-folded including: (1) A/V are indistinguishable by their intensity values in non-contrast CT images, (2) lack of intensity variation at sites of A/V fusion, (3) tight and complex A/V coupling or fusion with arbitrary and multi-scale geometry, especially at branching locations, and (4) limited signal to noise ratio (SNR) and relatively low resolution are typical for *in vivo* CT imaging. Patient-specific structural abnormalities of vascular trees further complicate the task. Several works have been reported in



**Fig. 1.** (a-c) A schematic description of fundamental challenges in A/V separation and their solutions using a multi-scale topo-morphologic opening approach. Hollow dots in (a) denote morphological erosion. (d-k) Results of intermediate steps in the multi-scale topo-morphologic opening; see text for details.

literature addressing A/V separation using improvised image acquisition techniques; a thorough discussion on difficulties of such approaches, especially for smaller vessels, have been presented by Bemmel *et al.* [2]. To the best of our knowledge, only a few post-processing methods have been published for separating arteries and veins [2,3,4,5]. Two of the previous methods [2,3] have only been applied to MR data and did not use morphological scale information. These methods primarily rely on intensity variations or presence of edge information at adhering locations between A/V trees and may not work for *in vivo* CT images where no such intensity variations are present at locations of adherence. Two methods for A/V separation from pulmonary CT images have been presented by Büelow *et al.* [4] and Yonekura *et al.* [5]. Büelow *et al.*'s method for A/V separation is based on prior knowledge of airway tree segmentation which may not produce optimal A/V separation at distal branches in a pulmonary vascular tree. The method by Yonekura *et al.* [5] is based on specific anatomical features of pulmonary A/V trees and prior airway segmentation. A major difficulty with such methods is the non-uniformity of different features over wide range of scales in a pulmonary tree and may not generalize to A/V separation in other body regions.

Separating pulmonary A/V trees via CT imaging is a critical first step in the quantification of vascular geometry for purposes of determining, for instance,

pulmonary hypertension, pulmonary emboli and more. Separation of A/V trees may also be useful to enhance airway tree segmentation based on the relationship of artery and airway, and to provide landmarks for intra- and inter-subject pulmonary data registration. Absence of contrast agents and tight spatial coupling between A/V trees leads to mutual fusion at limited resolution regime of CT imaging significantly enhancing the task of A/V separation. Often, the intensity variations at fusions may not be a reliable feature to separate the two structures. On the other hand, the two structures may frequently be locally separable using a morphological opening operator of suitable scale not known *a priori*.

Recently, a new multi-scale topo-morphometric opening algorithm has been developed to separate A/V trees via non-contrast CT imaging. Although, the method has been applied to separate A/V trees via pulmonary CT imaging, it may also be useful in other applications including tracking living cells in optical microscopic video imaging [6] or tracking moving objects/subjects in video imaging where multiple entities get mutually partially occluded from time to time. In this paper, we examine both accuracy and reproducibility of the method. Accuracy of the method is studied using mathematical phantoms as well as CT images of pulmonary vessel casting of a pig's lung. Reproducibility of the method is examined using results of multi-user A/V separation for non-contrast *in vivo* pulmonary CT images of patients as well as for contrast-enhanced CT images of a pig's lung at different lung volumes.

## 2 Methods and Experimental Plans

We briefly describe the idea and basic steps of the multi-scale topo-morphologic opening (MSTMO) algorithm for A/V separation [7]. Also, we present experimental methods and plans evaluating its accuracy and reproducibility.

### 2.1 Artery/Vein Separation Algorithm

Here, A/V trees are modeled as two similar-intensity tubular tree structures with significant overlaps at various locations and scales. The A/V separation task is solved using a MSTMO algorithm that starts with fuzzy segmentation of the assembly of A/V trees and two sets of seed points, one for each tree. Finally, the method outputs spatially separated A/V trees. It assumes that fusions of arteries and veins are locally separable using a suitable morphological operator. The method solves two fundamental challenges: (1) how to find the scale of local morphological operators and (2) how to trace continuity of locally separated regions. These challenges are met by combining fuzzy distance transform (FDT) [8], a morphologic feature, with topologic fuzzy connectivity [9][10][11] and morphological reconstruction to iteratively open finer and finer details starting at large scales and progressing towards smaller ones.

Let us consider two similar-intensity cylindrical objects with significant mutual overlap as illustrated in Figure 1(a). Often, intensity variations at fusions may not be a reliable feature to separate the two structures. On the other hand,

the two structures may frequently be locally separable using a suitable morphological erosion. The questions are how to determine the suitable local size of the operator and how to combine locally separated regions. First, the algorithm computes normalized FDT image of the conjoined structure using local scale to reduce the effect of spatial scale variation. Local scale at a point  $p$  is defined as the FDT value of the locally-deepest point (a point with locally maximum FDT value) that is nearest to  $p$ . The separation process is initiated by picking two seeds, one for each object, as shown in Figure 1(b) and continues by exploring connectivity on the FDT image. Using the knowledge that the two seeds represent two different objects, a threshold may be selected on FDT image that barely disconnects the two objects. Essentially, the FDT threshold indicates the radius of the optimal erosion operator partially separating the two cylinders. The immediate next question is how to proceed with separation to the next lower scales. The difficulty here is that the scale of the annular remainder after erosion is at least equal to that of the regions not yet separated. This difficulty is overcome using a morphological reconstruction operation that fills the annular remainder while maintaining separate identities of the two objects (Figure 1(c)). This step allows proceeding with separations at a lower scale and the method progresses iteratively separating fused structures at finer scales. In short, the method may be summarized in the following four steps:

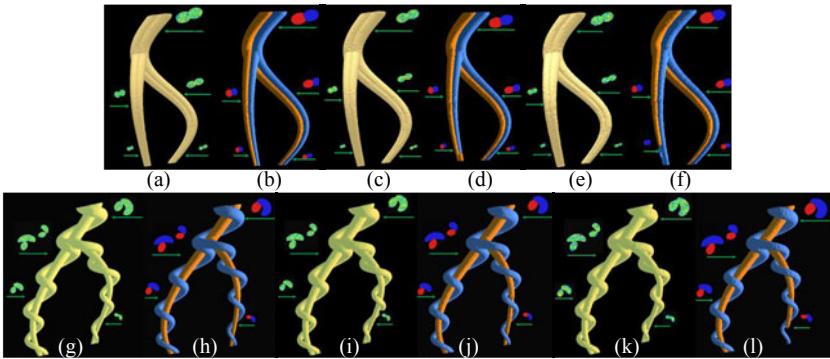
**Step 1:** Compute normalized FDT of the assembly of two conjoined structures.

**Step 2:** Find the optimum morphological operator scale separating the two objects specified by two sets of seeds.

**Step 3:** Block the annular remainder around currently separated regions using morphological reconstruction to proceed to finer scales.

**Step 4:** Repeat Steps 2 & 3 until convergence.

Results of different intermediate steps of the algorithm on a mathematical phantom with two similar-intensity cylinders with multi-scale fusions are presented in Figures 1(d-k). Figure 1(d) shows a 3D rendition of the phantom while a few cross-sectional images are presented in Figure 1(e); the diameter of one cylinder is significantly larger than that of the other. One seed is identified near the center of each cylinder at the top-most slice. To illustrate the functionality of intermediate steps we show the results on a 2D plane containing the central line of both cylinders. Results of normalized FDT computation (Step 1) and separation of two cylinders using connectivity analysis (Step 2) are shown in Figures 1(f) and (g), respectively. Step 3 builds a separator between the two objects using morphological reconstruction that simultaneously and radially dilates each currently segmented region over its morphological neighborhood until blocked by its rival (maximum radius of dilation is determined by FDT values). Morphological neighborhood [7] is defined as the region connected by paths with monotonic FDT values. In Figure 1(h), regions marked in cyan (or, yellow) represents the expansion of the red (respectively, blue) object after morphological reconstruction. In the next iteration, the FDT-connectivity paths of one object are not allowed to enter into the region already assigned to its rival. This strategy



**Fig. 2.** (a-f) Results of applying the MSTMO algorithm to a computer-generated 3D phantom at  $3 \times 3 \times 3$  (a,b),  $4 \times 4 \times 4$  (c,d) and  $5 \times 5 \times 5$  (e,f) down sampling. (a,b) 3D renditions of the phantom images before (a) and after (b) applying topo-morphologic separations. On each image, several 2D cross sectional images are presented to illustrate relative overlap at various scales. (g-l) Same as (a-f) but for another phantom.

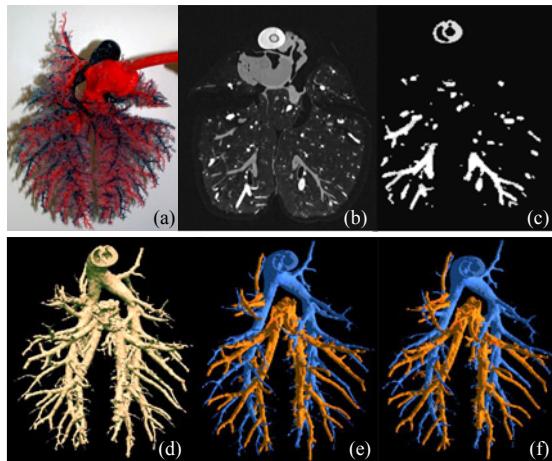
facilitates object separation at smaller scales (Figure 2(i)). The iterative process continues until it converges. For the phantom image of Figure 2(d), the method stopped after 12 iterations; see Figures 2(j,k) for final separation. A detailed mathematical formulation of different steps in the algorithm is presented in [7].

## 2.2 Mathematical Phantoms

Two phantoms were computer-generated where each phantom is an assembly of a pair of tubular tree objects running quasi-parallel across the slice direction with different geometry and varying levels of fuzziness, overlap, scale and noise (Figure 2). Initially, the phantom images were generated at high resolution and then down-sampled using  $3 \times 3 \times 3$ ,  $4 \times 4 \times 4$  and  $5 \times 5 \times 5$  windows to simulate partial volume effects. Each down-sampled image was further degraded with additive white Gaussian noise at SNR of 12. Using a graphical user interface, exactly one seed point was placed for each tubular object near its center on the top-most slice at the largest-scale level.

## 2.3 Pulmonary Vessel Cast Data

To evaluate the performance of the algorithm under true complexity of coupling and fusion of A/V trees, we designed an experiment using vessel cast of a pig's lung. To generate a vessel cast data, the animal was first exsanguinated. While maintaining ventilation at low positive end-expiratory pressures (PEEP), the pulmonary vasculature was flushed with 1L 2% Dextran solution and pneumonectomy was performed. While keeping the lungs inflated at approximately 22 cm H<sub>2</sub>O airway inflation pressure, a rapid-hardening methyl methacrylate compound was injected into the vasculature to create a cast of the pulmonary A/V trees. The casting compound was mixed with red oil paint for the venous

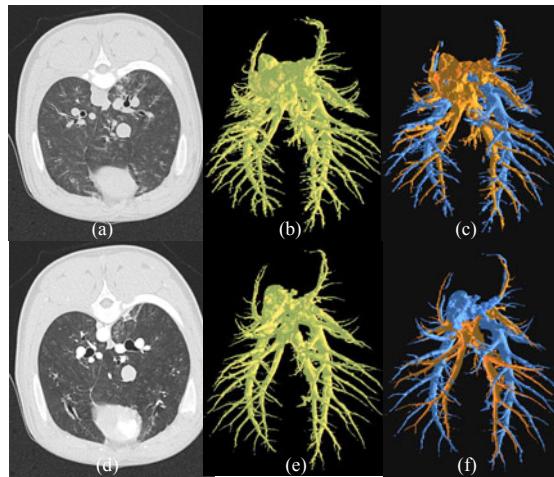


**Fig. 3.** A/V separation results on pulmonary pig vessel casting. (a) A photograph of the vessel cast. (b) A coronal slice from the original CT cast data with different contrast for A/V. (c) Same as (b) after contrast elimination. (d) 3D renditions of the contrast-eliminated vasculature. (e) True A/V separation from the original contrast-separated CT data. (f) A/V separation using the MSTMO algorithm on contrast-eliminated data.

(oxygenated) side and blue oil paint for the arterial (deoxygenated) side of the vascular beds (Figure 3(a)). The arterial side was also contrast-enhanced by the addition of 10 cc of Ethiodol to the casting compound. The vessel cast was scanned in a Siemens Sensation 64 MDCT scanner at 100 mAs, 120 kV, pitch 1 and the image was reconstructed at 0.5 mm slice thickness and 0.47 mm pixel size (Figure 3(b)). True A/V separation was obtained from the MDCT image by thresholding at a suitable CT value (Figure 3(c)). The effect of distinguishing contrast between A/V trees was subsequently eliminated using a post image processing algorithm and subsequently, the contrast-eliminated image was down-sampled by  $2 \times 2 \times 2$  and at the low resolution this image was used by the MSTMO algorithm (Figures 3(c,d)). For each of A/V trees, exactly ten seeds were manually selected on the post processed image using a 2D graphical interface and the A/V separation was computed (Figure 3(f)).

#### 2.4 *In vivo* Multi-volumes Pulmonary Pig CT Imaging

To evaluate the reproducibility of the method, we have used pig lung CT data at different PEEPs as well as human lung CT data. Specifically, an anesthetized pig was imaged in the prone body posture at two different PEEPs: 7.5 and 18 cm H<sub>2</sub>O with the intravenous infusion of 40 cc at 2.5 cc/sec of contrast agent (Omnipaque, GE Healthcare Inc, Princeton, NJ) into the external jugular vein. This contrast protocol was selected to match that used clinically for evaluating the pulmonary arterial bed for the presence of pulmonary emboli. The following



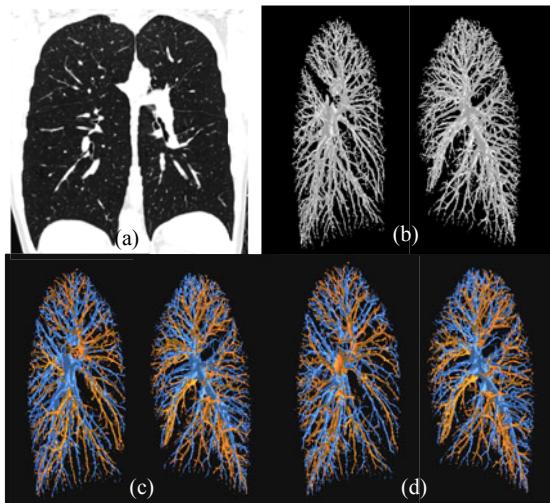
**Fig. 4.** Results of A/V separation on contrast-enhanced *in vivo* CT images of a pig's lung at two different volumes. (a,d) Visually matched coronal slices from original CT images at 7.5 cm (a) and 18 cm (d) H<sub>2</sub>O PEEP. (b,c,e,f) 3D renditions of pulmonary vasculature before (b,c) and after (e,f) A/V separation using the MSTMO algorithm.

CT protocol was used for imaging: 100 mAs, 120 kV, pitch of 1, 0.75 mm slice thickness, and 0.47mm pixel size (Figures 4(a,d)).

To evaluate multi-user reproducibility of the method, we have used clinical thoracic CT data of two patients, acquired at: 100 mAs, 120 kV, pitch 1, slice thickness 0.75 mm, and pixel size 0.6 mm. Binary pulmonary vasculature was segmented in a CT image using tree-connectivity on CT data, after applying a tubular enhancement filter as proposed by Shikata *et al.* [12]. Subsequently, a fuzzy representation of the vascular tree was computed using the following three steps - (1) determine the mean and standard deviation of intensity values over segmented vascular region, (2) dilate the binary vascular region by one voxel to include partial-volume voxels, and (3) assign fuzzy membership at each voxel over the dilated vascular region using a step up Gaussian function parameters as computed in Step 1. 3D rendition of segmented vascular tree is presented in Figure 4(b). Seed points for each of the A/V trees were manually selected by two independent experts using a self-developed 2D/3D graphical interface.

### 3 Results

Qualitative results of A/V separation on mathematical phantoms are presented in Figure 2. The algorithm successfully separated the two conjoined objects with significant overlaps excepts for 3 cases in first phantom at  $4 \times 4 \times 4$  down sampling and both phantoms at  $5 \times 5 \times 5$  down sampling. It may be noted that the effect of original overlap at high resolution gets compounded after down sampling at lower resolutions; note that, for the same phantom, the method successfully



**Fig. 5.** Results of A/V separation on pulmonary CT data. (a) A coronal image slice. (b) 3D rendition of the vasculature. (c,d) Color-coded renditions of A/V separations using seeds from two independent experts.

resolved the two structures at all scales under  $3 \times 3 \times 3$  down sampling. To quantitatively analyze the performance of the method on phantoms, the ground truth of separations of two objects in a phantom was computed based on the knowledge of their generation at the original resolution prior to down sampling. Let  $T_1$  and  $T_2$  denote true segmentations of the two objects in a phantom and let  $S_1$  and  $S_2$  denote their segmentations as computed by the MSTMO algorithm. *True positive* (TP) and *false negative* (FN) of the computerized separation of two objects are defined as follows:

$$TP = \frac{(T_1 \cap S_1) \cup (T_2 \cap S_2)}{T_1 \cup T_2} \quad \text{and} \quad FN = \frac{(T_1 \cap S_2) \cup (T_2 \cap S_1)}{T_1 \cup T_2}. \quad (1)$$

TP and FN results for the two mathematical phantoms at  $3 \times 3 \times 3$ ,  $4 \times 4 \times 4$  and  $5 \times 5 \times 5$  down sampling are  $96.47 \pm 2.65\%$ ,  $1.21 \pm 0.17\%$ ,  $95.70 \pm 2.99\%$ ,  $1.08 \pm 0.44\%$ ,  $94.32 \pm 3.27\%$ , and  $2.46 \pm 1.06\%$ , respectively; here, the results are presented as mean  $\pm$  std. These results demonstrate that the MSTMO algorithm produces high accuracy in spatial delineation of the two structures while its performance in terms of maintaining the identity of individual structures along their axes is qualitatively illustrated in Figure 2. It may be pointed out that the sum of TP and FN is not always 100% which means that some pixels are missed at conjoining locations to be part of any of the two objects. Such situations happen when the rights of grabbing a pixel by two objects become equal.

Results of A/V separation on vessel cast data are presented in Figure 3. CT imaging of the vessel cast at clinically recommended dose could successfully capture 8-10 branch levels (Figure 3(d)). Contrast-separated A/V trees and A/V separation results by the MSTMO algorithm from contrast-eliminated and

**Table 1.** Results of quantitative multi-user reproducibility analysis of A/V separation on clinical thoracic CT imaging of patients using the MSTMO algorithm

		Agreement	Disagreement
Data1	left lung	88.90%	10.07%
	right lung	84.57%	14.7%
Data2	left lung	94.54%	5.13%
	right lung	88.01%	11.43%

downsampled vasculature (Figure 3(d)) are shown in Figures 3(e) and (f), respectively. The two A/V separations results of Figures 3(e) and (f) show strong visual agreement except at a few locations with fine scales. A quantitative comparison produced a 94.4% TP and 1.6% FN of A/V separation obtained by the MSTMO algorithm as compared to contrast-separated A/V trees. This result indicates that most of the A/V separations by the algorithm are correct in the sense that only at 1.6% situations, arteries was labeled as veins or vice versa. For this example, approximately 4% of vasculature volume was missed by the MSTMO algorithm that includes the space between the two structures at conjoining locations and also, some of the missing small branches.

Results of A/V separation in pig lung at different lung volumes are presented in Figure 4. Both vasculature and A/V separations results at two lung volumes are displayed from similar views. Agreements of A/V separations results at two volumes are visually promising. Results of A/V separation by two independent users on patient pulmonary CT illustrated in Figure 5. Although no true A/V segmentations are available to compare with for this experiment, homogeneous distribution of A/V structure complies with biological knowledge of pulmonary artery/vein structures. Despite very dense tree structure of pulmonary vasculatures, for most A/V branches, results of separation by two independent experts agree. In order to quantitative evaluate multi-user reproducibility of the method, we computed following two measures:

$$Agreement = \frac{(A_1 \cap A_2) \cup (V_1 \cup V_2)}{A_1 \cup V_1 \cup A_2 \cup V_2}, \quad (2)$$

$$Disagreement = \frac{(A_1 \cap V_2) \cup (A_2 \cup V_1)}{A_1 \cup V_1 \cup A_2 \cup V_2}, \quad (3)$$

where  $A_i$  and  $V_i$  denote separated A/V using seeds selected by the  $i^{th}$  expert. Results of quantitative analysis of multi-user reproducibility of A/V separation for patient CT data are presented in Table 1. As shown in the table, although agreement between two independent users is generally high, there are some disagreements in A/V separations by two users. It may be noted that these results of reproducibility reflect the robustness of the entire process in the presence of human errors and effectiveness of the graphical interface.

## 4 Conclusions

In this paper, we have studied both accuracy and reproducibility of a multi-scale topo-morphologic opening algorithm for separating two similar-intensity objects with multi-scale fusions. For non-contrast pulmonary CT images, the geometry of coupling between A/V is quite challenging and unknown. However, the method has shown acceptable performance with a reasonable number of seeds. Approximately, 25-35 seeds were manually selected for each of the A/V trees on a patient CT lung data. Both qualitative and quantitative results on mathematical as well as physical phantoms have established viability of the new method in resolving multi-scale fusions of two simlimar-intensity structures with complex and unknown geometry of coupling. Experimental results have established reproducibility and usefulness of the current method in separating arteries and vein in pulmonary CT images with no contrast.

## References

1. Cho, Z.H., Jones, J.P., Singh, M.: Foundation of Medical Imaging. Wiley, Chichester (1993)
2. van Bemmel, C.M., Spreeuwiers, L.J., Viergever, M.A., Niessen, W.J.: Level-set-based artery-vein separation in blood pool agent CE-MR angiograms. *IEEE Trans. Med. Imag.* 22, 1224–1234 (2003)
3. Lei, T., Udupa, J.K., Saha, P.K., Odhner, D.: Artery-vein separation via MRA - an image processing approach. *IEEE Trans. Med. Imag.* 20, 689–703 (2001)
4. Buelow, T., Wiemker, R., Blaffert, T., Lorenz, C., Renisch, S.: Automatic extraction of the pulmonary artery tree from multi-slice CT data. In: Proc SPIE: Med. Imag., pp. 730–740 (2005)
5. Yonekuraand, T., Matsuhiro, M., Saita, S., Kubo, M., Kawata, Y., Niki, N., Nishitani, H., Ohmatsu, H., Kakinuma, R., Moriyama, N.: Classification algorithm of pulmonary vein and artery based on multi-slice CT image. In: Proc SPIE: Med. Imag., pp. 65142E1–65142E8 (2007)
6. Frigault, M.M., Lacoste, J., Swift, J.L., Brown, C.M.: Live-cell microscopy - tips and tools. *J. Cell. Sci.* 122 (2009)
7. Saha, P.K., Gao, Z., Alford, S.K., Sonka, M., Hoffman, E.A.: Topomorphologic separation of fused isointensity objects via multiscale opening: separating arteries and veins in 3-D pulmonary CT. *IEEE Trans. on Med. Imag.* 29, 840–851 (2010)
8. Saha, P.K., Wehrli, F.W., Gomberg, B.R.: Fuzzy distance transform: theory, algorithms, and applications. *Comp. Vis. Und.* 86, 171–190 (2002)
9. Rosenfeld, A.: Fuzzy digital topology. *Information and Control* 40, 76–87 (1979)
10. Udupa, J.K., Samarasekera, S.: Fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. *Grap Models and Imag. Proc.* 58, 246–261 (1996)
11. Saha, P.K., Udupa, J.K.: Iterative relative fuzzy connectedness and object definition: theory, algorithms, and applications in image segmentation. In: Proc. of IEEE MMBIA (2000)
12. Shikata, H., Hoffman, E.A., Sonka, M.: Automated segmentation of pulmonary vascular tree from 3D CT images. In: Proc. of SPIE: Med. Imag., pp. 107–116 (2004)

# Video Event Detection as Matching of Spatiotemporal Projection

Dong-Jun Park and David Eichmann

Institute for Clinical and Translational Sciences,  
The University of Iowa, Iowa City IA 52242 USA

**Abstract.** Detection of events and actions in video entails substantial processing of very large, even open-ended, video streams. Video data presents a unique challenge for the information retrieval community because it is hard to find a way to properly represent video events. We propose a novel approach to analyze temporal aspects of video data. We consider the video data as a sequence of images that form a 3-dimensional spatiotemporal structure, and multiview orthographic projection is performed to transform the video data into 2-dimensional representations. The projected views allow a unique way to represent video events, and we apply template matching using color moments to detect video events.

## 1 Introduction

Automatic event detection is increasingly important to the information retrieval research community in supporting applications such as content-based retrieval, video summarization and surveillance for security. The seemingly simple task of searching for digital video in large databases is enormously challenging. The scale (artifacts measures in gigabytes and collections measured in terabytes) and the rapid generation of video data creates special challenges for storage, annotation and retrieval. Accessing content is an inherently time consuming process due to the stream-based structure of video data. A video event can only be understood through a sequence of images (frames), and this multi-dimensionality makes video retrieval hard and time-consuming. Most importantly, the *temporal aspect* of these data has not been adequately addressed in most retrieval systems [15], which prohibits a true understanding of video events. Even further, the user's information need is dynamic, and a restricted approach with specialized parametric models is not desirable for numerous real-world applications [24]. We introduce a novel way to represent *video events* using multiview orthographic projection on spatiotemporal video stacks. Focusing on the notion of projection allows us to apply existing vision detection and image retrieval algorithms such as [20][19].

As a case in point, consider the British Broadcasting Corporation (BBC) seeking a capability to automatically index and retrieve their archive of programs [8]. They have 750,000 hours of video data in their archive with each week producing an 700 additional hours of programming. They are handling about 2000

enquiries each week to locate certain video segments satisfying user information needs. Now imagine an automated system that can search for video events with such dynamic data. This poses the challenging tasks not only for storage but also for video annotation, indexing and retrieval. Due to the sheer volume and complexity of data, most video retrieval systems treat video as sequences of still images and extract relevant low-level features from selected keyframes to compare visual contents [15]. We assert that video events can only be understood by examining temporal characteristics.

We conjecture that one of the main challenges for video event detection is properly *representing video events*. How can we represent the visual changes over time that represents a certain motion, allowing fast scanning of existing data as well as a high degree of adaptability to new video events? The current state of video retrieval research is not fully accommodating of these unique multi-dimensional data characteristics. Representing temporal video events using a spatiotemporal video stack is not a new idea [11][18][23]. However, instead of directly analyzing this 3D structure, we look at the remnants of video events by generating projected views, which represent change in visual information along the temporal domain. The notion of projection and its visual *meaning* is prevalent in other research areas such as vision detection and image retrieval. For instance, content-based image retrieval aims to retrieve objects or contents based on their projected properties onto image plane. As shown by [20][19], we choose color as our detection feature to compare video events in projection views. Instead of individual motion modeling, we apply a template matching approach for video event detection.

The rest of the paper is organized as follows. In section 2, we present an overview of video retrieval research, particularly research work that has dealt with the temporal aspect of video data. Section 3 discusses video stack projection model. Section 4 presents our approach to capture similarity of video event projection using color moments. Finally, we present our experiment results using above mentioned approaches in Section 5. We conclude our article with a remark on how our system can be improved for our future research direction.

## 2 Related Work

In the context of video information retrieval, temporal modeling by analyzing sequences of images is a relatively new research area [15]. It is not yet well-established how spatiotemporal information should be represented. It has been observed that the temporal features of video data have been ignored in TRECVID, leading to low performance in detecting events and actions in the data. Recently, the importance of events as a semantic concept has been acknowledged, and automatic event detection in the context of video retrieval on large-scale data has begun in the recent TRECVID workshops [1]. The participating groups employed a number of approaches, including optical flow, motion vector, background/foreground segmentation, and person detection/tracking. The results from this event highlight the current state-of-the-art in video systems, and

the general consensus within the video retrieval community is that effectively every aspect of VIR is still an open problem and calls for significant further research.

Human motion understanding based upon a sequence of images has been studied actively in computer vision [2][12][21]. For example, Moeslund *et al.* highlight more than 300 research papers from 2000 to 2006 in vision-based motion capture and analysis [12]. Zhao *et al.* discuss multiple person segmentation and appearance model-based tracking in complex situations [25]. Probabilistic recognition models such as Bayesian networks or Hidden Markov Models are discussed in [22][9][17]. More closely related, visual appearance-based work includes motion-based approaches using temporal templates [4] and spatiotemporal volumetric approaches [7][3][23]. In Bobick and Davis' work [4], a temporal template is constructed using a motion-energy image (MEI) and a motion-history image (MHI). An MEI is a binary cumulative motion image that shows a range of motion within a sequence of images. An MHI shows the temporal history of motion where more recently moving pixels are brighter. Thus, this MHI implicitly represents the direction of movements. This approach is not suitable to apply for video retrieval on large-scale corpus since this relies on well-constructed video segments (e.g., the segment exhibits a singular motion).

Spatiotemporal approaches interpret a video stream as a spatiotemporal 3D volume by stacking a sequence of images. The motion performed within this 3D volume is treated as a 3D object in the spatiotemporal space [3][23]. This approach requires reliable object segmentation to form a spatiotemporal event object. Spatio-temporal interest points have been applied to detect significant local variations in both space and time to detect events [1][13]. Spatio-temporal volume can be sliced to reveal certain patterns induced by moving objects [14][16]. Shechtman *et al.* used space-time correlation of the video with an action template [18]. To overcome the local segmentation issues, Ke *et al.* applied spatio-temporal volumetric feature to scan video sequences in space and time [10]. Zelnik-Manor and Irani recognized the need for simpler behavioral distance measure to capture events with different spatio-temporal extents and applied unsupervised event clustering based on behaviors [24].

Video retrieval systems need to effectively handle large-scale data to retrieve a variety of video events. Most current systems focus on feature-based approaches that require specialized parameter optimization. It is not straightforward how establish how (or if) a specific feature is related to specific video events, and building a detector for new video events usually means an iterative process involving newly selected visual features and metrics. Dynamic information needs with large-scale data require a more robust and scalable framework for video events retrieval task. The spatiotemporal volumetric approaches discussed above model a video segment as a 3D shape to compute motion similarities. We propose alternatively to analyze video events as inherently *lossy* projections of 3D structure onto orthogonal 2D representations. We argue that this reduced form of data supports a robust and scalable approach that can be applied onto large-scale data.

### 3 Projection of Spatiotemporal Volume

Video data can be considered a sequence of images representing 3-dimensional scenes in motion. The 3-dimensional world is projected onto a 2-dimensional screen by video recording devices. Photographic projection is performed repeatedly over time, and a stream of 2D images records the changes in scene. The projection function in this case exhibits binary characteristics. A 2D video of a 3D world inherently loses some information in the projection process. However, complexity is reduced significantly through the process while providing a nice summary of the 3D scene in motion.

We similarly represent video events using projection onto a spatiotemporal video stack, since the complexity of video data prohibits effective direct event-based retrieval. There are various ways of transforming 3D objects onto a 2D surface. We take a video as a stream of images and view this as a 3D volumetric structure by stacking images. The objects in motion within the video stack can be viewed as a 3D structure. Volumetric shape description can be achieved by mapping into multiple 2D projections. Orthographic projection can be performed by surrounding the object with projection planes forming a rectangular box. The Radon transform is one of the general methods to reconstruct an image from a series of projections. The projection function in this case is not binary. Rather this function is dependent on the density of the volume that is being projected.

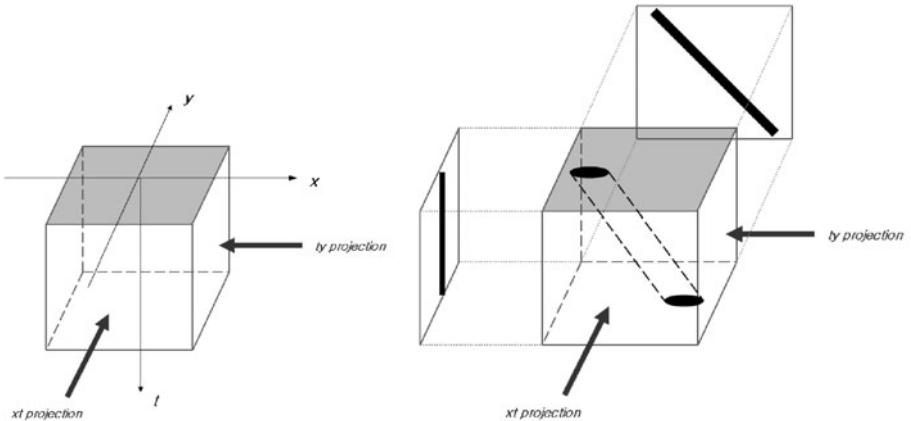
Our goal here is not to reconstruct the 3D structure perfectly from projections. Rather we are looking for a form of representation that provides a summary of the 3D spatiotemporal stack. We apply two orthogonal Radon projections to each image and stack the resulting rasters. These two resulting projection views reduce the complexity of the data significantly and opens up very interesting research opportunities.

A single image is viewed as a 2D structure with width and height. Suppose that physical objects in 3D space and their corresponding video stacks follow the following assumptions:

- The motions of physical objects are observed from a fixed camera. The background remains constant.
- The location, speed and direction of a physical object changes smoothly over time.
- The lighting does not change over time. The illumination of the object remains constant.

This set of assumptions allows the moving object to maintain relatively constant color and brightness over time. We regard the motion of the spatial object over time within spatiotemporal volume as 3D shapes induced by the contours in the spatiotemporal volume.

A 3D shape can be represented onto a 2D using multiview orthographic projection (*MOP*)[\[6\]](#), as shown in figure [11](#). The Radon transform specifies that



**Fig. 1. 2** Projection is performed onto video stack

the projection accumulates along the orthogonal dimension. The continuous 2D Radon transform of  $f(x, y)$  and the discrete case with  $W \times H$  image are:

$$g(\theta, s) = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} f(x, y) \delta(x \cos \theta + y \sin \theta - s) dx dy \quad (1)$$

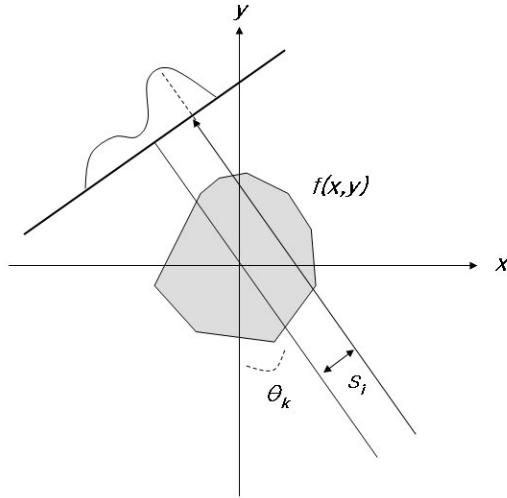
$$g(\theta, s) = \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} f(x, y) \delta(x \cos \theta + y \sin \theta - s) \quad (2)$$

Thus, we can see that projection accumulates the pixels of  $f(x, y)$  along the line defined by  $s$  and  $\theta$ . Performing this for all  $s$  values with given  $\theta$  produces one projection, as shown in figure 2. The number of projections required for effective detection can vary due to the complexity of the scene. In this paper, we deliberately choose 2 orthogonal projections with  $\theta$  equal to either 0 or  $\pi/2$ . Thus, the direction of the projection coincides with the image coordinate system. Each projection reduces a 2D image onto a 1D signal, which is stacked to produce a 2D projection view of the 3D video stack.

The  $xt$  projection is constructed by projecting through the stack along the  $y$  axis. Each image is projected into a single raster along the  $y$  axis, and this raster becomes a row of the  $xt$  view. Thus, this view captures the horizontal motion element in a video data. The dimension of the view will be  $W \times T$  where  $T$  is the number of images. Equation 2 with  $\theta = 0$  becomes

$$g(0, s) = \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} f(x, y) \delta(x - s) \quad (3)$$

The  $ty$  projection is constructed by projecting through the stack along the  $x$  axis. Each image is projected into a single raster along the  $x$  axis, and this line becomes a row of the  $ty$  view. This view captures the vertical motion element in



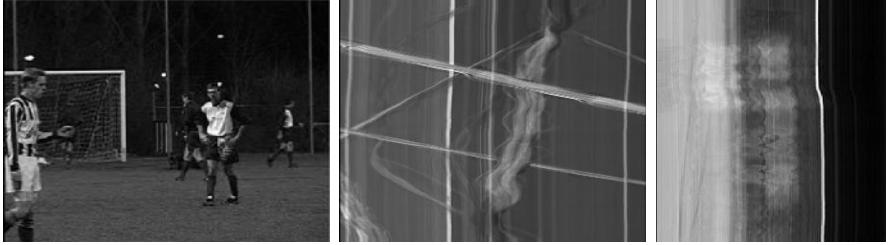
**Fig. 2.** The Radon Transform

a video data. The dimension of the view will be  $H \times T$ . Equation 2 with  $\theta = \pi/2$  becomes

$$g(\pi/2, s) = \sum_{x=0}^{W-1} \sum_{y=0}^{H-1} f(x, y) \delta(y - s) \quad (4)$$

If there is no video event (ie, no motion), the projection results in a constant raster value for each image. The objects and the background present within video data are not discernible, and this is represented as vertical bandings in  $xt$  and  $ty$  projection views. Any motion generates variation away from background and results in a distinct motion track. However, the projection may not be unique among all potential motions. We recognize that there are other projection functions that may deal with this type of data, but we conjecture that our choice of density projection is a logical step away from binary projection. By taking only two orthogonal projections, the data complexity reduces significantly but lose some details at the same time. However, the choice of two projections is justifiable under the systematic criteria of event-based retrieval framework. Also, it is easier for human users to understand the projection views when it coincides with the image coordinate system. As we can see from Equation 2, the choice of  $f(x, y)$  can result in different projection views. For this paper, we choose  $f(x, y)$  to be a 256-bin graylevel intensity value.

Figure 3 shows the projection output of a 10-second video segment with its keyframe. The projection model allows the temporal information to be projected into the spatial domain, which can be treated as 2-dimensional data. The Radon projection results in values much larger than the image pixel can hold. We applied 256-bin normalization to the projection views to store the result. Due to the



**Fig. 3.** People playing soccer: A keyframe and its  $xt$  and  $ty$  projections are shown. Two primary figures (the person in the middle and another walking across to the right side) are shown clearly in the  $xt$  view. The elongated shape of the primary action figure (i.e., human) seems to respond better to  $xt$  projection. The projection still represents the motion surprisingly well.

much reduced form of representation, our approach ensures a high level of scalability that can be applied to large-scale corpora. It is important to note that the projection is not feature-specific nor specialized for certain events. It is a part of the general preprocessing that takes place before any modeling or classification efforts and hence need only be done once for any given video sequence.

## 4 Similarity of Projections

Template matching is a technique in digital image processing for finding parts of an image which match a given template image. We conjecture that motion-induced projection produces similar templates and apply *SAD* (Sum of absolute differences) to compare between motions. However, one-to-one matching at the pixel level is computationally expensive. Also, the compressed video tends to be noisy, making pixel-level comparison less effective. Color has been one of the most dominant feature in various vision problems [20]. Similarity of color distribution using color moments has been successfully applied in vision retrieval systems [5][19]. We divide the projection views into non-overlapping subblocks and compare their similarity using the first 3 moments of each subblock. Since our input and output are 256-bin gray-levels, our color scheme is in a single channel.

For subblock  $a$  of projection view, the first moment is defined by

$$m_{1,a} = \frac{1}{N} \sum_{i=1}^N p_j, \quad (5)$$

where  $p_j$  is the gray-level value of the  $j$ -th pixel and  $N$  is the total number of pixels in the subblock  $a$ . The second and third moment of the same subblock are then defined as:

$$m_{2,a} = \left( \frac{1}{N} \sum_{i=1}^N (p_i - m_{1,a})^2 \right)^{\frac{1}{2}} \quad (6)$$

$$m_{3,a} = \left( \frac{1}{N} \sum_{i=1}^N (p_i - m_{1,a})^3 \right)^{\frac{1}{3}} \quad (7)$$

Then, the distance  $d$  of two subblocks  $a$  and  $b$  are calculated as absolute differences of each moments:

$$d(a, b) = |m_{1,a} - m_{1,b}| + |m_{2,a} - m_{2,b}| + |m_{3,a} - m_{3,b}| \quad (8)$$

Thus, the distance of image  $I$  at pixel location  $(x, y)$  and template  $T$  is  $SAD$  at that location:

$$D(I, T) = \sum_0^{T_x} \sum_0^{T_y} d(I_{xy}, T) \quad (9)$$

For temporal detection of event occurrence at  $t$ , we compare projection view  $V$  and template  $T$  using the following distance function. The  $xt$  projection view  $V_{xt}$  and its template  $T_{xt}$  as well as its corresponding  $ty$  distance function are the basis:

$$D(V_t, T) = \alpha D(V_{xt}, T_{xt}) + \beta D(V_{ty}, T_{ty}), \quad (10)$$

where  $\alpha$  and  $\beta$  are use specified weights for each term. Note that the smaller  $D(V_t, T)$  a matching template has, the more similar it is to the input event.

## 5 Experiment Results

To evaluate the performance of the proposed method, we conducted an experiment on surveillance video from the TRECVID event detection task. We selected 7 hours of video comprising 10GB of data (a 1-hour training set to extract templates and 6 hours comprising a testing set). The video data is mpeg-compressed, 720 by 526 pixels at 24 frames per second. As noted by TRECVID participants, the data is quite noisy due to the compression process, displaying both block artifacts and random pixel noise. The data displays two elevator doors with people going in and out of the elevators.

We performed our projection process and produced both  $xt$  and  $ty$  views. The projection is normalized to fit into a 256-bin gray-level projection view, and minimally processed (meaning no noise reduction or trail extraction). The testing set of 8.5GB is reduced to about 25MB in the form of projection views, and we based all detection solely on this data. We selected three events that are prevalent in the data - *elevator door open*, *getting into elevator* and *getting out of elevator*. Figure 4 shows the chosen templates for this experiment. We chose a single template for each event, which is a possible real-world scenario where a user specifies an input template using a query-by-example interface.



**Fig. 4.** The templates extracted from the training set - *elevator door open*, *getting into elevator* and *getting out of elevator*. For this experiment, we used the partial of the templates.

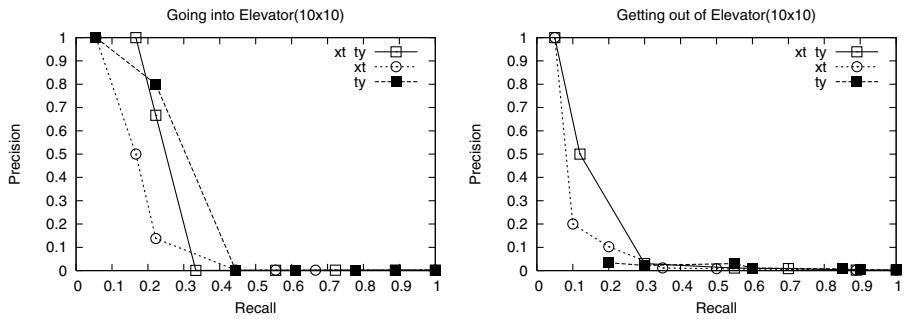
We measure performance using standard formulations of precision and recall:

$$P = \frac{Rel \cap Ret}{Ret}, \quad (11)$$

and

$$R = \frac{Rel \cap Ret}{Rel}, \quad (12)$$

where precision  $P$  is the fraction of the retrieved  $Ret$  that are relevant  $Rel$ , and recall is the fraction of the relevant  $Rel$  that are successfully retrieved  $Ret$ .



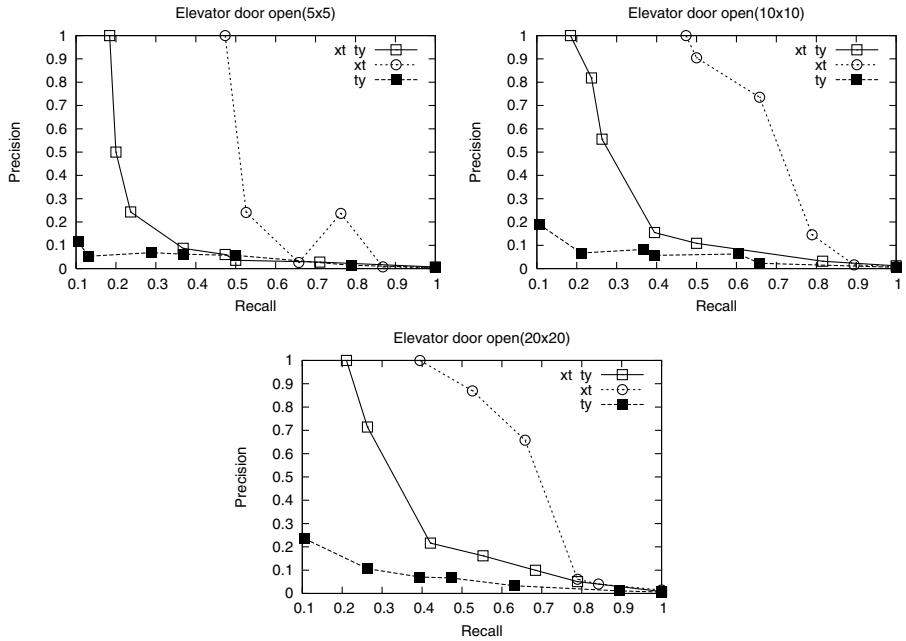
**Fig. 5.** The precision vs. recall curve for *people getting in and out of the elevator*. The plots show comparable results for each variation in  $\alpha$  and  $\beta$  values.

Figure 5 and 6 show the precision vs. recall curve for each event. To see the effect of each projection view on detection performance, we performed three sets of experiments for each event.

- $xt$  and  $ty$  composite:  $\alpha = \frac{1}{2}$  and  $\beta = \frac{1}{2}$
- $xt$  only:  $\alpha = 1$  and  $\beta = 0$
- $ty$  only:  $\alpha = 0$  and  $\beta = 1$

We also varied the size of subblock as shown in Figure 6. As we can see from the plots, *Elevator door open* shows generally better performance than other event sets. Also, the runs with  $10 \times 10$  and  $20 \times 20$  are slightly higher than  $5 \times 5$  runs. The projection value depends on the density of the volume being projected,

and the motion path generated can be varied due to light condition, color properties and noise. Among all the templates we chose, the *xt* view of “elevator door open” showed most stable template, which certainly gets tranalted onto higher performance. Considering the results from recent TRECVID evaluations, our experiment results show substantial promise. Instead of processing the raw video data, the experiment showed that much reduced representation with color moment supports rapid detection of large-scale video data.



**Fig. 6.** The precision vs. recall curve for *elevator door open*. Three different subblock size are matched in this case ( $5 \times 5$ ,  $10 \times 10$  and  $20 \times 20$ ). All three cases show comparable results. *Elevator door open* shows much higher performance than other event sets.

## 6 Conclusion

We present a new video event detection framework. We construct a spatiotemporal stack of the video data and perform a Radon projection onto the stack to generate a summary of the video data. We base our event detection on the projection views using template matching of color moments. Applying detection task onto the simpler representation allows for rapid generation of content-based video retrieval. Considering the volume and complexity of the data, it is desirable to employ such lossy projections of video stacks.

We intend to further our work by building a video retrieval engine supporting large-scale corpora. Once projection is complete, there are various ways to enhance the signal. The results from the experiment presented here indicate that the effect of cleaner projection and template may enhance the performance.

## Acknowledgement

This publication was made possible by Grant Number UL1RR024979 from the National Center for Research Resources (NCRR), a part of the National Institutes of Health (NIH). Its contents are solely the responsibility of the authors and do not necessarily represent the official views of the CTSA or NIH.

## References

1. 2008 trecvid event detection evaluation plan (2008),  
[http://www.itl.nist.gov/iad/mig/tests/trecvid/2008/doc/  
 EventDet08-EvalPlan-v06.htm](http://www.itl.nist.gov/iad/mig/tests/trecvid/2008/doc/EventDet08-EvalPlan-v06.htm)
2. Aggarwal, J.K., Cai, Q.: Human motion analysis: a review. In: Proceedings of the IEEE workshop on Nonrigid and Articulated Motion (1997)
3. Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. In: Proceedings of the IEEE International Conference on Computer Vision, vol. 2 (2005)
4. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(3) (2001)
5. Campbell, M., Haubold, A., Ebadollahi, S., Joshi, D., Naphade, M.R., Natsev, A., Seidl, J., Smith, J.R., Scheinberg, K., Tesic, J., Xie, L.: Ibm research trecvid-2006 video retrieval system. In: Proceedings of the TRECVID 2006 (2006)
6. Carlbom, I., Paciorek, J.: Planar geometric projections and viewing transformations. *ACM Computing Surveys* (1978)
7. Efros, A.A., Berg, A.C., Mori, G., Malik, J.: Recognizing action at a distance. In: Proceedings of the IEEE International Conference on Computer Vision (2003)
8. Evans, J.: The future of video indexing in the BBC. In: NIST TRECVID Workshop (2003)
9. Hongeng, S., Nevatia, R., Bremond, F.: Video-based event recognition: activity representation and probabilistic recognition methods. *Computer Vision and Image Understanding* 96(2) (2004)
10. Ke, Y., Sukthankar, R., Hebert, M.: Efficient visual event detection using volumetric features. In: Proceedings of the IEEE International Conference on Computer Vision, vol. 1 (2005)
11. Laptev, I., Lindeberg, T.: Space-time interest points. In: Proceedings of the IEEE International Conference on Computer Vision (2003)
12. Moeslund, T.B., Hilton, A., Kruger, V.: A survey of advances in vision-based human motion capture and analysis. *Computer Vision and Image Understanding* 104(2-3) (2006)
13. Niebles, J., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. *International Journal of Computer Vision* (2008)
14. Niyogi, S.A., Adelson, E.H.: Analyzing gait with spatiotemporal surfaces. In: Proceedings of the IEEE Workshop on Motion of Non-Rigid and Articulated Objects (1994)
15. Ren, W., Singh, S., Singh, M., Zhu, Y.S.: State-of-the-art on spatio-temporal information-based video retrieval. *Pattern Recognition* 42(2) (2009)
16. Ricquebourg, Y., Bouthemy, P.: Rela-time tracking of moving persons by exploiting spatio-temporal image slices. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8) (2000)

17. Robertson, N., Reid, I.: A general method for human activity recognition in video. *Computer Vision and Image Understanding* 104(2-3) (2006)
18. Shechtman, E., Irani, M.: Space-time behavior based correlation or how to tell if two underlying motion fields are similar without computing them? *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29(11) (2007)
19. Stricker, M.A., Orengo, M.: Similarity of color images. *Storage and Retrieval for Image and Video Databases* (SPIE), 381–392 (1995)
20. Swain, M.J., Ballard, D.H.: Color indexing. *Internation Journal of Computer Vision* 7(1), 11–32 (1991)
21. Turaga, P., Chellappa, R., Subrahmanian, V.S., Udrea, O.: Machine recognition of human activities: a survey. *IEEE Transactions on Circuits and Systems for Video Technology* 18(11) (2008)
22. Yamato, J., Ohya, J., Ishii, K.: Recognizing human action in time-sequential images using hidden markov model. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (1992)
23. Yilmaz, A., Shah, M.: A differential geometric approach to representing the human actions. *Computer Vision and Image Understanding* 109(3) (2008)
24. Zelnik-Manor, L., Irani, M.: Event-based analysis of video. In: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition* (2001)
25. Zhao, T., Nevatia, R.: Tracking multiple humans in complex situations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26(9) (2004)

# PixelLaser: Computing Range from Monocular Texture

N. Lesperance, M. Leece, S. Matsumoto, M. Korbel, K. Lei, and Z. Dodds

Harvey Mudd College

**Abstract.** The impressive advances in robotic spatial reasoning over the past decade have relied primarily on rich sensory data provided by laser range finders. Relative to cameras, however, lasers are heavy, bulky, power-hungry, and expensive. This work proposes and evaluates an image-segmentation pipeline that produces range scans from ordinary webcams. Starting with a nearest-neighbor classification of image patches, we investigate the tradeoffs in accuracy, resolution, calibration, and speed that come from estimating range-to-obstacles using only single images. Experiments atop the low-cost iRobot Create platform demonstrate the accessibility and power of this pixel-based alternative to laser scans.

## 1 Motivation and Context

Robots' spatial reasoning capabilities have matured a great deal over the past decade. Effective localization, mapping, and navigation algorithms have proven themselves in long-term autonomous driving [3,17], tour-guide [2,15], and office-navigation [8] applications. Robots' most robust and widespread examples of spatial reasoning rely upon the ubiquitous laser range finder (LRF), which uses the time-of-flight of projected laser light to directly compute the range to nearby obstacles within the laser range finder's field of view.

Although effective, LRFs have drawbacks that have prevented them from entering the fast-growing field of low-cost commercially viable autonomous robots. As an alternative, monocular vision offers advantages relative to laser scans across several axes: cameras are less power-hungry, less heavy, less bulky, less range-limited, and, perhaps most importantly, less expensive.

Less is more, however, when it comes to computation. Extracting range from pixel intensities requires far more algorithmic and computational effort than extracting range from time-of-flight. Range-from-vision approaches typically use temporal feature correspondence across a monocular image stream to deduce distance from pixels [6]. This body of work is mature, but it is worth noting that these techniques are most successful when significant spatial context is used to support feature matching. Large patches of pixels facilitate accurate and precise correspondence.

### 1.1 Related Work

In order Recent approaches have boldly asked, "What can we deduce from only those patches, and not the correspondence at all!?" For instance, Hoiem et al.'s photo

pop-out [4] software and Saxena et al.'s Make3d system [12] yield range at each of a single image's pixels. Spatial grouping, e.g., into edges between groundplane and vertical planes, enable the compelling visualizations those groups have produced.

Such work has seen many robot applications. Horswill's Polly [5] pioneered robotic range-from-texture, and many systems have followed. In [4] Hoiem et al. show confidence levels in terrain navigability; Saxena et al. drive an RC car safely and quickly through rough, natural terrain [12]. Similar to the fast color/textured segmentation work of [1], these projects emphasized machine-learning contributions and task-specific image interpretation. This work, in contrast, focuses on the *accuracy* of the range-to-obstacle scans produced by image segmentation. We hypothesize that image-based scans can, in indoor situations, replace laser scans in the localization, mapping, and navigation algorithms popularized over the past decade.

Thus, we follow the efforts of [14] in which Taylor et al. use color segmentation to produce tabletop-scale range scans, though without summative accuracy results. More recently, Plagemann et al. [11] used Gaussian Processes to learn maps from pixel columns to range. They reported ~1 meter precision, sufficient to support off-the-shelf SLAM algorithms under assumptions common to human-scale indoor environments.

## 1.2 Contributions

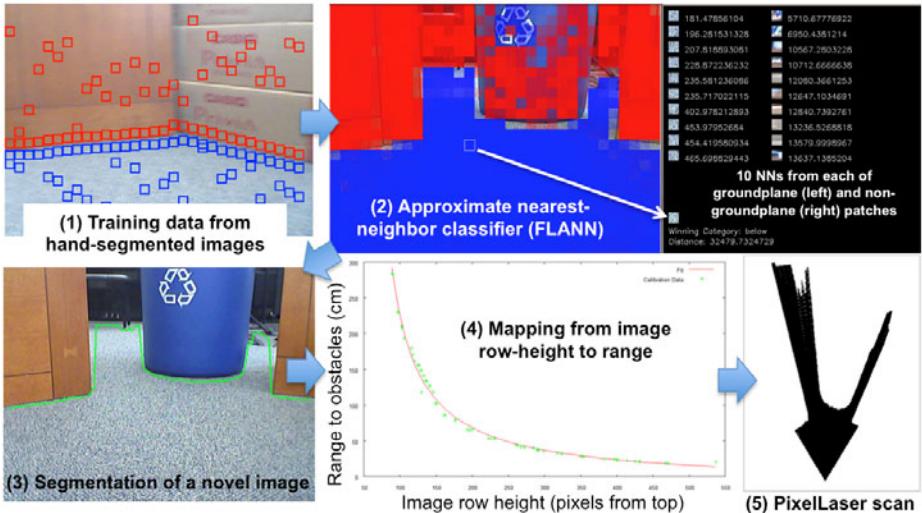
Yet the omnacam images and one-dimensional spatial context (pixel radii) of [11] make a tradeoff against accessibility and range accuracy in favor of field-of-view and angular resolution. This work offers a counterpoint to those efforts by (1) using unmodified webcam images, at much lower cost than omnacam images, (2) estimating groundplane segmentation using 2d image context, rather than pixel radii, and (3) classifying patches via nearest-neighbor matching, rather than Gaussian Processes. Section 2 details this pipeline's algorithms and implementation choices. Reporting on several robot test-runs, Section 3 validates the contributions of these design choices:

- Range accuracy comparable to the best previous results [11] using unmodified webcam images -- sufficient to replace laser scans in many algorithms
- Training efficiency that facilitates quick adaptation to new environments without presuming their visual characteristics
- Execution speed supporting real-time obstacle avoidance

We conclude in Section 4 by reflecting on how pixel-based scans may help bridge the accessibility of low-cost commercial robots with the capabilities of today's high-end experimental platforms.

## 2 PixelLaser's Algorithmic Pipeline

Figure 1 summarizes the algorithmic pipeline that creates range scans from single images: training, classification, segmentation, and transformation, as detailed below:



**Fig. 1.** PixelLaser’s pipeline for transforming images into range scans

## 2.1 Training Data

In order to create a classifier that can distinguish groundplane (*traversable*) from non-groundplane (*untraversable*) image patches, the system requires a segmented set of training images. We obtain these by driving around an environment of interest and then hand-segmenting the results. If another range-finder such as a laser were available, automatic segmentation could replace this human-guided step as in [11]. A small number of images, typically 20, suffice for training in a new setting.

The system then decomposes the 640x480 hand-segmented images into 20x20 pixel patches. To emphasize the traversable-to-untraversable terrain transitions, we extract 32 patches immediately above and below the correct segmentation boundary. To help generalize, we select 32 patches further away: the upper left panel of Figure 1 shows the patches chosen from an example image of our laboratory space.

## 2.2 Feature Extraction and Representation

To represent the  $32 \times 4 = 128$  patches from each image, we investigated the statistics of the nine 3x3 Laws texture filters [7] and average-color filters in both RGB and HSV space. We investigated the relative classification accuracy of each of these filters both individually and in pairs. The results led us to choose two color-band features, the red and the green means and variances, and two texture statistics, the variance and kurtosis of Laws’s sixth filter. Thus, the vector of those six component values represents each 20x20-pixel patch both for classifier training and classification testing.

## 2.3 Classifier-Building and Classification

We create two approximate nearest-neighbors classifiers from these six-component vectors with the FLANN library [6]. A 20-image training set contains 1280 patches in

each of these classifiers. One classifier holds those patches taken from traversable terrain; the other holds the untraversable terrain, or “obstacles.” The system is now prepared to analyze novel images.

To classify a new patch, that patch is first represented by its six-component feature vector  $P$ . Next, a predetermined quantity of  $P$ ’s (approximate) nearest neighbors are found from the traversable training patches. The average Euclidean distance  $d_{\text{trav}}$  from  $P$  to these  $N$  nearest neighbors is computed. Similarly we compute  $d_{\text{untrav}}$ , the average distance from  $P$  to the  $N$  nearest neighbors among the untraversable training patches. Finally, each patch receives an overall *traversability score*, the ratio of  $d_{\text{trav}}/d_{\text{untrav}}$ . Patches whose traversability score is lower than a threshold of 0.8 are considered untraversable. Figure 1’s top right panel shows an example of the 10 nearest neighbors used for a particular patch and a visualization of all patches: blue when traversable and red when untraversable.

## 2.4 Segmentation

Rather than classify *every* possible patch in a novel test image, the system uses a bottom-up search strategy. For a particular column in the image, patches are classified upwards from the bottom row until an untraversable patch is encountered. The resulting locations are connected together to provide the image’s segmentation, such as shown in Figure 1’s bottom left panel. We can tradeoff speed against accuracy by varying the horizontal resolution at which these computations proceed. Using 32 columns across the image requires only 0.7 seconds – sufficient to support real-time obstacle avoidance. Full-resolution processing, e.g., for more accurate map-building, requires less than 10 seconds per image in the current implementation.

## 2.5 From Segmentation to Range Scans

Our platform presumes a camera whose height and angle are fixed relative to the groundplane. Thus, there is a one-to-one correspondence between image height and obstacle distance. Rather than model all of its parameters explicitly, we use the fundamental form of this relationship to fit an empirical model that maps image row, measured as an offset from the horizon, to obstacle distance. Figure 1’s bottom right panels shows that real data fit this model well; the example illustrates that the resulting scans preserve fine environmental details at full horizontal resolution.

So, how well does the system work – and are the resulting scans useful even in algorithmic contexts designed for *laser* scans? Section 3 shows that both of these questions have promising answers.

## 3 Experiments, Empirical Validation, and Results

Figure 2 shows our iRobot Create platform whose on-board netbook runs Section 2’s algorithmic pipeline. Willow Garage’s OpenCV computer vision library provides the image acquisition and image processing through its Python bindings [10]. Similarly, the Python interface to the Fast Library for Approximate Nearest Neighbors, FLANN, provides both the offline classifier training and its online use for testing [9]. A collection of custom Python scripts provides the remainder of the pipeline’s processing. This section documents their results.

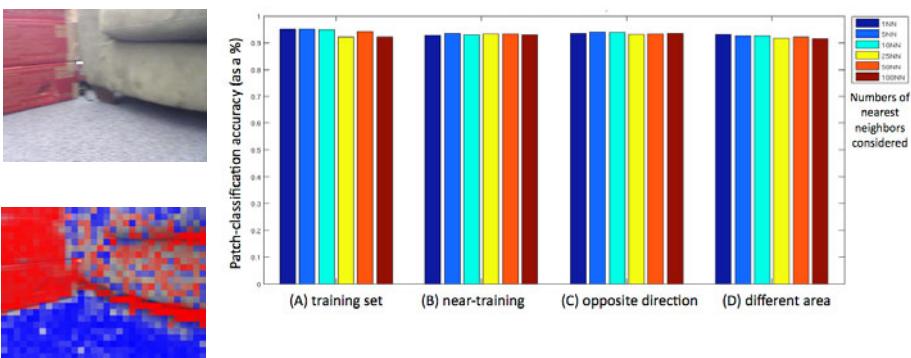


**Fig. 2.** The Create platform, netbook, and environment in which we tested PixelLaser's pipeline (left). At right are cropped images from the various test datasets.

In order to vary conditions and parameters in a principled manner, this section first focuses its attention on accuracy results from the environment depicted in Figure 2. This is a large, common indoor space in our laboratory building with a textured carpet subjected to varying lighting conditions and a variety of furnishings and objects forming the obstacles in and around the robot.

### 3.1 Texture Classification Accuracy

After training on 20 images from a run across the room, we measured the accuracy of the resulting image-patch classifier on 4 datasets of varying difficulty: (A) the training set itself, (B) another run taken under conditions similar to the training set – a nearby trajectory at the same time-of-day (and, thus, similar lighting conditions), (C) a non-overlapping visual trajectory from the same room, and (D) a non-overlapping spatial trajectory in a completely different part of the lab space. Figure 3 illustrates the strong performance of our nearest-neighbor classifier. It achieves over 90% recognition, regardless of the dataset or the number of neighbors used. Varying FLANN's accuracy parameter, the percentage of cases in which the approximate NN is the *actual* NN, affected training time but not classification accuracy.

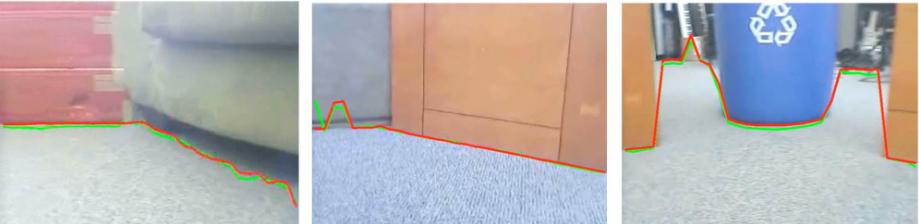


**Fig. 3.** The accuracy of the FLANN classifier in determining whether image patches are groundplane or not. At left is a test image and the resulting classification of its patches: blue indicates groundplane patches and red indicates those considered untraversable. At right are the classification accuracies across four datasets and five quantities of nearest neighbors. Even the smattering of misclassified patches at left does not push the accuracy below 90%.

### 3.2 Segmentation Accuracy

Yet classification accuracy is only the first step of the PixelLaser pipeline. Starting from classifications such as those in Figure 3, a bottom-up search seeks the correct segmentation between traversable and untraversable terrain. Figure 4 shows three such results, where the green contour represents the boundary computed purely from search through patch-classifications; the red contour has “snapped” the green one to the strongest intensity edge within a few pixels. This latter correction works well in many environments.

Figure 5 shows two histograms of errors across 10 images. The left-hand histogram holds the pixel offsets between hand-segmented and automatically segmented images from test set B, representing the conditions under which we imagine the system most often deployed. The average absolute-value pixel error is 4.24 pixels per image column or “ray,” with the median error far lower. Although [11] does not cite its errors in pixels, we believe that error measurements of pixels-per-ray *before* conversion to distance is the most useful metric for judging and comparing approaches involving image segmentation.



**Fig. 4.** Three examples of groundplane segmentations resulting from the classifications shown in Figure 3. Image patches at a horizontal resolution of 20 pixels provide the raw segmentation shown in green; the red line has “snapped” to nearby strong edges, when present.

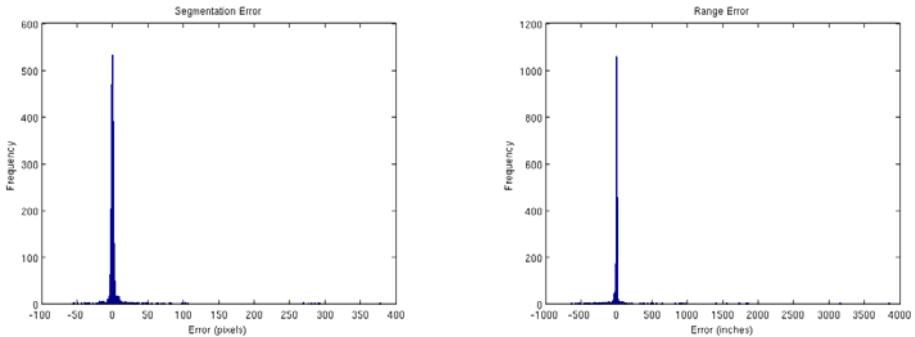
### 3.3 Scan Accuracy

Pixels-per-ray, after all, avoids the critical dependence between range accuracy and the size of the environment being sensed. A pixel close to the bottom of Figure 2’s images might represent only a millimeter of distance along the robot’s groundplane. Near the horizon, however, that same one-pixel image displacement can represent an arbitrarily large metric displacement in the robot’s environment.

To quantify these range errors, we transformed the pixel errors from left of Figure 5 to obtain the histogram of range errors that appears to its right. The average absolute error across test set B amounts to 91.2 cm. Note that this result is comparable to the approximately 100 cm average errors reported in [11]. The outliers account for almost all of this, as is indicated by the median range error of only 4.1 cm!

### 3.4 Applications to Spatial-Reasoning Tasks

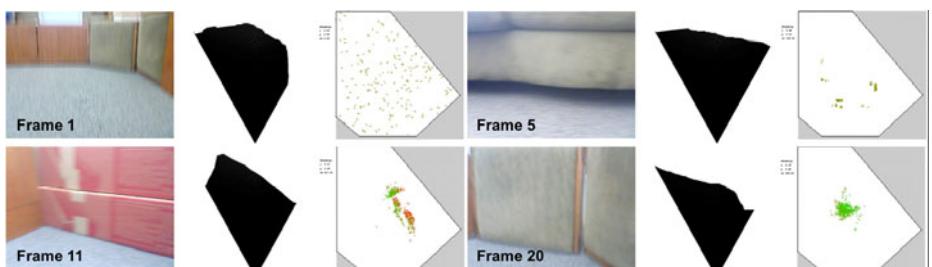
Although we have designed this test set specifically to have images with long indoor distances to obstacles, ~20 feet, we note that this comparison is not truly



**Fig. 5.** At left is a histogram of the segmentation errors for 2107 column-segmentations across a group of 28 images from dataset B. The corresponding range errors appear at right. The average absolute errors are 4.24 pixels and 91.2 cm, respectively. Note that the outliers are quite heavy, especially in the range errors at right: there, the *median* range error is only 4.1 cm!

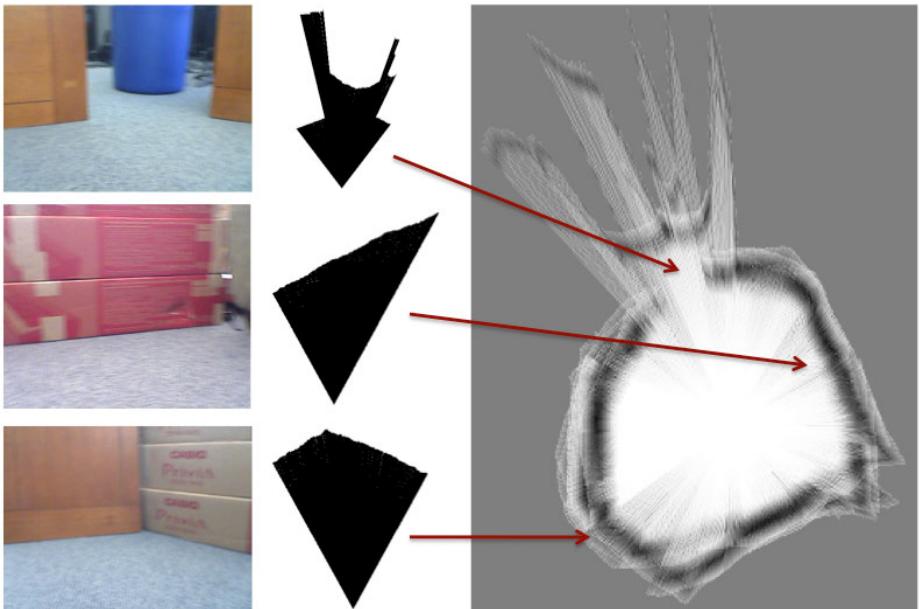
commensurate with [11]: in that work, the authors sacrifice pixels for field of view by using omnacam images. We believe that many applications can benefit from the improved accuracy and simplicity of unmodified webcam images, and so we have also tested whether these scans, with their 60° field-of-view, still support localization, mapping, and navigation tasks as do the full-circle scans of [11] or the 180° scans of LRFs.

To test localization, we mapped a small “playpen” and seeded it with a particle filter of 100 randomly selected poses. Using the Monte Carlo Localization algorithm from [16], Figure 6 shows the initial snapshot in the localization and three more instants as the particle filter converges around the correct pose after 20 scans are considered. Here, the motion model introduces considerable error to reflect the uncertainty in the iRobot Create’s odometry.



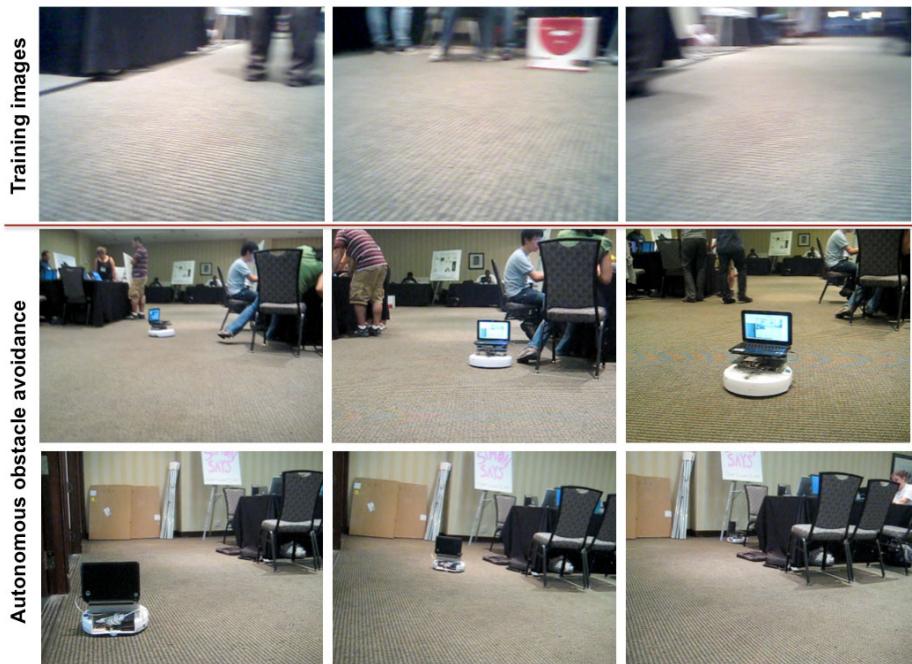
**Fig. 6.** Four frames from a localization run using MCL within a known map. The ambiguities apparent after 5 frames are handled appropriately by the particle filter: several clusters face different possible walls. Particles’ colors indicate the likelihood of matching the current scan: bright green are the likeliest; red are the least likely. The filter converges to a single, correct cluster by the final frame. Better noise models would produce even tighter convergence.

To test these scans' ability to support off-the-shelf mapping algorithms designed for use with laser scans, we used CoreSLAM [13], obtained from *OpenSLAM.org*. Three playpen images and their scans appear in Figure 7, along with the resulting map produced by CoreSLAM. As the authors point out, CoreSLAM uses only a single best-guess estimate of each scan's pose. This leads to the slippage seen in the map: there are two separate recycling bins mapped in the bird's-eye view although only one was present in the environment. Even so, we are heartened that this map is of the same qualitative accuracy as those published by CoreSLAM's authors.



**Fig. 7.** At left are three images from a partially enclosed area. The robot circled the space twice, collecting images every second. Afterwards, the scans (middle) were created and integrated into the single map shown at right using the off-the-shelf CoreSLAM algorithm

Finally, to test whether these *PixelLaser* scans can support autonomous navigation, we trained a classifier on the completely different environment of the Westin Atlanta Hotel during 2010's AAAI conference. Figure 8 shows three of the training images and several snapshots from its extended autonomous run. Guided by no sensors other than the camera running the PixelLaser pipeline, the Create wandered for twenty minutes around the conference's exhibition hall. The only obstacles the system could not handle through that time were the very thin legs of the easels and chairs. It turned out that the segmentation's horizontal resolution sometimes missed them entirely.



**Fig. 8.** Training images from a completely distinct environment (the downtown Atlanta Westin during AAAI 2010) appear at top; below are snapshots from an extended autonomous run in which the robot used no sensing other than the PixelLaser scans to avoid obstacles

## 4 Verdict and Perspective

Despite not having the sensors to take advantage of the past decade's advances in spatial reasoning, commodity robots have become a part of modern life. This work's examples of localization, mapping, and navigation are proofs-of-concept – certainly they offer broad opportunities for possible improvements. Yet even as we refine these applications, their common foundation – that of extracting range scans from image segmentations – has proven to be an accurate, flexible, and *inexpensive* approach for supporting reasoning about a robot's local surroundings. We look forward to the next generation of commercial platforms that, at no greater cost, will add such spatial reasoning to their repertoire of capabilities.

## References

1. Blas, R., Agrawal, M., Sundaresan, A., Konolige, K.: Fast color/textured segmentation for outdoor robots. In: Proceedings, IEEE IROS, Nice, France, pp. 4078–4085 (September 2008)
2. Buhmann, J., Burgard, W., Cremers, A.B., Fox, D., Hofmann, T., Schneider, F., Strikos, J., Thrun, S.: The Mobile Robot Rhino. AI Magazine 16(2), 31–38 (Summer 1995)

3. Fletcher, F., Teller, S., Olson, E., Moore, D., Kuwata, Y., How, J., Leonard, J., Miller, I., Campbell, M., Huttenlocher, D., Nathan, A., Kline, F.R.: The MIT - Cornell Collision and Why it Happened. *Journal of Field Robotics* 25(10), 775–807 (2008)
4. Hoiem, D., Efros, A.A., Hebert, M.: Recovering Surface Layout from an Image. *International Journal of Computer Vision* 75(1), 151–172 (2007)
5. Horswill, I.: Analysis of Adaptation and Environment. *Artificial Intelligence* 73, 1–30 (1995)
6. Kanade, T., Kanade, B.Y., Morris, D.D.: Factorization methods for structure from motion. *Phil. Trans. of the Royal Society of London, Series A* 356, 1153–1173 (2001)
7. Laws, K.: Rapid texture identification. In: *Proceedings, SPIE. Image Processing for Missile Guidance*, vol. 238, pp. 376–380 (1980)
8. Marder-Eppstein, E., Berger, E., Foote, T., Gerkey, B., Konolige, K.: The Office Marathon: Robust Navigation in an Indoor Office Environment. In: *IEEE ICRA 2010*, pp. 300–307 (2010)
9. Muja, M., Lowe, D.G.: Fast Approximate Nearest Neighbors with Automatic Algorithm Configuration. In: *Proceedings, VISAPP 2009* (2009)
10. OpenCV's, homepage <http://opencv.willowgarage.com/wiki/> (accessed 07/17/2010)
11. Plagemann, C., Enres, F., Hess, J., Stachniss, C., Burgard, W.: Monocular Range Sensing: A non-parametric learning approach. In: *IEEE ICRA 2008*, pp. 929–934. IEEE Press, Los Alamitos (2008)
12. Saxena, A., Chung, S.H., Ng, A.: 3-D Depth Reconstruction from a Single Still Image. *International Journal of Computer Vision* 76(1), 53–69 (2008)
13. Steux, B., El Hamzaoui, O.: CoreSLAM: a SLAM Algorithm in less than 200 lines of C code. In: *Submission ICARCV 2010* (2010), <http://www.openslam.org/coreslam.html>
14. Taylor, T., Geva, S., Boles, W.W.: Monocular Vision as Range Sensor. In: *Proceedings, CIMCA, Gold Coast, Australia*, July 12-14, pp. 566–575 (2004)
15. Thrun, S., Bennewitz, M., Burgard, W., Cremers, A.B., Dellaert, F., Fox, D., Hähnel, D., Rosenberg, C., Roy, N., Schulte, J., Schulz, D.: MINERVA: A second-generation museum tour-guide robot. In: *Proceedings, IEEE ICRA 1999*, pp. 1999–2005. IEEE Press, Los Alamitos (1999)
16. Thrun, S., Burgard, W., Fox, D.: *Probabilistic Robotics*. MIT Press, Cambridge (2005)
17. Urmson, C., Baker, C., Dolan, J., Rybski, P., Salesky, B., Whittaker, W.L., Ferguson, D., Darms, M.: Autonomous Driving in Traffic: Boss and the Urban Challenge. *AI Magazine* 30(2), 17–29 (2009)

# A Spatio-Spectral Algorithm for Robust and Scalable Object Tracking in Videos

Alireza Tavakkoli<sup>1</sup>, Mircea Nicolescu<sup>2</sup>, and George Bebis<sup>2,3</sup>

<sup>1</sup> Computer Science Department, University of Houston-Victoria, Victoria, TX, USA

<sup>2</sup> Computer Science and Engineering Department, University of Nevada, Reno, NV, USA

<sup>3</sup> Computer Science Department, King Saud University, Riyadh, Saudi Arabia  
tavakkolia@uhv.edu, {mircea,bebis}@cse.unr.edu

**Abstract.** In this work we propose a mechanism which looks at processing the low-level visual information present in video frames and prepares mid-level tracking trajectories of objects of interest within the video. The main component of the proposed framework takes detected objects as inputs and generates their appearance models, maintains them and tracks these individuals within the video. The proposed object tracking algorithm is also capable of detecting the possibility of collision between the object trajectories and resolving it without losing their models.

## 1 Introduction

Tracking of humans and other objects of interest within video frames is a very important task in many applications such as; video surveillance [1], perceptual user interfaces [2], and driver assistance [3]. Any reliable and robust object tracking mechanism may include two components [4]. The first component is responsible for generating and maintaining a model for the objects while the second process searches for potential new locations for these objects in the new frames. The target model generation deals with the dynamics of the tracked objects, learning of the scene priors and the evaluation of multiple hypotheses. The search components of the visual tracking mechanism mostly deals with the target representation localization and changes in the target appearance.

Shalom in [5] presents the filtering and data association process through a state space approach. The tracking given by the state space approach can be performed by an iterative Bayesian filtering [4]. The Kalman filter and the Extended Kalman Filter (EKF) fail when applied to scenes with more clutter or when the background contains instances of the tracked objects. Through Monte Carlo based integration methods the particle filters [6] and the bootstrap filters [7] were proposed. Also in discrete state cases the Hidden Markov Models are used for tracking purposes in [8]. These methods do not provide reliable tracking results for non-rigid objects and deformable contours. The process of probability density propagation through sequential importance sampling algorithm, employed in particle filters, is computationally expensive.

The bottom-up approach to object tracking generates and maintains the target models and searches in new frames for their potential locations [4]. This approach assumes that the amount of changes in the location and appearance of the target is small. Tuzel *et al.* in [9] proposed a new non-statistical method under the target representation and localization by employing Lie algebra to detect and track objects under significant pose changes. Loza *et al.* in [10] presented a structural similarity approach to object tracking in the video sequences. Recently SIFT features have been used in [11] tracking objects. However, reliable extraction and maintenance of the SIFT features and occlusion issues negatively affect this approach. Due to high computational cost - especially in cases where several object should be tracked over a long period of time- these methods fail to perform efficiently. Another issue in object tracking in video sequences is the ability to resolve occlusion. In this paper we propose an algorithm to reliable track multiple objects of interest and resolve possible occlusions which may occur as the number of tracked objects increase. The proposed approach is scalable to accommodate for increased number of objects within the field of view.

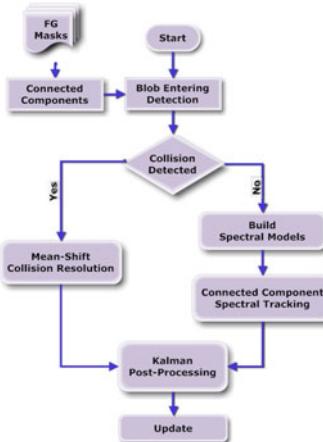
## 2 The Proposed Tracking Framework

The proposed algorithm is composed of two main stages. The first stage is the appearance correspondence mechanism. Once detected, photometric appearance based models are generated for the objects of interest. These models are considered to be the first degree estimation of the probability distribution of pixel colors. These models are then employed in the spatio-spectral connected component tracking mechanism to solve a correspondence problem between the set of detected connected components and the set of target appearance models. In the second phase, an occlusion resolution stage is used to halt the update process in case two or multiple targets occlude each other.

Since the appearance of the objects of interest are generally not known a priori, the only visual cue that can be used for detecting and tracking them is image motion. Our approach uses detected object from an efficient and reliable technique, based on background modeling and segmentation [12].

Figure 1 shows an overview of our proposed visual tracking algorithm. The proposed system uses the foreground masks detected by an object detector to generate connected components for each of the objects of interest. These connected components are then used to generate blobs for the objects which contain their spatial and spectral information. This information includes the height and width of the object, its center of mass, and the first degree statistical estimation of its photometric appearance. The algorithm looks for the possibility of objects occluding each other. We call this event a collision. If no collision is occurring in the scene, the spectral data association process is performed to solve the blob correspondence problem and track individual objects.

If the algorithm detects the possibility of the collision, a multi-hypothesis data association is performed to find the occluding object. Since the visible photometric information for current frame does not represent the occluded object(s),



**Fig. 1.** The overview of the visual tracker using a spatio-spectral tracking mechanism

their model will not be updated until the collision has been resolved. Since there will be only one occluding object, a simple kernel based tracking process will be used to track it. The blob information for this object is then retrained and updated for tracking purposes after the collision has been resolved.

In the final step of the algorithm, a simple Kalman filter is performed on the center of the blobs. Employing a Kalman filter in order to track individual points is an efficient process and does not complicate the computation requirements of the algorithm. This step helps refine the objects' tracking trajectories and remove the jitters in the trajectories that might occur during the object detection stage on the foreground region centers.

## 2.1 The Visual Tracking Algorithm

We propose an efficient Spatio-Spectral Tracking module (SST) to track objects of interest in the video sequence. The detected foreground regions are processed further by employing a connected component processing in conjunction with a blob detection module to find objects of interest. These objects are tracked by their corresponding statistical models which are built from the objects' spectral (color) information. It is important to take note that the spatio-spectral coherency of tracked objects may be violated when two or more objects occlude each other.

A collision resolution mechanism is devised to address the issue of occlusion of objects of interest. This mechanism uses the spatial object properties such as their size, the relative location of their center of mass, and their relative orientations to predict the occlusion – i.e. collision.

**Blob detection and object localization.** In the blob detection module, the system uses a spatial connected component processing to label foreground regions. However, to label objects of interest a blob refinement framework is used to compensate for inaccuracies in physical appearance of the detected blobs due

```

Maintain the list of tracking objects: O-Lt-1[1 : n]
For new frame t containing the foreground masks
1. Detect the connected components
2. Perform morphological smoothing to detect contingent objects
3. Detect collision
   if no-collision:
4. Maintain the new object list: CC-Lt[1 : k]
5. if  $k > n$  determine if new objects are to be added to the new list
   if new objects then create = 1
   for  $i = 1 : k$ 
      5.1. Generate the following:
         CC-Lt[i].Center
         CC-Lt[i].width
         CC-Lt[i].height
         CC-Lt[i].appearance
   for all unassigned O-Lt-1 list objects
      5.2. find object O-Lt-1[j] : argmax [mean ( $p(CC-L_t[i]|O-L_{t-1})$ )]
      5.3. if probability is larger than threshold
         Assign: O-Lt[j] ← CC-Lt[i]
         Make object O-Lt[j] visible
      else
         Make object O-Lt[j] invisible
      5.4. if (create = 1)
         Assign: O-Lt[n + 1] ← CC-Lt[k]
         Make object O-Lt[n + 1] visible
6. if collision:
   Maintain colliding object list: CO-Lt[1 : k]
   for colliding objects:
      6.1. find CO-Lt[j] : argmax [mean ( $p(CO-L_t[i]|O-L_{t-1})$ )]
      6.2. find maximum probability among colliding list objects
         suspend update for all the other objects in colliding list
         perform mean-shift tracking on the occluding object
7. perform Kalman filter on the centers of visible objects

```

**Fig. 2.** The spatio-spectral object tracking algorithm

to unintended region split and merge, inaccurate foreground detection, and small foreground regions. A list of objects of interest corresponding to each detected blob is created and maintained to further process and track each object individually. This raw list of blobs corresponding to objects of interest is called the spatial connected component list. Spatial properties about each blob such as its center and size are kept in the spatial connected component list. The process of tracking individual objects based on their appearance and their corresponding spatial features is performed in the spatio-spectral tracking mechanism.

**Spatio-spectral tracking (SST) mechanism.** To track moving objects our proposed algorithm requires a model for individual objects. These "appearance models" are employed to search for correspondences among the pool of objects detected in new frames. Once the target for each individual has been found in the new frame they are assigned a unique ID. In the update stage the new information for only visible individual are updated.

Figure 2 show the pseudo-code of the proposed object tracking algorithm. Our modeling module represents an object with a set of statistical representation for its appearance. In the SST module a list of known objects of interest is maintained. During the tracking process the raw spatial connected component list is used as the list of observed objects. A statistical correspondence matching is employed to maintain the ordered objects list and track each object individually. The tracking module is composed of three components, appearance modeling, correspondence matching, and model update.

- **Appearance modeling.** Once each connected component is detected and processed their appearance models are generated. These appearance models

along with the objects location and first order geometric approximation produce an extended blob structure for the detected objects. In order to produce the geometric appearance of the detected objects, we use their corresponding connected components and geometric moments analysis. The 2-D geometric moments of a region are particularly important since they encode relevant visual and simple geometric features. In order to use these moments in computing geometric features of the objects in our work, we use the connected components. Along with these geometric features for the objects we extract orientation and their major and minor axis lengths. The objects' centers and their width are used in the process of collision detection.

The other component of the models of the objects in our algorithm is their photometric representation. Our current photometric appearance models are the first order statistical estimation of the probability density functions of pixel colors within the object.

- ***Correspondence matching.*** After the models are generated and objects are tracked in the previous frame at time  $t - 1$ , a correspondence matching mechanism is employed in the current frame to keep track of the objects at time  $t$ . Unlike many target representation and localization methods our mechanism takes a better advantage of the object detection. The traditional approaches usually ignore the foreground objects and search in a neighborhood of the object in the previous frame to find the local maxima for the presence of the object.

Foreground objects generated using the connected component process from the foreground image populate a finite list of un-assigned objects in the current frame. We call this list  $\{\text{CC-L}\}_t$  and the list of object appearance models from the previous frame  $\{\text{O-L}\}_{t-1}$ . The idea is to search on the un-assigned list of objects their corresponding blob (appearance model) from the previous frame. Notice that in our algorithm instead of a spatial search over a window around each object and finding the best target match, we perform the search over the object list in the new frame. This decreases the computational cost of the algorithm compared to the traditional methods.

The proposed matching algorithm works by starting from the current frames connected component list. Let's denote the  $i$ th object from this list as;  $\text{CC-L}_t(i)$ . The algorithm goes through the object models from the  $\{\text{O-L}\}_{t-1}$  list and finds the model which maximizes the likelihood of representing the  $\text{CC-L}_t(i)$ . If such model exists and is denoted by  $\text{O-L}_{t-1}(j)$  then:

$$\text{O-L}_{t-1}(j) = \arg \max_k [mean(P(\text{C-L}_t(i)|\text{O-L}_{t-1}(k)))] \quad : \forall k \quad (1)$$

***Collision resolution.*** In order for the system to be robust to collisions – when objects get too close that one occludes the other – the models for the occluded individual may not reliable for tracking purposes. Our method uses the distance of detected objects as a means of detecting a collision. After a collision is detected we match each of the individual models with their corresponding representatives. The one with the smallest matching score is considered to be occluded. The occluded object's model will not be updated but its new position is predicted

```

For new frame t
  1. Calculate the speed of the objects
  2. for each object pair
    2.1. predict the new object centers in the
          next frame using Kalman filter
    2.2. if ( the two objects overlap ) then Collision = 1
    2.3. else Collision = 0
  3. return Collision

```

**Fig. 3.** The collision resolution algorithm

by a Kalman filter. The position of the occluding agent is updated and tracked by a mean-shift algorithm. After the collision is over the spatio-spectral tracker resumes its normal process for these objects.

Figure 3 shows the algorithm which performs the collision detection and resolution. The collision detection algorithm assumes the center of the object and their velocity within consecutive frames are linearly related and the observation and measurement noises are normal with zero mean. Therefore, a Kalman filter can be used to predict the new locations of object centers. These information along with the width and height of the objects are used to predict the possibility of collision between multiple objects.

In our approach we assume that the discrete state of the objects is represented by their center of mass. Therefore, we have:  $\mathbf{x}(k) = [C_x(k), \dot{C}_x(k), C_y(k), \dot{C}_y(k)]^T$ , where  $[C_x(k), C_y(k)]$  is the center of the object at frame  $k$ . Since the measurements are taken every frame at discrete time-steps with the rate of 30 frames per second, it is important to be able to predict whether the objects will collide given the observation and measurement parameters in the current frame. We assume that the object centers undergo a constant acceleration from frame to frame with unknown rates. We also assume that between each time-step the acceleration is randomly distributed as a normal probability density function with zero mean and an unknown covariance matrix. The governing equation that rules the relationship of consecutive states is given by:  $\mathbf{x}(k+1) = \mathbf{F}\mathbf{x}(k) + \mathbf{G}a_k$ , where  $a_k$  is the constant acceleration from time  $k$  to  $k+1$ ,  $\mathbf{G} = [1/2 \ 1]^T$  is the acceleration vector and  $\mathbf{F}$  is the velocity matrix:

$$\mathbf{F} = \begin{bmatrix} 1 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

Since we assume that the acceleration is drawn from random white zero mean noise, the state equation will become a linear function  $\mathbf{h}$  affected by noise  $\mathbf{n}$ . Also we assume that the measurements are subject to a normal noise  $\mathbf{v}$  which is independent from the observation noise. Therefore:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{h}(\mathbf{x}(k)) + \mathbf{n} \\ \mathbf{z}(k) &= \mathbf{f}(\mathbf{x}(k)) + \mathbf{v} \end{aligned} \quad (3)$$

where  $n = \mathcal{N}(\mathbf{0}, \mathbf{Q})$  and  $v = \mathcal{N}(\mathbf{0}, \mathbf{R})$ . Since the state and measurement equations are linear and the noise is Gaussian, a Kalman filter can be used to predict

the location of the object centers in the new frames [13]. For each object pairs, a collision is about to occur if any of the following is true:

$$\begin{cases} C_1^{new} \cdot x < C_2^{new} \cdot x \Rightarrow C_1^{new} \cdot x + O_1 \cdot \frac{w_1}{2} \geq C_2^{new} \cdot x - O_2 \cdot \frac{w_2}{2} \\ C_1^{new} \cdot y < C_2^{new} \cdot y \Rightarrow C_1^{new} \cdot y + O_1 \cdot \frac{h_1}{2} \geq C_2^{new} \cdot y - O_2 \cdot \frac{h_2}{2} \\ C_1^{new} \cdot x > C_2^{new} \cdot x \Rightarrow C_1^{new} \cdot x - O_1 \cdot \frac{w_1}{2} \leq C_2^{new} \cdot x + O_2 \cdot \frac{w_2}{2} \\ C_1^{new} \cdot y > C_2^{new} \cdot y \Rightarrow C_1^{new} \cdot y - O_1 \cdot \frac{h_1}{2} \leq C_2^{new} \cdot y + O_2 \cdot \frac{h_2}{2} \end{cases} \quad (4)$$

where  $C_1$  and  $C_2$  are the center coordinates of each pair of objects and  $h$  and  $w$  is their respective height and width.

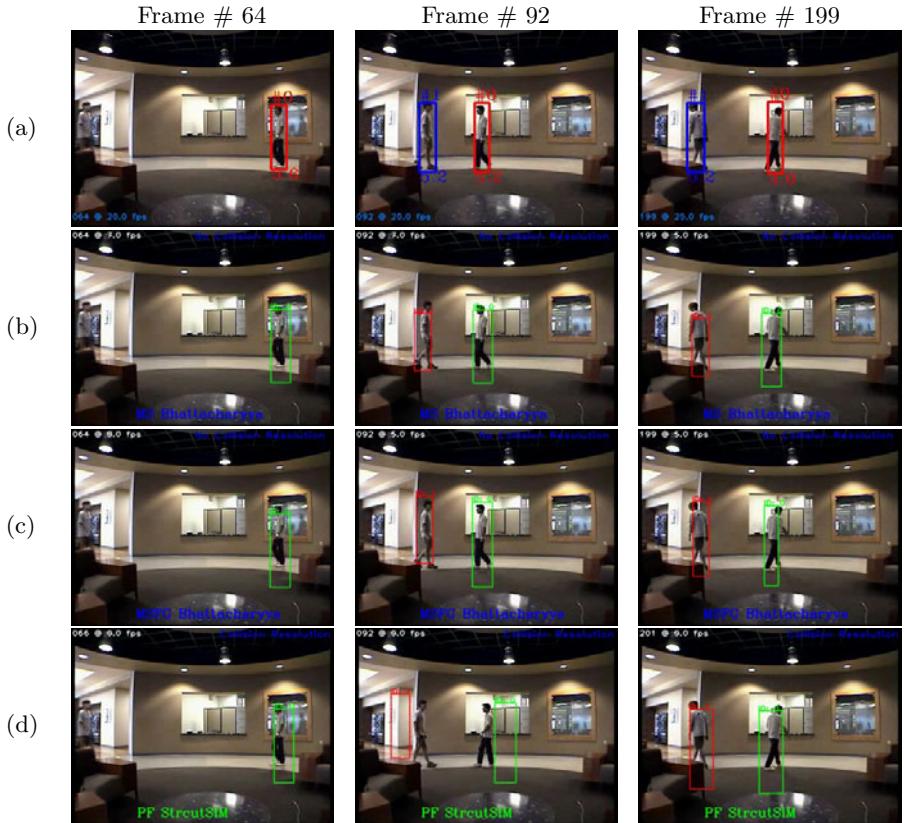
### 3 Experimental Results

In this section we compare the performance of the proposed technique in object tracking using several real video sequences that pose significant challenges. Also our algorithm's performance is compared with that of kernel-based tracking using Bhattacharyya Coefficient [4] with and without foreground detection and the structural-similarity measure of [10]. The performance evaluation consists of an evaluation of the frame rates of the systems the algorithms ability to resolve collisions and their robustness to changes in the objects' appearances.

**Frame rate and tracking speed.** As discussed in the previous sections the computational complexity of the proposed method is less than the existing object tracking algorithms. By limiting the search for potential targets to the linear list of detected objects in new frames, we decreased the search space. The advantage of this mechanism is the increased speed while tracking multiple objects. Figure 4 shows the performance of our method in terms of frame rate in comparison with two kernel based approaches as well as a particle filter algorithm.

Figure 4(a) shows the results of our proposed tracking algorithm while Figure 4(c)-(d) present the tracking results of the mean-shift algorithm [4], a mean-shift algorithm which uses the detected foreground object masks, and a particle filter with structural similarity measure [10], respectively. Our approach tracks the object with real-time performance of 20-25 frames per second (fps) while the single object tracking speed for the mean-shift algorithm is 7-10 fps (Figures 4(b) and (c)). By introducing the second object to the scene the mean-shift speed drop to 4-7 fps compared to 20-25 fps in our algorithm. The particle filtering approach tracking time is more than one second (4(d)).

**Tracking performance in the presence of collision.** Figure 5 presents the results of the proposed algorithm, the two mean-shift based methods, and the particle filter based algorithm on a video with two successive collisions. The figure shows two frames per collision, one taken before the collision and the other afterwards. Rows (a)-(d) represent the results of our approach, the mean-shift algorithm, the mean-shift algorithm with the foreground masks, and the particle filter technique, respectively. From the figure the proposed collision resolution mechanism in was able to effectively detect the collision and resolve it accordingly without the loss of object tracks. In this case the mean-shift and particle filter

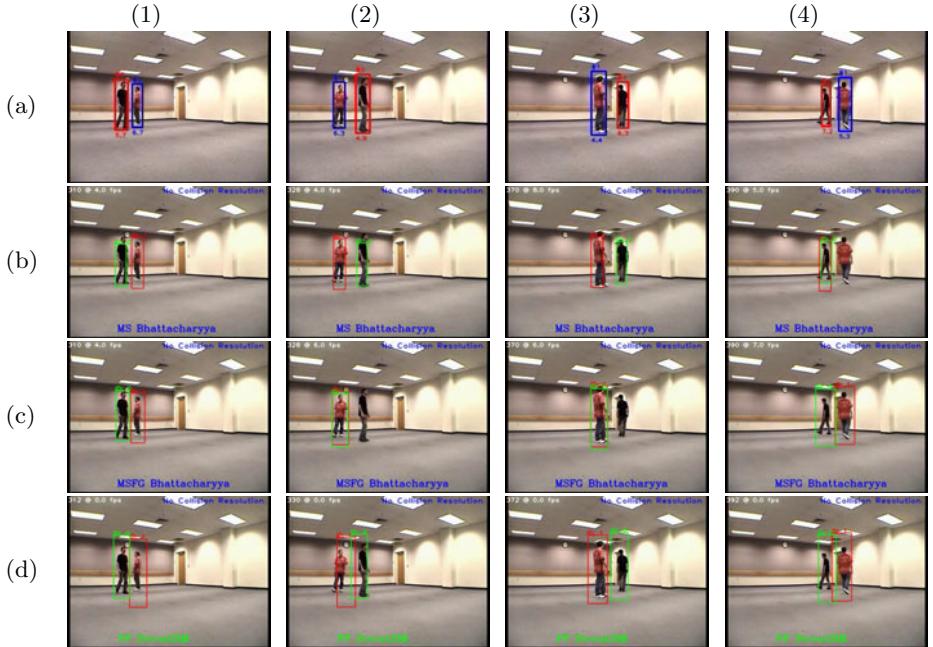


**Fig. 4.** Comparison of our visual object tracking algorithm with several traditional methods: (a) our method, (b) mean-shift algorithm, (c) mean-shift algorithm using foreground masks, (d) Particle Filter using structural similarity measure

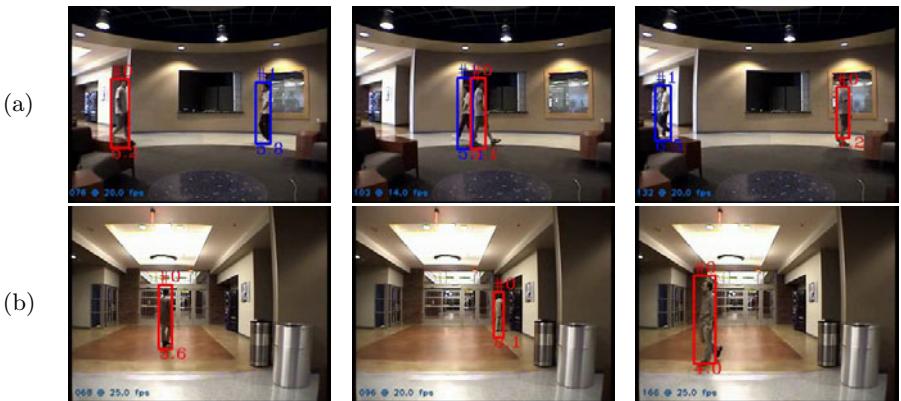
based methods -(b) and (d)- could also keep the tracking trajectories of the objects while the mean-shift algorithm which used the foreground regions -(c)- lost the track of occluding object.

By examining columns 3 and 4 we confirm that the proposed collision resolution mechanism in our approach was able to handle the second collision as well. However, the mean-shift algorithm in lost the track of the occluding object. Notice that the particle filter approach was robust to both collisions. However, as noted earlier this approach is very slow compared to our proposed technique and its tracking results are not as accurate as our algorithm.

**Other challenging experiments.** Figure 6 shows two challenging scenarios for visual object tracking algorithms which deal with target representation and localization. In Figure 6 three frames of videos taken in a dark lobby where lighting and reflective surfaces slightly change the objects' appearances in different locations. The proposed algorithm shows robustness to these challenge. The algorithm was also able to resolve the collision while the persons were passing.



**Fig. 5.** Comparison of our tracking algorithm with several traditional methods in the presence of collision: (a) our method, (b) mean-shift algorithm, (c) mean-shift algorithm using foreground masks, and (d) Particle Filter using structural similarity measure



**Fig. 6.** The tracking results of the proposed visual object tracker under challenging conditions: (a) illumination changes, and (b) reflective surfaces

## 4 Conclusions and Future Work

Based on the requirements of the real-time applications we proposed a non-parametric object tracking framework in this paper. Our approach takes advantage of the object detection process for tracking purposes. The detected objects

are used to generate photometric and geometric appearance models. These appearance models are employed to reduce the target localization search space. The experimental evaluation indicate that our technique is faster than kernel-based approaches and shows more scalability. The performance of the proposed tracking algorithm is also compared to particle filter based methods. The results obtained from our technique showed superior performance over the traditional methods. In addition, A collision detection and resolution mechanism is introduced to our object tracking framework. This modules is responsible for predicting the possibility of collision between the foreground masks of two or more objects. The collision resolution mechanism is tested in several scenarios and the results show significant improvement over the kernel-based methods.

In our current implementation the photometric appearances of the objects are estimated by a single degree statistical model. Another future direction to this work is to introduce more complex and accurate models for the objects' appearances to achieve more accurate tracking results.

## References

1. Greiffenhagen, M., Comaniciu, D., Neumann, H., Ramesh, V.: Design, analysis and engineering of video monitoring systems: An approach and a case study. Proceedings of the IEEE 89, 1498–1517 (2001)
2. Bradski, G.R.: Computer vision face tracking as a component of a perceptual user interface. In: IEEE Workshop on Applications of Computer Vision, pp. 214–219 (1998)
3. Handman, U., Kalinke, T., Tzomakas, C., Werner, M., von Seelen, W.: Computer vision for driver assistance systems. In: Proceedings of SPIE, vol. 3364, pp. 136–147 (1998)
4. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. IEEE Trans. Pattern Anal. Mach. Intell. 25, 564–575 (2003)
5. Bar-Shalom, Y.: Tracking and data association. Academic Press Professional, Inc., San Diego (1987)
6. Kitagawa, G.: Non-gaussian state-space modeling of nonstationary time series. Journal of American Statistical Association 82, 1032–1063 (1987)
7. Gordon, G., Salmond, D., Smith, A.: A novel approach to non-linear and non-gaussian bayesian state estimation. Proceedings of IEEE 140, 107–113 (1993)
8. Rabiner, L.: A tutorial on hidden markov models and selected applications in speech recognition. Proceedings of IEEE 77, 257–285 (1989)
9. Tuzel, O., Porikli, F., Meer, P.: Learning on lie groups for invariant detection and tracking, Minneapolis, MN, pp. 1–8 (2008)
10. Loza, A., Mihaylova, L., Bull, D., Canagarajah, N.: Structural similarity-based object tracking in multimodality surveillance videos. Mach. Vision Appl. 20, 71–83 (2009)
11. Zhou, H., Yuan, Y., Westover, C.S.: Object tracking using SIFT features and mean shift. Computer Vision and Image Understanding 3, 345–352 (2009)
12. Tavakkoli, A., Nicolescu, M., Bebis, G.: Efficient background modeling through incremental support vector data description. In: Proceedings of the 19th International Conference on Pattern Recognition (2008)
13. Broida, T., Chellappa, R.: Estimation of object motion parameters from noisy images, vol. 8, pp. 90–99 (1986)

# Driving Fatigue Detection Using Active Shape Models

Hernán García<sup>1</sup>, Augusto Salazar<sup>1,2</sup>, Damián Alvarez<sup>1</sup>, and Álvaro Orozco<sup>1</sup>

<sup>1</sup> Grupo de Investigación en Control e Instrumentación, Universidad Tecnológica de Pereira, La Julita, Pereira, Colombia

<sup>2</sup> Universidad Nacional de Colombia - Sede Manizales, Manizales, Km 7 vía al aeropuerto la Nubia

aesalazarj@unal.edu.co, {damianalvarez,aaog}@utp.edu.co

**Abstract.** Driver fatigue is a major cause of traffic accidents. The fatigue detection systems based on computer vision have great potential given its property of non-invasiveness. Major challenges that arise are fast movements of eyes and mouth, changes in pose and lighting variations. In this paper an Active Shape Model is presented for facial features detection of features extracted from the parametric model Candide-3. We describe the characterization methodology from parametric model. Also quantitatively evaluated the accuracy for feature detection and estimation of the parameters associated with fatigue, analyzing its robustness to variations in pose and local variations in the regions of interest. The model used and characterization methodology showed efficient to detect fatigue in 100% of the cases.

## 1 Introduction

Currently, a high number of traffic accidents are mainly caused by driver fatigue. Driver fatigue detection has been the center of attention of a lot of paper work [1], that have as goal an the traffic accidents decrease. In the last decade there have been developed monitoring systems that allow to detect driving fatigue and also to alert the driver by using different techniques. However finding an efficient way to detect constantly fatigue is been one of the most important issues to find out. Over the past several decades, much research has been conducted on human fatigue prevention, focusing on two main thrusts. The first one is to understand the physiological mechanism of human fatigue and how to measure fatigue level [2], [3]. The second thrust focuses on developing human fatigue monitors for commercial transportation based on the achievements from the first effort [4]. So far, these fatigue monitoring systems can be classified into two categories, which are: 1) measuring the extent and time-course of loss of driver alertness and 2) developing real-time in-vehicle drowsy driver detection and alerting systems.

However, human fatigue results from a very complicated mechanism, and many factors affect fatigue in interacting ways [4], [5]. Up to now, the fatigue mechanism is still not well understood, and few of these existing fatigue monitors

are effectively used in the real world. The main drawback of them is that most of the present safety systems only acquire information from limited sources (often just one). Therefore, as pointed by some researchers [6], [7], many more efforts are still needed to develop systems for fatigue prevention.

This paper work proposes a fatigue detection system based on the computer vision techniques, where they are being analyzed by the visual answers of the driver face, from the variations that present the facial features, specially those in regions of the eyes and mouth. Those features are detected by using holistic techniques as ASMs proposed on [8], which from the a-priori object knowledge (face to be analyzed) and being help by a parametric model (Candide-3 in this case), allow to estimate the object shape with a high precision level. ASMs bring the advantage of handle with problems as noise, occlusions, illumination changes and elements that could add variations to the analyzed image background [9]. From the detected features, it is possible to perform eye closure and opening mouth rank measurement, to calculate the fatigue associated parameters as PERCLOS, AECS and YawFrec [10], [11], [12].

## 2 Building and Fitting an ASM

### 2.1 Building the ASM

The process of building an ASM can affect its performance during the alignment of an ASM to an input image. An ASM is derived from a set of training images, each represented by a set of  $n$  landmarks on characteristic facial features [8], [9].

Training of an ASM is accomplished by:

- i* Landmarking the training set
- ii* Deriving the shape model

**Landmarking The Training Set:** The initial step is the selection of images to incorporate into the training set. The final use of the model must be taken into account at this point. When deciding which images to include in the training set, the desired variations must be considered. These variations must be present within the training set, else the ASM will not incorporate those specific modes. However, it is also possible to *overtrain* the model [8] by including too many images that are similar to the mean shape.

In order to include the whole region of the labeled face, there will be used the Candide-3 [13], which consist in a group of 113 points that depict the whole face regions in detail (eyes, nose, mouth, chin, etc), as shown in Figure II.

**Deriving the Shape Model:** To obtain the mean shape and its modes of variation, the images in the training set firstly need to be aligned with respect to a common set of axes. *Procrustes analysis* [8], [14] accomplishes this, creating shapes that are independent of translation, rotation and scaling. This allows statistical analysis for variations due to the nature of the face itself.

Principle component analysis (PCA) is performed on the set of shapes, to obtain the eigenvectors  $\mathbf{p}_k$  and eigenvalues  $\lambda_k$  represent the modes of variation and their significance in the training set. The original data can be approximated by the first  $t$  modes, and the training set can be obtained by [8], [9]

$$\mathbf{x} = \bar{\mathbf{x}} + \mathbf{P}\mathbf{b} \quad (1)$$

Where  $\bar{\mathbf{x}} = (x_1, y_1, \dots, x_n, y_n)^T$  is the mean shape (a vector of mean landmark coordinates),  $\mathbf{P} = (\mathbf{p}_1, \mathbf{p}_2, \dots, \mathbf{p}_t)^T$  is the matrix of the first  $t$  eigenvectors and  $\mathbf{b} = (b_1, b_2, \dots, b_t)^T$  is a vector of shape parameters.

## 2.2 Fitting the ASM

An example of a model in an image is described by the shape parameters,  $\mathbf{b}$ , combined with a transformation from the model co-ordinate frame to the image co-ordinate frame. Typically this will be a Similarity transformation defining the position,  $(X_t, Y_t)$ , orientation,  $\theta$ , and scale,  $s$ , of the model in the image [8].

The positions of the model points in the image,  $\mathbf{x}$ , are then given by:

$$\mathbf{x} = T_{X_t, Y_t, s, \theta}(\bar{\mathbf{x}} + \mathbf{P}\mathbf{b}) \quad (2)$$

Where the function  $T_{X_t, Y_t, s, \theta}$ ; performs a rotation by  $\theta$ , a scaling by  $s$  and a translation by  $(X_t, Y_t)$ .

To find the best pose and parameters so that from the model estimate a new  $\mathbf{Y}$  point set in the image, there must be minimized the square of the distance between the model and the image points as the Equation 3 [9].

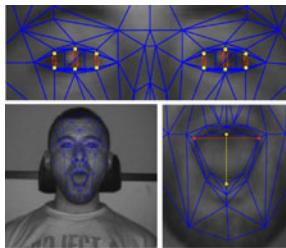
$$|Y - T_{X_t, Y_t, s, \theta}(\bar{\mathbf{x}} + \mathbf{P}\mathbf{b})|^2 \quad (3)$$

## 3 Fatigue Characterization

As mentioned in the earlier section, ASM model will be used to estimate the shape of the subject's face and with this get the points that make up the parametric model Candide-3. These points will be used to determine the movements and variations in shape of the regions of eyes and mouth, on which there will be realized the measurement of three parameters PERCLOS, AECS and YawnFrec.

PERCLOS and AECS are measurements that characterize the movement of the eyelids. PERCLOS has been already validated and it has been that is the parameter most adapted to detect the fatigue [15]. AECS is a good indicator of fatigue, and it has been defined as the necessary quantity of time to close or open completely the eyes. The degree of eye closure is characterized by the shape of the pupil, it has been observed that when the eyes are closed, the pupil is occluded by the eyelids doing that his form makes to itself more elliptical. The degree of eye closure is computed as the ratio between the axis of the pupil's ellipse and with this one a record takes in the time to obtain the PERCLOS [16]. Otherwise, studies such as [17], have shown that a person tired AECS is typically different from that of an alert person.

A fatigued person is characterized by few expressions show because there is minimal activity of facial muscles, with the yawning or opening the mouth the most common expression. Monitoring the movements of the lips can be detected on open mouth position, provided that the features around the mouth to deviate from its closed configuration. The opening of the mouth is computed as the ratio between of its height and width. This measurement is used to estimate the YawFrec. The Figure 1 shows the landmarks chosen to analyze the subject's fatigue.



**Fig. 1.** Overall description of the face using the model Candide-3 and landmarks chosen to represent the ocular and mouth regions

### 3.1 Eyelid Movement Characterization

To calculate PERCLOS and AECS on [17] y [18] there has been propose to follow the pupil steadily and determinate the eye closure rank in an accumulative way on time, by using the axis reason on the pupil ellipse. An individual eye closure is defined as the difference of time between two moments to which the pupil size is 20% or less compared to the normal size. One individual closure velocity is defined as the time period where the pupil size is between 20% and 80% compared to the nominal pupil size.

In order to realize those measurements, it is proposed to apply the described methodology on [18], with the difference not to be calculated the eye closure rank using its ellipse reason, but using the eye vertex defined on the Candide-3 model. More specifically using the 98, 54, 106, 100, 55 and 108 eyelid vertex for the right eye and 105, 21, 97, 107, 22 and 99 for the left one.

This way the eye closure rank is calculated by

$$C_{RE} = \frac{d(98 - 100) + d(54 - 55) + d(106 - 108)}{3} \quad (4)$$

$$C_{LE} = \frac{d(105 - 107) + d(21 - 22) + d(97 - 99)}{3} \quad (5)$$

From the Equations 4 and 5, when the eye closure rank is less or equal to 20% of maximum distance between the eyelid, it is considered that the eyes are closed. According to the work accomplish on [19], if the eyes are close during 5 consecutive frames, it could be considered as the diver falling sleep.

### 3.2 Lips Movement Characterization

To calculate the mouth opening frequency, is necessary to know the mouth opening rank, which is represented by the mouth's high and the width reason. The mouths high is represented by the distance between upper lip and down lip, and the mouths width is represented by the distance between the left corner and the right one. The opening rank graphic is known as the YawnFrec and this can be seen as peaks yawns

To perform the mouth opening rank measurement, it is proposed to use the mouth vertexes gotten by the Candide-3 model. More specifically there must be use the vertexes that would define the mouth extremes (right 64, left 31, up 7 and down 8).

The mouth opening is defined as

$$OpenMouth = \frac{d(7 - 8)}{d(64 - 31)} \quad (6)$$

Through the work on [20], if the mouth opening rank is above 0.5 in more than 20 consecutive frames, it could be consider as the driver yawning.

## 4 Experimental Setup

### 4.1 Data Base

The base used in this work, it was acquired in Control and Instrumentation Laboratory of the Technological University of Pereira, Colombia. Which has recordings of 5 subjects on alert and fatigued estate to measure the presence of fatigue in the subjects. Also was recorded a protocol video for each subject, in which to consider making changes in pose, blinks and yawns controlled, to measure the accuracy with which the ASM model estimates the facial features. These videos were acquired with an OptiTrack infrared camera at a sampling rate of 25 frames per second and a resolution of  $640 \times 480$  pixels. It is noteworthy that the algorithms were executed on a *HP Dv1413* notebook with an *AMD Athlon X2* processor 2.1Ghz and 3GB of RAM.



**Fig. 2.** A sample of the to analyze

## 4.2 Evaluation Metrics

The shape model will be estimated for every subject, contemplating 5 changes of pose exposure ( $-30^\circ$ ,  $-15^\circ$ ,  $15^\circ$  and  $30^\circ$  in the axis  $Y$  and  $25^\circ$  in the axis  $X$  to simulate the nods) and 1 frontal, with which 25 images will be had for changes of pose exposure and 25 frontal images. The first step is to compute the average error of the distance between the manually labeled points  $\mathbf{p}_i$  and points estimated by the model  $\hat{\mathbf{p}}_i$ , for all training and test images as:

$$\text{Error} = \frac{1}{N_I N_{pts}} \sum_{i=1}^{N_{pts}} \|\mathbf{p}_i - \hat{\mathbf{p}}_i\| \quad (7)$$

Where  $N_I$  is the number of manually labeled images and  $N_{pts}$  is the number of points to be estimated.

To perform a quantitative analysis of the accuracy in adjusting of the ASM, we calculate the relative error between the manually labeled points and points estimated by the model for the region of the eyelids. The points of this region are described in Section 3.1 by obtaining a set of 6 points. The relative error is computed by:

$$\text{Rerr} = \max(d_{l98}, d_{l100}, d_{l54}, d_{l55}, d_{l106}, d_{l108}) / d_{l53-56} \quad (8)$$

Where  $d_{l98}$  is the euclidean distance between the vertex 98 manually labeled and vertex estimated, and so on; also  $d_{l53-56}$  is the euclidean distance between the vertexes 53 and 56 manually labeled, which gives the width of the left eye. following the lead of the criterion presented in [21], in which if  $\text{Rerr} = 0.25$ , the match of the model to the face is considered to be succesful. That is why for  $\text{Rerr} = 0.25$ , the maximum argument of the Equation 8 is equivalent to  $1/4$  of the width of the eye. Therefore, the detection rate for a DataBase of a set with  $N$  images, is defined as:

$$R = \sum_{i=1}^N \frac{i}{N} \times 100, \text{Rerr}_i < 0.25 \quad (9)$$

## 5 Results

### 5.1 Error in the Estimation of the Shape Model

From the measurements described in section 4.2, we calculate the accuracy with which the ASM estimated the characteristic points of face.

In Table II, It can be seen that although the accuracy in the estimation of the points is greater for images of the training set, the average error is also small for the test images. This owes to a rigorous procedure in the training and model building in which there were considered to be the biggest quantity of possible

**Table 1.** Average estimation error

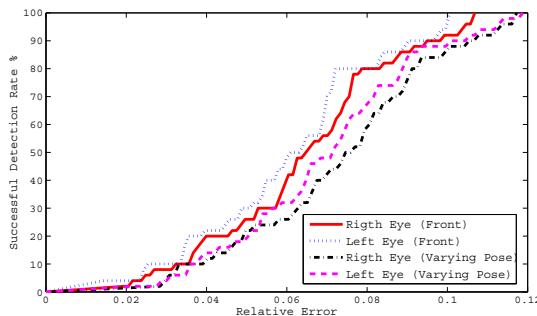
	N	Total Set		Front		Pose Changes	
		Error [pix]	T [ms]	Error [pix]	T [ms]	Error [pix]	T [ms]
Training	25	$2.4882 \pm 0.51$	20.8	$2.2569 \pm 0.42$	19.7	$2.5807 \pm 0.5$	20.3
Test	25	$3.1438 \pm 0.49$	21.3	$3.0668 \pm 0.67$	19.3	$3.2373 \pm 0.53$	20.7

forms to estimate. Moreover, it is noted that although the average error for images with changes in pose is a bit higher than in the case of frontal images, indicates that the accuracy in estimating the model is so high for images that show changes pose of the face. Also, it is of highlighting that the times average of estimation of the model ASM, they are relatively small which would help in applications on line.

## 5.2 Distribution on the Relative Error

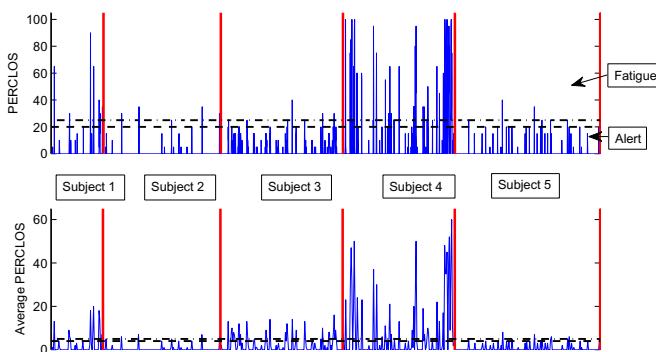
Figure 3 shows the distribution function of the relative error against successful detection rate, on which it is observed that for a relative error of 0.1 in the case of the adjustment of the right eye and 0.135 for the left eye in frontal images, the detection rate is 100%, indicating that the accuracy in the ASM model fit is high. In addition to images with pose variations is achieved 100% for adjusting the eye region for relative errors to 0.18 and 0.19; being this much lower than the established criterion of 0.25.

It is considered that the criterion  $Rerr < 0.25$  is not adapted to establish a detection like correct and can be not very suitable when it is desirable to realize facial features detection in images with minor scale. That's why it is considered to be a successful detection for errors  $Rerr < 0.15$  [22]. Based on this, it shows itself that the model ASM used in this work is efficient and fulfills this request.

**Fig. 3.** Relative error versus detection rate for DataBase images

### 5.3 Fatigue Estimation

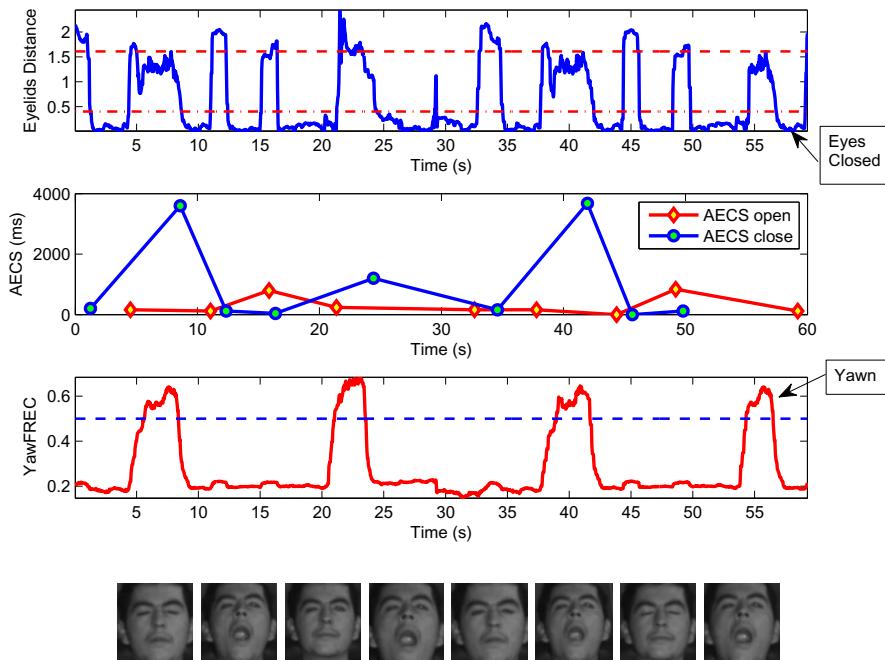
Figure 4 shows the estimation results PERCLOS for recordings of the subjects present in the DataBase in a state of fatigue, which together account for about 1 hour. From this figure each of the different subjects shows moments of fatigue (above the upper threshold) in more than one moment, being the subject 5, continued of 2, with fewer moments of fatigue and also a shorter, compared with other subjects that show as many moments of fatigue. This behavior is best seen in the figure of PERCLOS average, suggesting a more general behavior and where it is also quite clear that is the subject 4 that has a lower level of alert, for the reasons given above.



**Fig. 4.** PERCLOS analysis for five subjects in the DataBase

A quantitative analysis of the measurement of parameters associated with fatigue such as the PERCLOS and average PERCLOS can be seen in Figure 5. In which there appears a segment of 60 seconds of recording of a subject under influence of fatigue; for which clearly shows the state of fatigue due to repeated closures of eyes for a period of time (greater than one second). Besides presenting some very high values of AECS indicating that the subject is entering a state of somnolence. This finding is bolstered by looking at the graph YawFrec, which occur frequently a lot of yawns by the subject at the time of recording.

Continuing the analysis of the subject presented in Figure 5, was detected 100% of the eyes and mouth movements, presented by the subject; indicating that the accuracy of the characterization of fatigue supported by the ASM model is very high, because it presents a high performance in detecting movements of eyes and mouth regions, such as blinks and yawns. This is reflected in the good performance in the estimation of fatigue. Besides the time it takes the whole fatigue detection system is an average of 28.2ms for each frame analyzed, indicating that can be implemented in real-time.



**Fig. 5.** Close Eyes, AECS and YawFrec for a fatigued subject of DataBase

## 6 Conclusions

This paper was presented, the characterization of fatigue based on the parametric model Candide-3. The analyzed regions correspond to the eyes and the mouth on which detection and tracking of features is done using ASM. The results show that the estimation of the points is exact and complies with the requests for this type of systems. Through quantitative analysis evaluated the robustness of the ASM model in feature detection, which is maintained in nominal pose for a range of between  $[-30^\circ - 30^\circ]$  on Y and  $[0^\circ - 25^\circ]$  for X. The used model and the methodology of characterization showed efficiency to detect the fatigue in 100% of the evaluated cases. In addition, due to high accuracy in detecting and characterizing features proposed to estimate parameters associated with fatigue as the PERCLOS, AECS and YawFrec to determine the presence of this, the designed system has great potential for detect fatigue in the early stages, being of great interest in vial research prevention.

## References

1. Hancock, P.A., Verwey, W.B.: Fatigue, workload and adaptive driver systems. *Accid. Anal. Prev.* 29, 495–506 (1997)
2. Crew factors in flight operations xiii: A survey of fatigue factors in corporate/executive aviation operations

3. Sherry, P.: Fatigue countermeasures in the railroad industry: Past and current developments. Aar press, Washington (2000)
4. Zhu, Z., Ji, Q., Lan, P.: Real time non-intrusive monitoring and prediction of driver fatigue. *IEEE Trans. Veh. Technol.* 53, 1052–1068 (2004)
5. Co, E.L., Gregory, K.B., Johnson, M.J., Rosekind, M.R.: Crew factors in flight operations xi: A survey of fatigue factors in regional airline operations. NASA, Ames Res. Center, Moffett Field, CA, Tech, Rep. NASA/TM-1999-208799 (1999)
6. Zhang, Z., Zhang, J.S.: Driver fatigue detection based intelligent vehicle control. In: ICPR 2006: Proceedings of the 18th International Conference on Pattern Recognition, Washington, DC, USA, pp. 1262–1265. IEEE Computer Society, Los Alamitos (2006)
7. Hartley, L., Australia, National Road Transport Commission University Melbourne: Review of fatigue detection and prediction technologies / prepared by Laurence Hartley.. [et al.]. National Road Transport Commission, Melbourne (2000)
8. Cootes, T.F., Taylor, C.J., Copper, D.H., Graham, J.: Active shape models-their training and application. *Computer Vision and Image Understanding* 61, 38–59 (1995)
9. Cootes, T.F., Taylor, C.J., Manchester, M.P.: Statistical models of appearance for computer vision (2004)
10. Rajinda, S., David, H., Vanderaa, B., Halgamuge, S.: Driver fatigue detection by fusing multiple cues. In: ISNN 2007: Proceedings of the 4th International Symposium on Neural Networks, pp. 801–809 (2007)
11. Smith, P., Shah, M., da Vitoria Lobo, N.: Determining driver visual attention with one camera. *IEEE Transactions on Intelligent Transportation Systems* 4, 205–218 (2003)
12. Bergasa, L.M., Nuevo, J., Sotelo, M., Barea, R., Lopez, M.: Real-time system for monitoring driver vigilance. *IEEE Transactions on Intelligent Transportation Systems* 7, 63–77 (2006)
13. Ahlberg, J.: Candide-3 an updated parameterized face. Report No. LiTH-ISY-R-2326, Dept. of Electrical Engineering, Linköping University, Sweden (2001)
14. Larsen, R.: Functional 2D procrustes shape analysis. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) SCIA 2005. LNCS, vol. 3540, pp. 205–213. Springer, Heidelberg (2005)
15. Dinges, D.F., Mallis, M., Maislin, G., Powell, J.: Evaluation of techniques for ocular measurement as an index of fatigue and the basis for alertness management (1998)
16. Zhu, Z., Ji, Q., Lan, P.: Real time non-intrusive monitoring and prediction of driver fatigue. *IEEE Trans. Veh. Technol.* 53, 1052–1068 (2004)
17. Ji, Q., Yang, P.: Real time visual cues extraction for monitoring driver vigilance. In: ICVS, pp. 107–124 (2001)
18. Ji, Q., Yang, X.: Real-time eye, gaze, and face pose tracking for monitoring driver vigilance. *Real-Time Imaging* 8, 357–377 (2002)
19. Dong, W.H., Wu, X.: Driver fatigue detection based on the distance of eyelid. In: IEEE Int. Workshop VLSI Design and Video Tech Suzhou, pp. 28–30 (2005)
20. Wang, T., Shi, P.: Yawning detection for determining driver drowsiness. In: Proceedings of 2005 IEEE International Workshop on VLSI Design and Video Technology, pp. 373–376 (2005)
21. Hassaballah, M., Ido, S.: Eye detection using intensity and appearance information. In: IAPR Conference on Machine Vision Applications (2009)
22. Song, J., Chi, Z., Liu, J.: A robust eye detection method using combined binary edge and intensity information. *PR* 39, 1110–1125 (2006)

# Outlier Removal in Stereo Reconstruction of Orbital Images

Marvin Smith<sup>1</sup> and Ara Nefian<sup>2</sup>

<sup>1</sup> University of Nevada, Reno

<sup>2</sup> NASA Ames Research Center

**Abstract.** NASA has recently been building 3-dimensional models of the moon based on photos taken from orbiting satellites and the Apollo missions. One issue with the stereo reconstruction is the handling of “outliers”, or areas with rapid and unexpected change in the data. Outliers may be introduced by issues such as shadows on the surface, areas with low amounts of surface detail, or flaws in the camera systems. These errors may result in elevation spikes which cause the model to differ significantly from accurate ground truth. We are seeking to remove outliers from reconstructions by using a pair of filters which target the characteristics of these outliers. The first filter will use edge detection to filter areas with low detail and the second filter will remove areas in the disparity map which differ too far from their surrounding neighbors.

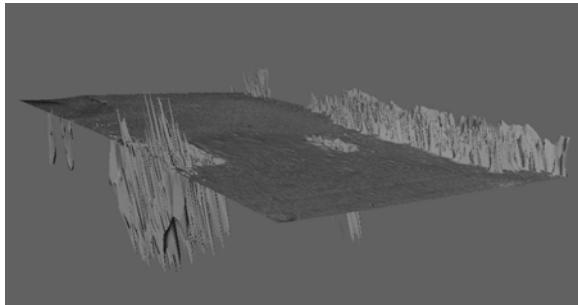
## 1 Introduction

The NASA Vision Workbench and the Ames Stereo Pipeline have created accurate, high resolution 3D maps of the Lunar surface. These maps support a variety of NASA missions as well as commercial products. Examples include finding future landing sites for human missions, developing computer-based landing systems [1], and Google Moon. One aspect of the Ames Stereo Pipeline that we are looking to improve upon is in the removal of outliers, or areas of the disparity map which are extreme to the points around them.

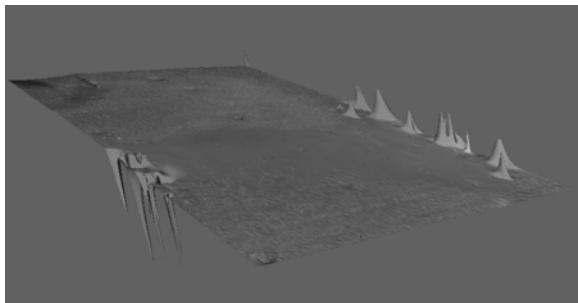
Outliers can cause unsightly and inaccurate results on a disparity map when ignored. Figure [1] shows the effect of outliers on an image. Also shown is the effect of the current outlier removal system in Figure [2].

Currently, there is a system in place which seeks to filter outliers. This system relies on a morphological erosion-like operation which is described in the following steps.

- For each pixel in disparity map  $D(i, j)$ 
  - Search a surrounding window of  $M \times N$  pixels
  - Compare each pixel in window to input pixel s.t.  $|D(i, j) - D(i - m, j - n)|$
  - If difference is greater than threshold, increment count of invalid pixels
- If invalid pixels divided by total pixels is greater than threshold, invalidate pixel
- Else, return original pixel



**Fig. 1.** Original image



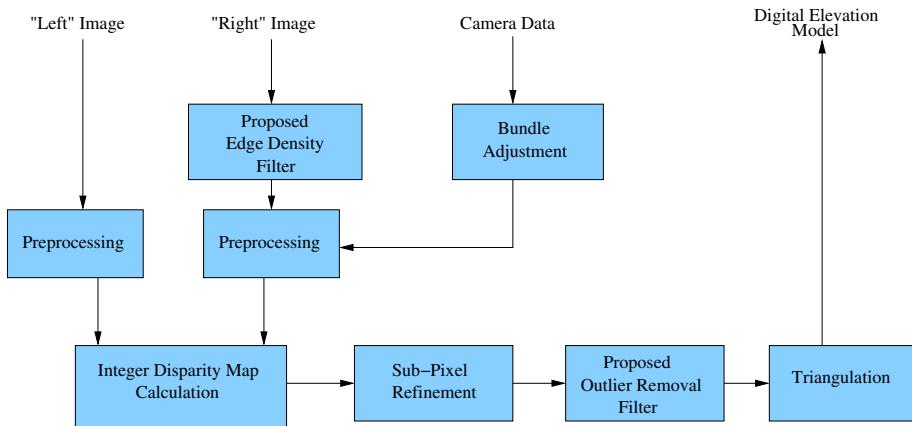
**Fig. 2.** Result of the current method

The current method occurs after the sub-pixel refinement stage [2]. Other outlier rejection techniques were analyzed and found not to fit within the framework of the Ames Stereo Pipeline [3][4].

There are several aspects of the original method which we seek to improve. The first issue is that there is not a preprocessing method to invalidate regions which have little to no detail. This method does not directly address this issue itself either as regions which have no detail, may appear the same, therefore giving the appearance of the same disparity. Another challenging alternative is an outlier being created when the pixel correlator takes a pixel inside a shadow and is required to guess which pixel is the corresponding match. As the correlation window searches for a match [5], if it searches over an area where many of the pixels are similar, there is no guarantee as to where in the window a match may be found.

The next issue to improve is the iterative process which treats every pixel in the disparity map as a logical value based on the difference in disparity from the center of the window. This may incorrectly invalidate regions which have rapid change, but a consistent gradient. If the pixels inside the window are different from the pixel of interest beyond a threshold, the pixels will be added to the invalid count. A process which compares the pixel to the window *average* would allow for more flexibility.

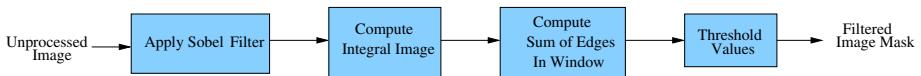
These solutions will be included in different phases. The Edge Density filter will be located at the start of this process. It will filter the initial image. Figure 3 shows the proposed changes to the Ames Stereo Pipeline.



**Fig. 3.** The Ames Stereo Pipeline with the proposed outlier filtering techniques

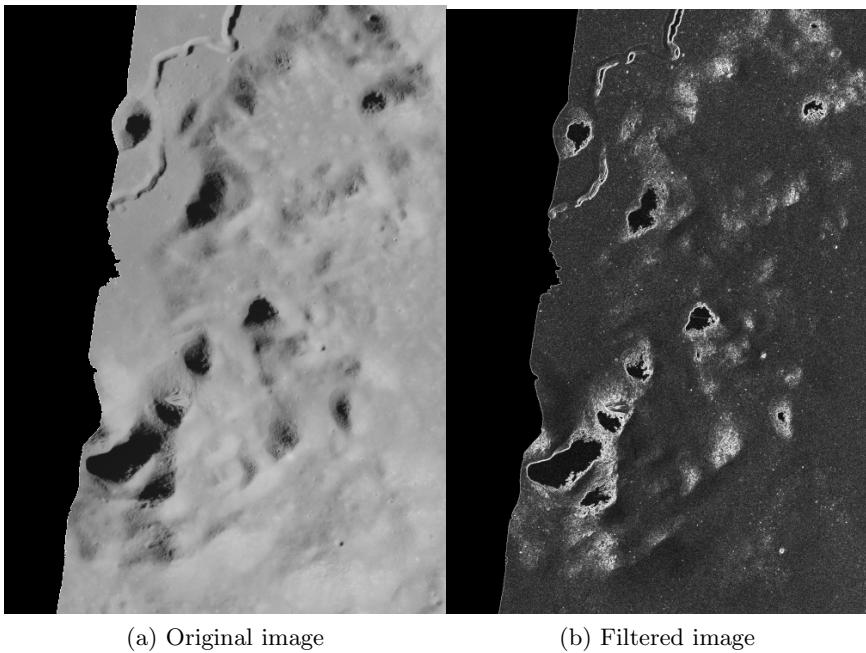
### 1.1 Edge Density Filter

The Edge Density filter is a preprocessing filter which is designed to remove regions of the image which have little to no extractable detail. This filter will be applied to the original image before being sent to the Integer Disparity Map module of the Ames Stereo Pipeline. A general outline of the process is shown in Figure 4.



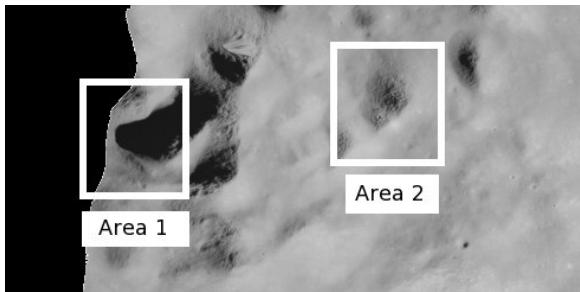
**Fig. 4.** The Edge Density Filter pipeline

The first and primary technique is to perform the Sobel edge detector [6]. The purpose is to describe and quantify the amount of detail or change thereof. Once we compute the edges, we compute the integral image [4]. This will allow us to rapidly and efficiently compute the sum of edges around a window for every pixel. Due to the size of the images and the large landscapes that are being dealt with, window sizes around 35 pixels are used. Next divide each result by the area of the window. Once these values have been computed, we apply a final threshold to create a binary mask. The binary mask will be intersected with the results from the sub-pixel correlation to prevent the invalidated regions from being computed in the triangulation module. Figure 5 shows an example output of the Edge Density Filter prior to thresholding.



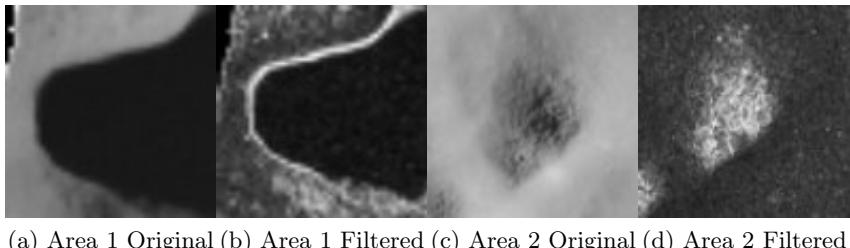
(a) Original image

(b) Filtered image

**Fig. 5.** Effect of Edge Density Filter prior to thresholding**Fig. 6.** Areas under investigation

What is relevant about this example is that regions which are completely black correspond to areas inside a shadow or have extremely low detail. Also note the distinction between shadows which persist over large regions such as craters, and small shadows which relate to the rapid elevation change of a hill or other natural surface. In Figure 6, regions marked *Area 1* and *Area 2* show two types of shadows which we want to analyze with respect to the Edge Density Filter.

*Area 1*, as seen in Figure 7a and Figure 7b, is a large blanketing shadow which shows no discernible detail inside of it. This will cause undesirable effects when



(a) Area 1 Original (b) Area 1 Filtered (c) Area 2 Original (d) Area 2 Filtered

**Fig. 7.** Effect of the Edge Density Filter on regions investigated in Figure 6. Note that both regions (a) and (c) are covered inside a shadow, yet the filtering effects are noticeably different (b & d). Area 1 is completely occluded inside the shadow, whereas Area 2 has a much lighter covering. The resulting contrast from the shadow in (c) enhances the detail, making the filtering results (d) possibly even stronger than regions not in a shadow.

correlated. In comparison, the shadow in Area 2 as seen in Figure 7c and Figure 7d is very unique and has much detail. The Edge Density Filter is very effective for this purpose as shown in Figure 7.

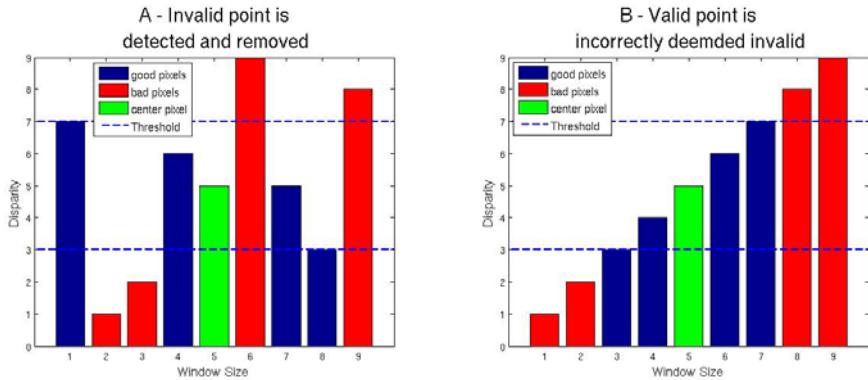
As a result, different types of shadows can result in completely opposite reactions. This is advantageous for comparing shadows of different structures.

## 1.2 Outlier Removal Filter

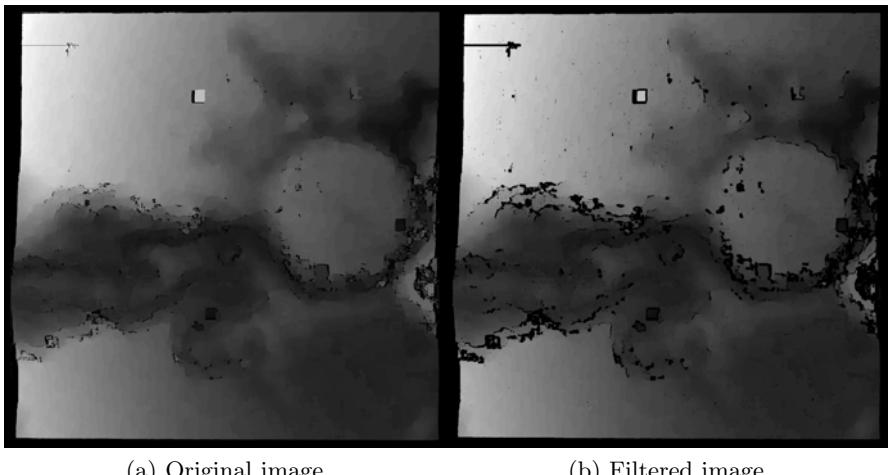
The next phase is the outlier removal filter. The outlier removal filter will search the disparity map and invalidate pixels where the disparity between itself and its surrounding average are different beyond a threshold. This is similar to the original method described in Section 1, but differs in that we want to compare it to the average inside the window and not compare it to each pixel independently. The difference between these methods become apparent in Figure 8.

Another strength of computing an average is that it is more likely to find and remove turbulent areas much like regions in Figure 11. If there is rapid change in a large window, the threshold may allow a number of outliers to validate a region. This makes the average much more secure to threats as the remaining ‘holes’ would affect the average value.

The first step of this process is to convert the horizontal and vertical disparities into a single scalar magnitude. This will allow for a single pass through the image and treats the separate disparities as a combined result. Once the scalar magnitude is computed, an integral image is created and the density of the disparities is calculated. This is very similar to the process outlined in the Edge Density Filter in Section 1.1. Finally, for each pixel, compare the scalar magnitude of the disparities to the results of the disparity density. If the difference is beyond a threshold, invalidate the pixel. Figure 9 shows the results from the new filter.



**Fig. 8.** Example of pixels rejected using the current method. (A) is an obvious outlier. The range of data shows no pattern and has extreme pixel values, therefore the system works. In (B), the rapid but constant increase in disparity could mean a legitimate increase in depth, making the increase a natural phenomenon. However, with the current system, the outer limits are rejected since they lie outside of the threshold. For the Outlier Rejection Filter, the average would be equal to the center pixel value, making this a legitimate disparity.

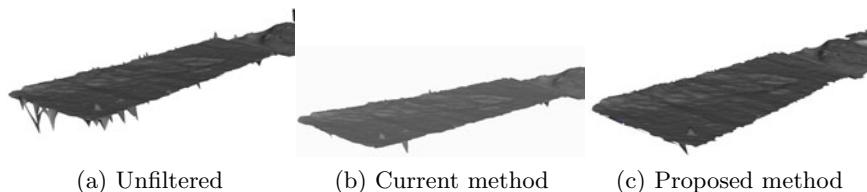


**Fig. 9.** Sample results from the Outlier Filter on a disparity map

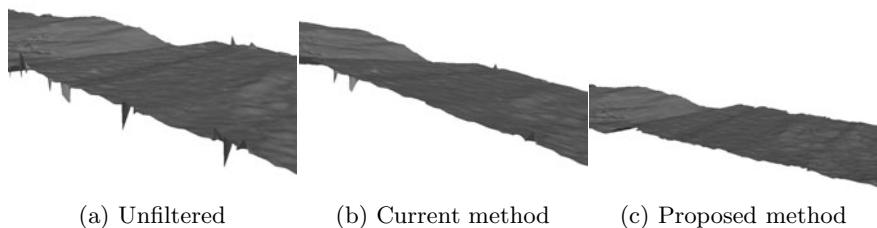
## 2 Results

Our results showed improved results as compared to the original and unfiltered systems.

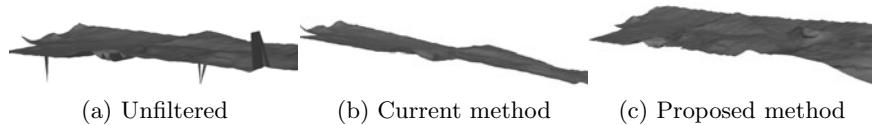
One element that is incredibly important is choosing proper thresholds as well as window sizes [7]. If the window size is too small, large outliers may survive as it may be gradual enough to exploit the example in Figure 8(b). Likewise,



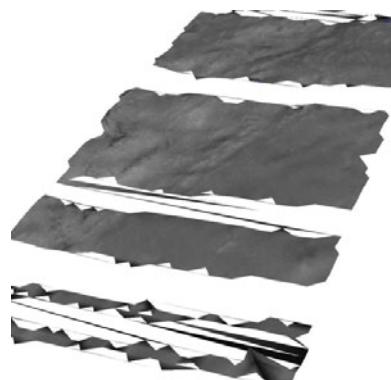
**Fig. 10.** Results from a trial run on similar segment using different filter. Notice that the new system performs equal, if not better than current system.



**Fig. 11.** Results from a trial run on similar segment using different filter. The new filter performs clearly better than current system.



**Fig. 12.** Results from a trial run on similar segment using the different filters. The new system performed equally to the current system.



**Fig. 13.** Example of an image with a threshold which is too selective

if the window size is too large, then the average will be minimally impacted by large outliers, thus some outliers will survive. The thresholds are similar, as very high thresholds make the filters very selective. Figure 13 shows an example of selecting a threshold for the Edge Density Filter which is too selective.

### 3 Conclusion

We have achieved a successful set of cascade filters for outlier detection and removal. The Edge Density Filter and the Outlier Removal Filter were able to increase the rejection rate of true outliers as well as decrease the rate of false positives. These filters are currently being implemented in the Ames Stereo Pipeline with direct application on the Apollo lunar reconstructions. This system may be implemented on other types of orbital imagery or other stereo vision applications. Examples include the Mars satellite orbiters and rovers.

### 4 Future Work

One aspect that would further increase run-time performance would be to run the filter on just a segment of each image. With orbital images, there may be only a small amount of each image which overlaps, making full computation unnecessary. Another interesting concept would be to create a method of automatically computing a threshold or window size. Currently, the best method for choosing an appropriate threshold or window size is by observation.

## References

1. Broxton, M.J., Nefian, A.V., Moratto, Z., Kim, T., Lundy, M., Segal, A.V.: 3d lunar terrain reconstruction from apollo images. In: ISVC 2009: Proceedings of the 5th International Symposium on Advances in Visual Computing, pp. 710–719. Springer, Heidelberg (2009)
2. Nefian, A.V., Husmann, K., Broxton, M., To, V., Lundy, M., Hancher, M.D.: A bayesian formulation for sub-pixel refinement in stereo orbital imagery. In: ICIP 2009: Proceedings of the 16th IEEE International Conference on Image Processing, Piscataway, NJ, USA, pp. 2337–2340. IEEE Press, Los Alamitos (2009)
3. Koch, R., Pollefeys, M., Gool, L.J.V.: Multi viewpoint stereo from uncalibrated video sequences. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 55–71. Springer, Heidelberg (1998)
4. Viola, P., Jones, M.J.: Robust real-time face detection. Int. J. Comput. Vision 57, 137–154 (2004)
5. Menard, C.: Robust stereo and adaptive matching in correlation scal-space. Technical report (1997)
6. Jain, R., Kasturi, R., Schuck, B.: Machine Vision. McGraw-Hill, New York (1995)
7. Gerrits, M., Bekaert, P.: Local stereo matching with segmentation-based outlier rejection. In: Canadian Conference Computer and Robot Vision, p. 66 (2006)

# Random Sampling Nonlinear Optimization for Camera Self-calibration with Modeling of Intrinsic Parameter Space

Houman Rastgar<sup>1</sup>, Eric Dubois<sup>1</sup>, and Liang Zhang<sup>2</sup>

<sup>1</sup> School of Information Technology and Engineering

University of Ottawa, Ottawa, Ontario, K1N 6N5 Canada

<sup>2</sup> Communications Research Center Canada, 3701 Carling Avenue

Ottawa, Ontario, K2H 8S2 Canada

**Abstract.** This paper presents a framework for random sampling nonlinear optimization for camera self-calibration with modeling of the camera intrinsic parameter space. The focal length is modeled using a Gaussian distribution derived from the results of the Kruppa equations, while the optical center is modeled based on the assumption that the optical center is close to the image center but deviates from it due to some manufacturing imprecision. This model enables us to narrow the search range of parameter space and therefore reduce the computation cost. In addition, a random sampling strategy is utilized in order to avoid local optima, where the samples are drawn according to this model. Experimental results are presented to show the effectiveness of the proposed nonlinear optimization algorithm, even in the under-constrained case involving only two frames.

## 1 Introduction

Camera self-calibration is the process of estimating the camera parameters using only information available through an image sequence [1]. It enables 3D reconstructions from a sequence of uncalibrated images without having to rely on a formal calibration process. There are generally three categories of self-calibration methods available in the literature. The first category uses the projective geometry of a scene and the Absolute Quadric to estimate the camera parameters [2]. An initial projective reconstruction has to be available before self-calibration can be achieved. The second category is based on the Kruppa equations that use an imaginary conic with complex points, namely the Absolute Conic [3]. Its performance relies on precise localization of the epipoles, which is not always possible since the locations of the epipoles are sensitive to noise. The third category uses algebraic properties of the essential matrix to provide camera parameter estimates [4–6]. However, the methods in this category have to deal with finding a global minima from a difficult objective function.

In order to facilitate the search for this global minima, Fusiello et. al. proposed an optimization algorithm based on Interval Analysis to guarantee the solution to be a global minima [6]. This method could take up to an hour to produce solutions for a set of cameras. Mendonca and Cipolla extend self-calibration to the case of multiple varying intrinsic parameters and larger image sequences [5]. They make no assumptions on the

parameter space, which makes their algorithm susceptible to failure when not enough frames are available. Whitehead and Roth propose uniformly sampling the parameter space and running a gradient descent in conjunction with a Genetic Algorithm iteratively [4]. They ignore the estimation of the optical center. Furthermore, since no prior distribution is assumed over the parameter space, this method suffers from redundant computations due to over-sampling and might not always give optimum results.

This paper proposes a random sampling nonlinear optimization algorithm for the minimization of an objective function derived from the algebraic properties of the essential matrix. The first contribution made in this paper is a model of the camera intrinsic parameters, namely the focal length and optical center. This is used to narrow the parameter search range for nonlinear optimization and to reduce computation time. The other contribution is the random sampling framework for nonlinear optimization, in which initial values of the intrinsic parameters are randomly selected according to a probabilistic model. This random sampling of initial parameter values enables the algorithm to avoid local optima.

Furthermore, we have observed that, although a wrong value for the optical center might not have a drastic impact on 3D visualization tasks [4], ignoring the optical center in the estimation by assuming that it is located in the middle of the image could lead to incorrect focal length estimates. Therefore, unlike the algorithms [3-5], which only attempt to estimate the focal length, we have included the estimation of camera optical center in the proposed optimization. Experiments in both two-frame and three-frame cases are performed to evaluate the proposed algorithm.

The paper is outlined as follows. Section 2 briefly reviews the process of camera self-calibration via the essential matrix. Section 3 describes the modeling of camera focal length and optical center as well as the random sampling framework of nonlinear optimization. Experimental results are shown in Section 4. Section 5 concludes the paper.

## 2 Self Calibration via the Essential Matrix

The task of camera self-calibration is to find the intrinsic parameters of a camera. These parameters, put in a matrix form, can be represented as:

$$K = \begin{bmatrix} f & s & c_x \\ 0 & \alpha f & c_y \\ 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where  $f$  denotes the focal length in pixels,  $s$  represents the skew,  $c_x$  and  $c_y$  are the optical center of the camera, and  $\alpha$  is the aspect ratio. Similar to most conventional self-calibration techniques, we assume that the skew parameter,  $s$ , is zero and that the aspect ratio,  $\alpha$ , is unity. Therefore, only focal length and optical image center need to be determined. It is known from [7] that given the intrinsic parameters  $K$ , one can obtain the essential matrix  $E$  from the fundamental matrix,  $F$ , i.e.,

$$E = K^T F K \quad (2)$$

An important property of the essential matrix that is utilized in our search for the camera parameters is that:

$$2EE^T E - \text{tr}(EE^T)E = 0 \quad (3)$$

must be satisfied so that  $E$  is a valid essential matrix [1]. The operation  $\text{tr}$  in equation (3) denotes the trace of a matrix. Therefore, given a fundamental matrix, equation (3) provides an objective function to find the camera intrinsic parameter by cost minimization. To this end, a nonlinear optimization algorithm is required.

### 3 Algorithm Description

The proposed method is to find an optimal  $K^*$  matrix so that the Frobenius norm of equation (3) is minimal, i.e.,

$$K^* = \min_K \|2EE^T E - \text{tr}(EE^T)E\|_F \quad (4)$$

where the subscript F stands for the Frobenius norm and the essential matrix,  $E$ , is derived from the fundamental matrix according to (2). Note that this objective function may have many local minima. Therefore, the search for a global minimum can be difficult and time consuming. To remedy this, we model camera intrinsic parameter space to narrow the search range and further use a random sampling technique to avoid the local minima.

#### 3.1 Modeling of Focal Length

We use a Gaussian distribution to model the search range of camera focal length. The parameters of this Gaussian distribution are determined from the solution of camera focal length using the Kruppa equations [3]. The Kruppa equations provide a rough approximation for the focal length. Its results can come close to the true solution but are not sufficiently accurate. From the results of the Kruppa equations, we determine the mean and variance of this Gaussian distribution as:

$$\mu_{\text{focal}} = \sum_{i=1}^N w_i f_{\text{Kruppa}}^i \quad (5)$$

$$\sigma_{\text{focal}}^2 = \sum_{i=1}^N w_i (f_{\text{Kruppa}}^i - \bar{f}_{\text{Kruppa}})^2 \quad (6)$$

where  $f_{\text{Kruppa}}^i$  is the focal length estimate as calculated from the  $i$ -th fundamental matrix in the sequence according to the Kruppa equations,  $\bar{f}_{\text{Kruppa}}$  is the mean of the focal length values and  $N$  represents the number of fundamental matrices in a sequence. The weights,  $w_i$  can be set according to the confidence in the estimation of the given fundamental matrix as in [5]. In our experiments, we set these weights equal to  $\frac{1}{N}$  since all the fundamental matrices were estimated from the ground truth data provided with the image sets. Moreover, in the two-frame case where only one fundamental matrix is available, the variance is set to 15% of the sum of image width and height.

### 3.2 Modeling of the Optical Center

The model for the optical center is chosen as a Gaussian distribution centered on the image center. This strategy was adopted by noting that the optical center is usually close to the image center but deviates from it due to some manufacturing imprecision that could be modeled as a Gaussian distribution. In the experiments we model the x and y components of the optical center as individual Gaussian distributions whose mean is the center of the image and whose standard deviation is set to be 10% of image width and height.

### 3.3 Nonlinear Optimization Algorithm

We have chosen the reflective Newton method, which is a constrained nonlinear optimization technique [8] to carry out the search for the solution of equation (4). This optimization technique attempts to minimize a smooth nonlinear function subject to constraints on the parameters as:

$$\min_{s \in \mathbb{R}^3} \|2EE^T E - \text{tr}(EE^T)E\|_F, \quad u > s > l \quad (7)$$

where  $l$  and  $u$  denote the upper and lower limits on the solution  $s$  and are defined in the next section.  $s$  is searched within the interior of the space defined by  $l$  and  $u$ . Since we have included the camera center as part of the optimization, the search for  $s$  is carried out in a three dimensional space where the first component of  $s$  is the focal length and the other two are the x and y locations of the optical center.

The nonlinear optimization will stop when the change of the costs (objective function) between two consecutive iterations of the reflective Newton method has reached its predefined threshold. In the experiments we have set this threshold to  $10^{-6}$  and the algorithm usually takes less than 10 iterations to reach this stopping criteria.

### 3.4 Complete Random Sampling Framework

In order to guarantee that we have not chosen a local minimum, we perform the nonlinear optimization combined with a guided sampling technique. In other words, the optimization is run multiple times where each instance of the optimization is initiated with a sampled point in the parameter space according to the distributions over these parameters as defined previously. The final result is chosen as the solution with the lowest residual.

Algorithm 1 details the summary of the proposed random sampling nonlinear optimization framework. The lines 1-3 show the distributions defined over each of the variables in the parameter space. Therefore, initial points for the focal length will be chosen from the distribution  $p(f)$  where  $\mu_{\text{focal}}$  and  $\sigma_{\text{focal}}$  are defined in (6), and initial points for the camera center are chosen from  $p(c_x)$  and  $p(c_y)$  where the parameter  $\tau$  is set to 10%. This is to say that the initial points for the camera center are chosen from a Gaussian having a standard deviation that is 10% of the image sides and whose mean is the image center, denoted by  $(I_x, I_y)$ .  $\text{Residual}(s)$  returns the error of a given solution,  $s$ . The lines 4-5 set the upper and lower constraints on the feasible solutions  $u$  and  $l$ . We have chosen these limits on the parameter space experimentally and set  $f\text{limit}$  to be

**Algorithm 1.** Random Sampling Least Squares Self-calibration

---

```

1:  $p(f) = N(\mu_{\text{focal}}, \sigma_{\text{focal}}^2)$ 
2:  $p(c_x) = N(I_x, \tau w)$ 
3:  $p(c_y) = N(I_y, \tau h)$ 
4:  $u = [\mu_{\text{focal}} + \text{flimit}, I_x + \text{xLimit}, I_y + \text{yLimit}]$ 
5:  $l = [\mu_{\text{focal}} - \text{flimit}, I_x - \text{xLimit}, I_y - \text{yLimit}]$ 
6: while  $\text{numSample} < \text{MaxIter}$  do
7:    $s_{\text{init}} = \text{RandomSelection}[f_0 \leftarrow p(f), x_0 \leftarrow p(c_x), y_0 \leftarrow p(c_y)]$ 
8:    $s_{\text{numSample}} = \text{ReflectiveNewton}(s_{\text{init}}, u, l)$ 
9:   if  $\text{Residual}(s_{\text{numSample}}) < \text{minResidual}$  then
10:     $s_{\text{Best}} = s_{\text{numSample}}$ 
11:     $\text{minResidual} = \text{Residual}(s_{\text{numSample}})$ 
12:   end if
13:    $\text{numSample} = \text{numSample} + 1$ 
14: end while
15: set  $s_{\text{optimal}} = s_{\text{Best}}$ 

```

---

15% of the sum of image width and height, and  $xLimit$  and  $yLimit$  to 15% of the image width and height respectively. These hard limits are imposed on the three dimensional parameter spaces in order to guarantee that the optimizer does not get lost in the parameter space. *RandomSelection* performs the random selection of the initial solution  $s_{\text{init}}$  according to the models of parameter space.

*ReflectiveNewton*( $s_{\text{init}}, u, l$ ) shows the optimization step subject to the given constraints. In addition, the maximum allowable number of random samplings, *MaxSamples*, was experimentally set to 150 in our implementation. After the maximum number of random samplings has been reached, the optimization stops and sets the kept solution,  $s_{\text{Best}}$  as the final optimal estimates  $s_{\text{optimal}}$ . Note that the limits  $u$  and  $l$  are adaptively changed depending on the confidence in the estimates of the focal length. If the Kruppa equations fail to produce results or have a variance above a threshold we can adjust the limits to allow for a broader range of values.

## 4 Experimental Results

The proposed method is tested on synthetic as well as real data sets against the reference method based on the Kruppa equations [3]. In addition, we have compared our algorithm with an identical method to the proposed technique, but without the guided sampling strategy, in order to demonstrate the effectiveness of our model of the parameter space. This method is simply the proposed algorithm where the parameter space is uniformly sampled rather than sampled based on our estimated distributions. This method is shown as the "Uniform Sampling" algorithm in the reported graphs. For comparison, we measure the accuracy of the estimated focal lengths by taking the absolute differences against the ground truth values.

For each synthetic test, data were generated by creating a set of projection matrices with arbitrary intrinsic and extrinsic parameters and then picking 200 points that are visible among all these cameras. These points are then projected in all the cameras using

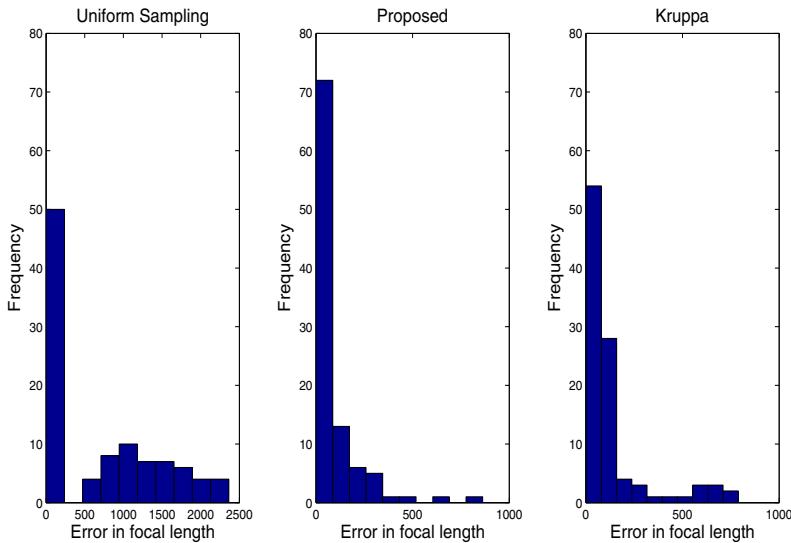
their projection matrices and the fundamental matrix between each pair is calculated using the 8-point algorithm [7]. This test was repeated 100 times with different camera arrangements.

We first evaluate the performance for the three-frame self-calibration application where the camera centers deviate from the image center according to a Gaussian distribution with a 90-pixel mean for images with a spatial resolution of  $512 \times 512$  pixels and the focal lengths are chosen randomly between 200 and 2000 pixels. Figure 1 shows the accuracy comparison for the estimation of the focal lengths between the proposed algorithm and the reference methods. As seen in these histograms, out of 100 different camera arrangements, the proposed algorithm found more than 70 solutions with less than 20 pixels error whereas the Kruppa method found only 53 and the Uniform Sampling strategy found only 50. In addition, the averaged sum of absolute errors over the 100 tests is 83 pixels for the proposed algorithm whereas it is 123 pixels for the Kruppa method and 680 for the Uniform Sampling method. The runtime of the proposed algorithm for each test is approximately 9s on a P4 dual core processor when the maximum allowable number of samples is reached. The Uniform Sampling method has an identical runtime to the proposed method and the Kruppa equations run in a negligible amount of time since they only require the solution to a quadratic equation.

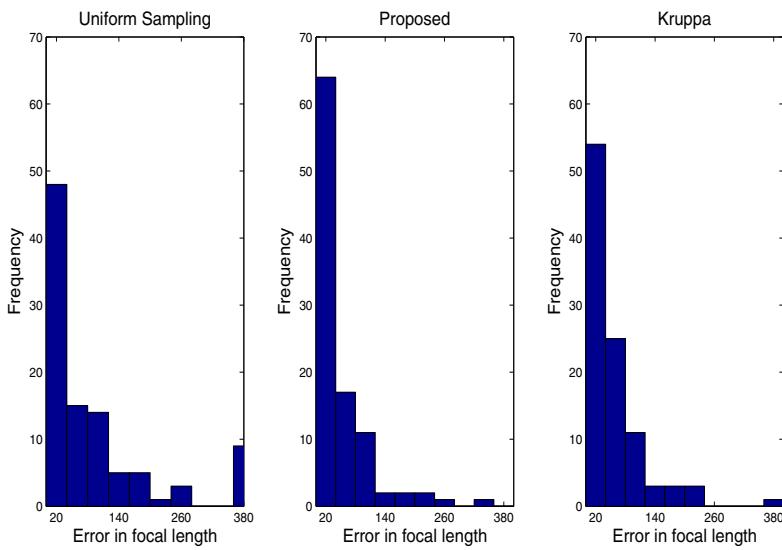
We then evaluate the performance for the two-frame self-calibration problem using the synthetic data with camera arrangements similar to the three-frame case. Figure 2 shows the histogram of the errors for the three algorithms in this set of test scenarios. In this set of experiments the proposed method has the highest number of solution with error below 20 pixels. The proposed method achieves 64 solutions with errors below the 20-pixel threshold out of a total of 100 experiments whereas the Uniform Sampling technique achieves 48 and the Kruppa method 54. In addition, the averaged value of errors over 100 tests was 45 pixels for the proposed method, while it was 55 pixels for Kruppa method and 109 for the Uniform Sampling method. Clearly the three-frame case has a higher gain in performance since the proposed algorithm is able to utilize the additional constraints.

We further evaluate the performance using real image sets, *Metron college*, *House*, *Corridor* and *Valbonne church* image sequences, from the Visual Geometry Group of Oxford University. The ground truth focal length was 1110 pixels for the *Metron college* sequence and 506 pixels for *Corridor* sequence, 626 pixels for the *House* sequence and 682 pixels for the *Valbonne church* data set as shown in Table 1. In these experiments the fundamental matrices are calculated using the 8-point algorithm [7] from the ground truth matches that are provided with the image sets. Since the *Valbonne church* does not have any ground truth matches we use SIFT [6] to estimate the feature matches across this sequence and then a RANSAC implementation of the 7-point algorithm to obtain the fundamental matrix [7].

The experiments confirmed that the proposed algorithm performs better than the reference methods. Table 1 shows the results for both two-frame and three-frame cases. *Metron college* images have almost zero skew and the camera center is very close to the image center and so the ideal assumption of the reference method are met and so the proposed method only slightly outperforms the Kruppa algorithm. In the cases of the *Corridor* and *Valbonne church* sequences the proposed algorithm significantly



**Fig. 1.** Histogram of sum of absolute errors for the reference methods versus the proposed method. Three frames are used in the nonlinear minimization.



**Fig. 2.** Histogram of sum of absolute errors for the reference methods versus the proposed method. Two frames are used in the nonlinear minimization.

reduces the error percentage. In the case of the *Corridor* sequence the high error percentage is caused by the fact that the fundamental matrices from which the self-calibration results are derived are nearly degenerate due to a dominant plane in the scene. The *House* sequence also incurs a high error due to its nearly degenerate camera geometry since the optical axis of the initial frames are parallel [3]. However, our method obtains an approximate solution where the Kruppa method fails to obtain any solutions at all. In addition, the results for the Uniform Sampling method on real image sets are generally very poor due to starting from arbitrary locations that are far from the actual solution.

**Table 1.** Error percentages for focal length computation

<i>Metron Sequence</i> , focal length = 1110 pixels		
	two frames	three frames
our method	1.22%	0.3351%
Kruppa Equations	1.95%	0.498%
Uniform Sampling	1.25%	78.23%
<i>House Sequence</i> , focal length = 626 pixels		
	two frames	three frames
our method	55.716%	55.716%
Kruppa Equations	NA	NA
Uniform Sampling	57.336%	99.97%
<i>Corridor Sequence</i> , focal length = 506 pixels		
	two frames	three frames
our method	51.548%	24.050%
Kruppa Equations	92.143%	54.671%
Uniform Sampling	79.23 %	80.234%
<i>Valbonne Sequence</i> , focal length = 682 pixels		
	two frames	three frames
our method	6.076%	2.21958%
Kruppa Equations	21.6664%	10.6646%
Uniform Sampling	68.123%	99.666%

In another set of experiments on real images the self-calibration algorithm is tested within a structure from motion framework. In these experiments we have used self-calibration to estimate the intrinsic parameters. These parameters are then used in conjunction with an image matching technique to obtain a 3D point cloud for the features that have been matches across the images in a sequence. We can then assess the quality of the intrinsic parameters obtained via self-calibration as the relative error in the 3D reconstruction using those parameters. We have excluded the Uniform Sampling method from these experiments since the results generally are not accurate enough for starting a structure from motion algorithm. The *Metron college* and the *Valbonne* data sets have been utilized for this set of experiments. During this experiment we use the SIFT algorithm to obtain matches between the frames. Following this, the fundamental matrices are found using a RANSAC based algorithm and the result is used as input for the self-calibration algorithms. After obtaining the intrinsic parameters using self-calibration

we find the extrinsic parameters by post and pre-multiplying the fundamental matrix with the intrinsic matrices according to [2]. Once the essential matrices are calculated, two frames are chosen and the projection matrices for these two frames are obtained from the essential matrix between these two frames [7]. After these projection matrices are found, the feature matches between these two frames are triangulated to find a 3D point cloud corresponding to the matches. After this set of 3D points are reconstructed using the first two frames, additional camera matrices are found using a traditional calibration technique as long as they share some features with the two frames used in the initial reconstruction. At the last stage, Bundle Adjustment [10] is used to refine the 3D point cloud and the camera parameters. The SBA library [11] was used to carry out the Bundle Adjustment refinement. It must be noted that the final reconstruction depends to a high degree on the initial camera parameters used in the first two frames.

Table 2 shows the respective reconstruction errors and errors in the focal length estimates after Bundle Adjustment for the two sequences with respect to the proposed self-calibration algorithm and the reference method.

**Table 2.** Reconstruction error and error percentages for focal length computation after Bundle Adjustment

<i>Metron Sequence</i> , focal length = 1110 pixels		
	reconstruction error	error in focal length
our method	0.135690%	7.0397%
Kruppa Equations	0.136602%	14.2599%
<i>Valbonne Sequence</i> , focal length = 682 pixels		
	reconstruction error	error in focal length
our method	0.149717 %	10.0523 %
Kruppa Equations	0.671684%	22.13 %

Table 2 shows that even with a nonlinear refinement our algorithm performs better since we start the nonlinear minimization of structure and motion from a more suitable starting location. Note that for both cases the error in the focal length increases after Bundle Adjustment. This occurs since Bundle Adjustment attempts to find a set of parameters to better fit the reconstruction and so even if a few outliers are present in the image matches this could lead to the algorithm diverging from the actual camera intrinsic parameters in order to better fit the data. However it can be seen that in both cases the reconstruction error is quite small.

## 5 Conclusion

The paper concludes that the proposed random sampling nonlinear optimization with modeling of the intrinsic parameter space can indeed improve the performance of camera self-calibration. The modeling of camera intrinsic parameter space enables us to narrow the search range for the optimization and therefore reduce the computation burden; while the random sampling strategy, according to the parameter model, effectively prevents the algorithm from getting stuck in local minima. These claims were confirmed by experimental results with the synthetic and real data.

## References

1. Huang, T., Faugeras, O.: Some properties of the E matrix in two-view motion estimation. *IEEE Transactions on PAMI* 11, 1310–1312 (1989)
2. Pollefeys, M., Koch, R., Gool, L.: Self-calibration and metric reconstruction inspite of varying and unknown intrinsic camera parameters. *IJCV* 32, 7–25 (1999)
3. Sturm, P., Cheng, Z., Chen, P.C.Y., Poo, A.N.: Focal length calibration from two views: Method and analysis of singular cases. *CVIU* 99, 58–95 (2005)
4. Whitehead, A., Roth, G.: Estimating intrinsic camera parameters from the fundamental matrix using an evolutionary approach. *EURASIP Journal on Applied Signal Processing*, 1113–1124 (2004)
5. Mendonca, P., Cipolla, R.: A simple technique for self-calibration. In: *CVPR*, vol. 1, pp. 500–505 (1999)
6. Fusiello, A., Benedetti, A., Farenzena, M., Busti, A.: Globally convergent autocalibration using interval analysis. *IEEE Transactions on PAMI*, 1633–1638 (2004)
7. Hartley, R., Zisserman, A.: *Multiple View Geometry in Computer Vision*. Cambridge Univ. Pr., Cambridge (2003)
8. Coleman, T., Li, Y.: On the convergence of reflective Newton methods for large-scale non-linear minimization subject to bounds. *Mathematical Programming* 67, 189–224 (1994)
9. Lowe, D.: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60, 91–110 (2004)
10. Triggs, B., McLauchlan, P., Hartley, R., Fitzgibbon, A.: Bundle adjustment: modern synthesis. *Vision Algorithms: Theory and Practice*, 153–177 (2000)
11. Lourakis, M.A., Argyros, A.: SBA: A Software Package for Generic Sparse Bundle Adjustment. *ACM Trans. Math. Software* 36, 1–30 (2009)

# Facial Fraud Discrimination Using Detection and Classification

Inho Choi and Daijin Kim

Department of Computer Science and Engineering  
Pohang University of Science and Technology (POSTECH)  
`{ihchoi,dkim}@postech.ac.kr`

**Abstract.** This paper proposes facial fraud discrimination using facial feature detection and classification based on the AdaBoost and a neural network. The proposed method detects the face, the two eyes, and the mouth by the AdaBoost detector. To classify detection results as either normal or abnormal eyes and mouths, we use a neural network. Using these results, we calculate the fraction of face images that contain normal eyes and mouths. These fractions are used for facial fraud detection by setting a threshold based on the cumulative density function of the Binomial distribution. The FRR and FAR of eye discrimination of our algorithm are 0.0486 and 0.0152, respectively. The FRR and FAR of mouth discrimination of our algorithm are 0.0702 and 0.0299, respectively.

## 1 Introduction

The increasing number of automated financial machines such as Automated Teller Machines (ATM) and Cash Dispensers (CD) has increased not only the convenience of performing financial transactions but also the frequency of illegal financial transactions. These illegal transactions consist of attempts to use stolen cash cards or credit cards. Because biometric information cannot be stolen, the use of biometric information can prevent illegal financial transactions.

According to “*Financial success for biometrics?*” [1], many approaches can be used to verify a customer’s identity at the ATM, including fingerprint, iris, vein, and signature verification, and keystroke dynamics. These methods not only act as substitutes for identification cards such as cash cards or credit cards but also restrict illegal financial transactions with lost or stolen cards. However, many people refuse to use these methods because they are unwilling to provide biometric information.

For this reason, we study methods for preventing illegal financial transactions. In general, CCTV captures face images of ATM users. If an illegal financial transaction occurs, captured images are used in order to apprehend the suspect. But captured images are not reliable if the customer wears a mask or sun-glasses. If we know the person is wearing a mask or sun-glasses (Fig. ⑩), we can restrict a his or her financial transaction. Dong and Soh presented an approach to image-based fraud detection [2] based on moving object detection, skin color and a face template. This approach is not simple or very heuristic for face,



**Fig. 1.** Some example of facial fraud

eyes and mouth. Lin and Liu proposed vision-based system [3] using tracking people and detecting face occlusion. Tracking people is based on moving object detection and face occlusion is based on the ratio of skin color region under face area. Moving object detection is insufficient for face detection if user hold still. Also, skin color region is very heuristic. So, we propose a simple and a intuitive algorithm based on detection and classification methods using machine learning and pattern recognition approaches.

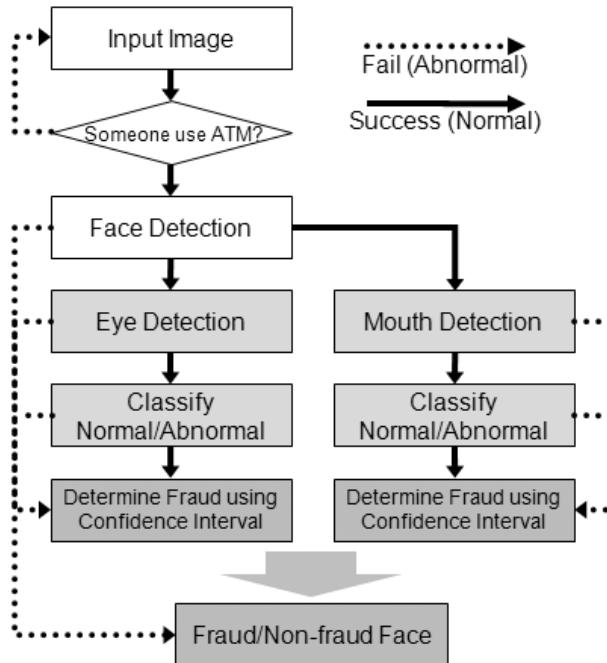
In this paper, we use AdaBoost with the modified census transform (MCT) [4,5] due to its simplicity of learning and high speed of detection. Fig. 2 shows an overview of the facial fraud detection method. First, it detects the face region using the MCT-based AdaBoost face detector. Second, it divides the face into subregions and detects the eyes and mouth using the MCT-based AdaBoost eye detector and mouth detector in input images. Third, a neural network classifies the eyes and mouth in each image as either normal or abnormal. Finally, the method decides whether fraud has occurred using the confidence interval.

If the facial parts are not detected at all by facial components detection, fraud is declared. Fraud is also declared if the parts are detected, but are classified as non-eye or non-mouth by neural network-based classifier. If video data is available, the overall decision is based on the fraction of correctly detected regions.

## 2 Facial Feature Detection

The Modified Census Transform (MCT) is a non-parametric local transform which modifies the census transform by Fröba and Ernst [5]. It is an ordered set of comparisons of pixel intensities in a local neighborhood representing which pixels have lesser intensity than the mean of pixel intensities.

We present the detection method for the face, the eyes and the mouth by the AdaBoost training with MCT-based features [6,7] (Fig. 3) using positive and negative samples (Fig. 4).



**Fig. 2.** An overview of the proposed method

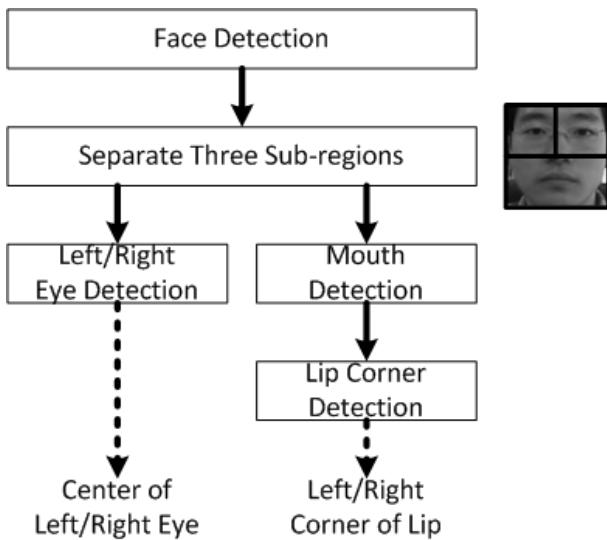
In the detection phase, if facial feature detection fails, our algorithm determines that the input image is a fraud face. But if our detection algorithm detects facial features, a classification process is needed. This process will be explained in the next section “Normal/Abnormal Face Classification using Neural Network”.

### 3 Facial Fraud Detection

To determine whether eyes and mouths are normal or abnormal, we use a neural network classifier. To improve accuracy, we use an image sequence because doing so complements the detection and classification method. If the number of the normal eyes and mouths is greater than the specified threshold, then the input face is a non-fraud face. Otherwise, we determine that the input face is a fraud face. The threshold value is computed by the probability mass function and cumulative density function of Binomial distribution. To simplify the calculation, we suppose that face detection has not failed with the input images. If face detection fails, we assume that the input face is a fraud face.

#### 3.1 Normal/Abnormal Face Classification Using Neural Network

To classify detected eyes and mouths as either normal or abnormal, we use the deep belief network by Hinton [8]. This neural network has a fast running

**Fig. 3.** An overview of the facial feature detection

(a) Eye and non-eye data (15x15).



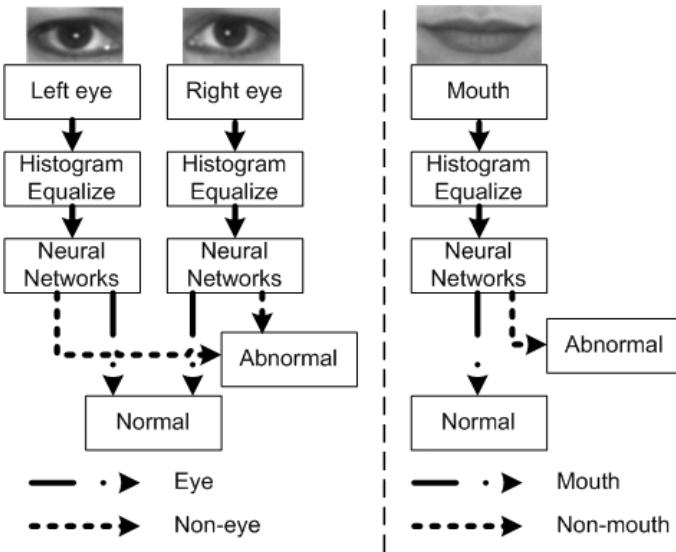
(b) Mouth and non-mouth data (15x9).



(c) Lip corner and non-corner data (11x11).

**Fig. 4.** Training data for facial feature detection

speed and good classification performance. To train normal/abnormal eye classification, we use 4,000 eye image and 4,000 non-eye images. To train normal/abnormal mouth classification, we use 4,500 mouth images and 4,500 non-mouth images. Histogram equalization is used to reduce illumination variations in the training images.



**Fig. 5.** Normal/abnormal classification using a neural network

Fig. 5 shows a flowchart of the normal/abnormal classification process. To normalize, we use the center of the eyes and corners of the lips.

Fig. 6 shows some examples of the normal/abnormal classification training data. Non-eye and non-mouth data include sun-glass images and mask images.

### 3.2 Fraud Detection Using Confidence Interval

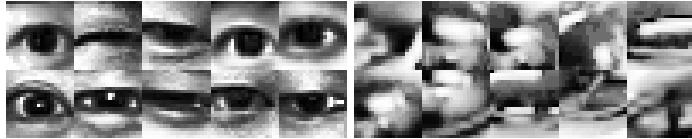
To determine whether or not eyes are fraudulent, we use the fraction of the number of normal eyes divided by the total number of face images. In other words, the fraud eye detection method uses the image sequence and discriminates fraud eyes in the image sequence when a customer uses an ATM. The fraud eye detector is defined as

$$D_E(\mathbf{I}) = \begin{cases} \text{non-fraud eye if } \frac{ne(I)}{nf(I)} \geq \theta_E, \\ \text{fraud eye otherwise,} \end{cases} \quad (1)$$

where  $I$  is the image sequence,  $ne(I)$  is the number of normal eyes in  $I$ ,  $nf(I)$  is the number detected face images in  $I$  and  $\theta_E$  is a threshold of the  $\frac{ne(I)}{nf(I)}$ .  $\theta_E$  is defined as

$$\theta_E = \frac{x}{n}, \quad (2)$$

where  $n$  is the number of images which have of a face and  $x$  is the expected number of images which have at least one correct eye. The expected number  $x$  is calculated from the Binomial distribution. To calculate  $x$ , we use the probability of the false acceptances calculated by Binomial random variable given the false acceptance rate (FAR).



(a) Eye and non-eye data (20x20).



(b) Mouth and non-mouth data (40x20).

**Fig. 6.** Training data for normal/abnormal classification to facial components

The Binomial probability mass function (PMF) is defined as

$$f(x; n, p) = P\{\mathbf{y} = x\} = \binom{n}{x} p^x (1-p)^{n-x}, \quad (3)$$

where  $x$  denotes the number of times that a certain event occurs in  $n$  trials and  $p$  is the probability that the event occurred. The fraud eye detection method uses the cumulative density function (CDF) because it is defined with the probability that the event will occur from the specified interval. The CDF is defined as

$$F(x; n, p) = P\{\mathbf{y} \leq x\} = \sum_{i=1}^x f(i; n, p) \geq c. \quad (4)$$

where  $c$  is the confidence value. The probability is less than  $1 - c$  in  $n$  images, where the probability that the number of false detected images is greater than  $x$ .

The fraud eye detection method uses the CDF of the FAR in the eye detection and classification method, and the CDF is defined as

$$F(x; n, p) = \sum_{i=1}^x \binom{n}{i} p^i (1-p)^{n-i} \geq c, \quad (5)$$

where  $p$  is the FAR of the eye detection and classification and  $c$  is the desired confidence of fraud eye detection. We can calculate  $x$  given  $n$  iteratively. Finally, the fraud eye detection method computes  $\theta_E$  using  $x$ .

For examples, suppose that the FAR of the eye detection and classification method is 0.03, and the total number of input images is 50. In that case,  $x$  is 6 by Eq. 6 where the desired confidence is 99.9%.

$$\sum_{i=1}^x \binom{50}{i} (0.03)^i (0.97)^{50-i} \geq 0.999, \quad (6)$$

From these values,  $\theta_E$  is calculated ( $\theta_E = \frac{6}{50} = 0.12$ ). That is to say, the input images are deemed to consist of non-fraud eyes when the number of the detected normal eyes is greater than or equal to 6 in 50 input images.

We applied fraud eye detection to fraud mouth detection. The fraud mouth detector is defined as

$$D_M(I) = \begin{cases} \text{non-fraud mouth if } \frac{nm(I)}{nf(I)} \geq \theta_M, \\ \text{fraud mouth otherwise,} \end{cases} \quad (7)$$

where  $I$  is the image sequence,  $nm(I)$  is the number of normal mouths in  $I$ ,  $nf(I)$  is the number of detected face images in  $I$  and  $\theta_M$  is a threshold of the  $\frac{nm(I)}{nf(I)}$ .  $\theta_M$  is defined as

$$\theta_M = \frac{x}{n}, \quad (8)$$

where  $n$  is the number of the input images and  $x$  is the expected number of images which have at least one correct result. Because the calculation process is equivalent to the fraud eye detection, we omit the detailed description.

## 4 Experimental Result

For face detection training, we use 50,000 face images on the Internet. For training the eye detector, we use the Asian face image database PF01 [9] and POSTECH face database (PF07) [10] and eye images from the Internet, including 29,000 eye images and 300,000 non-eye images whose size is  $15 \times 15$ . For mouth detection, we use PF01 and PF07, 14,000 prepared mouth images and 31,000 non-mouth images whose size is  $15 \times 9$ . The average processing time is 10ms using a Core2Duo 3.2GHz system.

### 4.1 Facial Fraud Detection

To evaluate facial fraud detection, we make a subset of the AR face database [11] called AR-FDD that has the purpose of testing facial fraud detection. The AR-FDD face database contains 1086 images ( $96 \text{ people} \times 2 \text{ different conditions} \times 3 \text{ different illuminations} \times 1 \text{ or } 2 \text{ sessions}$ , some people have first session). It has 2 different conditions, which are wearing sun-glass or a mask, and it consists of 3 different illuminations, including normal illumination, right illumination, and left illumination.

Fig. 7-(a) shows some example results of the detection and classification algorithm in the AR-FDD face database.

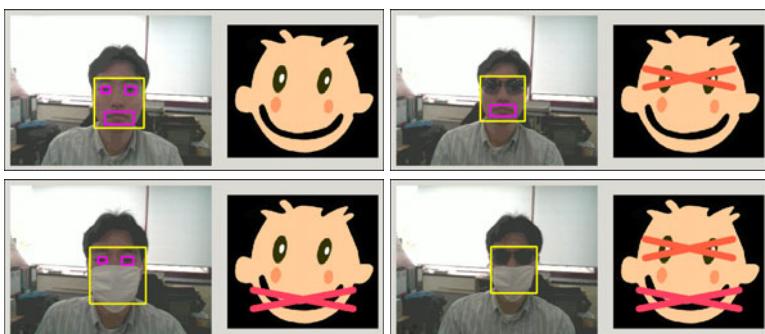
Table 1 shows the performance of the facial feature detection and fraud detection algorithm in the AR-FDD face database. To compute accuracy, we use only correctly detected face images. The false acceptance of eye detection and classification is calculated in the wearing mask group, because the wearing sun-glasses group does not have detectable eyes. With sameness, the false acceptance of the mouth detection and classification is calculated in the wearing sun-glasses group. The classification algorithm is helpful to reduce the false acceptance rate in eye and mouth fraud detection.

**Table 1.** The FRR and FAR of the facial fraud detection algorithm using facial feature detector and classifier in the AR-FDD face database

Category	Using detector (%)	Using detector and classifier(%)
FRR of Eye	0.93( $\frac{5}{535}$ )	4.86( $\frac{26}{535}$ )
FAR of Eye	82.68( $\frac{411}{527}$ )	1.52( $\frac{37}{527}$ )
FRR of Mouth	0.02( $\frac{1}{527}$ )	7.02( $\frac{37}{527}$ )
FAR of Mouth	70.09( $\frac{375}{535}$ )	2.99( $\frac{16}{535}$ )



(a) Detection and classification result.



(b) Some results in the real situation.

**Fig. 7.** Some results of facial fraud detection

## 4.2 Calculate Threshold of Fraud Detection

In the AR-FDD face database, if the number of input face images is 50 and the confidence is 99.9%, we calculate  $\theta_E$  in the fraud eye detection. First,  $x$  is calculated as

$$F(x; 50, 0.0152) \geq 0.999, \quad (9)$$

where  $x$  is 4 and  $\theta_E = \frac{x}{n} = \frac{4}{50} = 0.08$ .

Similarly,  $\theta_M$  is computed in the fraud mouth detection. If the confidence is 99.9%, by Eq. 9,  $x$  is calculated as

$$F(x; 50, 0.0299) \geq 0.999, \quad (10)$$

where  $x$  is 6 and  $\theta_M = \frac{x}{n} = \frac{6}{50} = 0.12$ . In this case, two thresholds of fraud detection for fraud eye and fraud mouth are 0.08 and 0.12 to get 99.9% accuracy.

In the real situation, the performance of the facial fraud detection are 91.3% (one image) and 98.5%(image sequence) (Fig. 7(b)).

## 5 Conclusion

In this paper, we presented a facial fraud detection algorithm that helps automated teller machines take reliable images. The algorithm was based on detection and classification algorithms using AdaBoost with MCT-based facial features and a neural network. It determined fraud eyes and mouths with the confidence interval of Binomial distribution for the facial feature detector and fraud classifier. The proposed algorithm is helpful to reduce illegal financial transaction on ATMs, and it helpful to increase the reliability of face recognition system and their applications. To improve performance, we will consider a liveness problem and nonuniform illuminations.

## Acknowledgements

This work was partially supported by the Intelligent Robotics Development Program, one of the 21st Century Frontier R&D Programs funded by the Ministry of Knowledge Economy (MKE). Also, this work was partially supported by the MKE (The Ministry of Knowledge Economy), Korea, under the Core Technology Development for Breakthrough of Robot Vision Research support program supervised by the NIPA (National IT Industry Promotion Agency) (NIPA-2010-C7000-1001-0006).

## References

1. Financial success for biometrics? Biometric Technology Today (2005)
2. Dong, W., Soh, Y.: Image-based fraud detection in automatic teller machine. International Journal of Computer Science and Network Security 6, 13–18 (2006)

3. Lin, D., Liu, M.: Face occlusion detection for automated teller machine surveillance. In: Chang, L.-W., Lie, W.-N. (eds.) PSIVT 2006. LNCS, vol. 4319, pp. 641–651. Springer, Heidelberg (2006)
4. Freund, Y., Schapire, R.: A short introduction to boosting. *Journal of Japanese Society for Artificial Intelligence* 14, 771–780 (1999)
5. Fröba, B., Ernst, A.: Face detection with the modified census transform. In: Proceedings of IEEE International Conference on Automatic Face and Gesture Recognition, pp. 91–96 (2004)
6. Jun, B., Kim, D.: Robust real-time face detection using face certainty map. In: Proceedings of 2nd International Conference on Biometrics, pp. 29–38 (2007)
7. Choi, I., Kim, D.: Eye correction using correlation information. In: Yagi, Y., Kang, S.B., Kweon, I.S., Zha, H. (eds.) ACCV 2007, Part I. LNCS, vol. 4843, pp. 698–707. Springer, Heidelberg (2007)
8. Hinton, G., Salakhutdinov, R.: Reducing the dimensionality of data with neural networks. *Science* 313, 504–507 (2006)
9. Je, H., Kim, S., Jun, B., Kim, D., Kim, H., Sung, J., Bang, S.: Asian Face Image Database PF01. Database, Intelligent Multimedia Lab, Dept. of CSE, POSTECH (2001)
10. Lee, H., Park, S., Kang, B., Shin, J., Lee, J., Je, H., Jun, B., Kim, D.: The postech face database (pf07) and performance evaluation. In: Proceeding of 8th IEEE International Conference on Automatic Face and Gesture Recognition, pp. 1–6 (2009)
11. Martinez, A., Benavente, R.: The AR Face Database. CVC Technical Report #24 (1998)

# Segmentation of Abdominal Organs Incorporating Prior Knowledge in Small Animal CT

SooMin Song<sup>1</sup> and Myoung-Hee Kim<sup>1,2,\*</sup>

<sup>1</sup> Department of Computer Science and Engineering  
Ewha Womans University, Seoul, Korea  
[smsong@ewhain.net](mailto:smsong@ewhain.net)

<sup>2</sup> Center for Computer Graphics and Virtual Reality  
Ewha Womans University, Seoul, Korea  
[mhkim@ewha.ac.kr](mailto:mhkim@ewha.ac.kr)

**Abstract.** For quantification of drug's delivery using small animals measuring biochemical changes in abdominal organs based on functional images is essential. However, in those images, the object boundaries are not clearly enough to locate its shape and position. And even though the structural information is compensated using image registration technique, delineation of organs is difficult and time-consuming. So we suggested an automatic procedure for delineation of organs in mouse PET image with the aid of atlas as a priori anatomical information. Prior information was given by voxel label number. CT used to construct an atlas is transformed to match mouse CT to be segmented. For each label corresponding voxels represent the same organ. Then, mouse CT-PET pairs should be aligned to identify organ area in PET. After all images are aligned and fused each other both structural and functional information can be observed simultaneously for several organs.

## 1 Introduction

For efficacy test of newly developing drug using small experimental animals, measuring absorption of radiopharmaceuticals and tracking its temporal change at each organ is essential. Recently, the advent of dedicated small animal imaging equipments such as microPET, microCT, and microMR enables in-vivo imaging and long term follow up of biochemical changes without animals' sacrifice. However, even in same animal, the speed and sensitivity of injected drug differs on each abdominal organ. Therefore, for accurate quantifying metabolic activity at each organ, delineating organ's ROI is required.

For measuring radioactivity, functional imaging such as PET has been a major tool. However, due to the poor resolution and blurry artifacts of microPET, it is difficult to recognize the exact organ's shape and position in images. Furthermore, PET image quality is influenced by animal's condition. For example, if animal are fed before imaging, area of the heart or the bladder has high intensities and is more blurry, thus neighbouring organs cannot be shown clearly because of low image contrast.

---

\* Corresponding author.

Therefore, aligning structural images such as CT or MR providing anatomical information of internal structures on to functional images is helpful to locate organs' boundaries. However, even in structural images automatic identification of abdominal organs has been a high challenging task due to partial volume effects, grey-level similarities of adjacent organs, contrast media affect, and the relatively high variation of organ position and shape. In small animal images these artifacts makes it more difficult due to small size of animal organs ranging from few cubic millimeters to several centimeters. Moreover, delineating several organs simultaneously is time-consuming task. It is more serious when image data are large for example, when time serial data are required for measuring kinetics of radiopharmaceuticals.

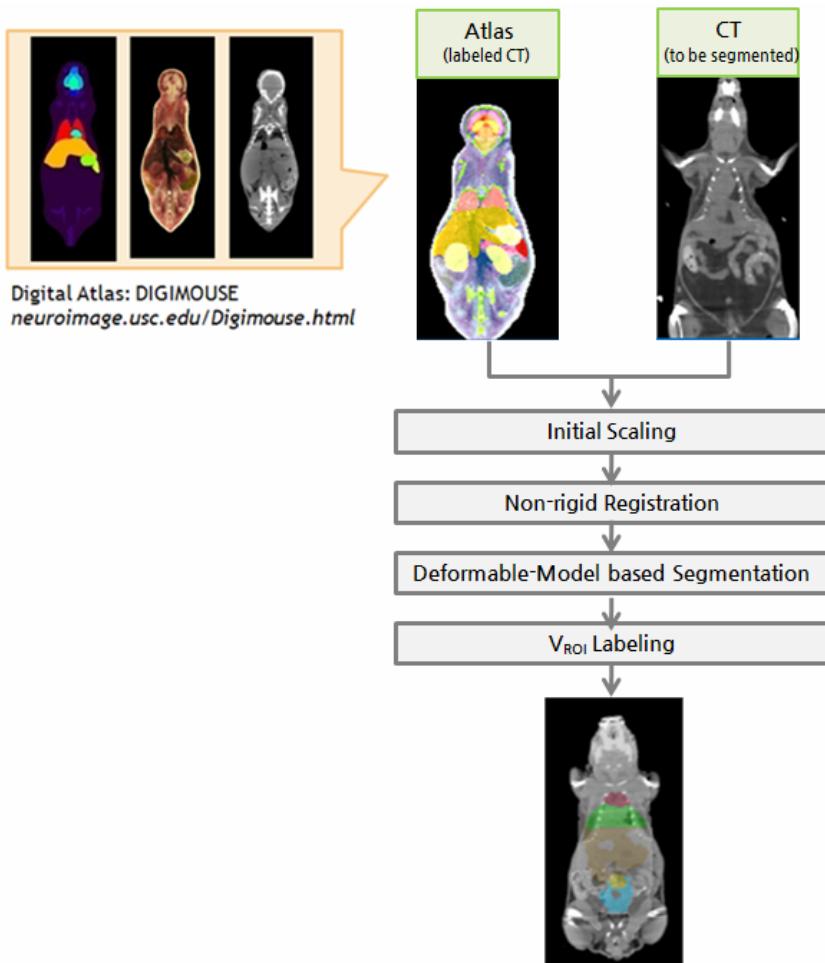
One promising approach to perform automatic and fast segmentation is to incorporate prior knowledge. Prior knowledge in segmentation process provides shape, position or texture information as a model and usually provided as image volume which is already segmented and each voxel is labeled with organ's ID. These prior models are also called atlas or template. In model-based segmentation, extraction of each organ's ROI is achieved by computing an anatomically correct coordinate transformation between atlas and the image to be segmented. This atlas-based method is becoming a standard paradigm in biomedical image segmentation [2].

Previous works to combine anatomical and functional information in small animal images, several algorithms were proposed. Jan [4] proposed a way of combining PET, CT, and SPECT images using a calibration phantom and an animal holder, and Chow [5] aligned coordinate systems between images by moving the imaging cylinder. Rowland [6] and Shen [7] developed PET-MRI, MR-SPECT image registration algorithms respectively. But they were all focused to merge two image modalities and overlay them. So the exact organ shape couldn't be recognized easily. And segmentation method for multiple abdominal organs is not suggested yet nevertheless of their necessity. On the other side, some model-based algorithms are applied to rat brain images [x, x].

In this paper, we suggest a method to extract the boundary of abdominal organs such as liver, heart, and bladder, etc in microCT. We exploit prior anatomical knowledge for high accuracy and speedup of segmentation procedure. Our proposed method is composed of two procedures. Firstly, we perform a registration process to align prior model to image. Then, the coordinate of two image volumes are matched, we extract each organ's contour at segmentation step. In the rest of this paper, we describe registration and segmentation process in Section 2 and Section 3, respectively. And experimental results and conclusion follow in Section 4 and 5.

## 2 Spatial Correspondence between Anatomical Model and Image

As mentioned in previous chapter, to incorporate prior anatomical information registration between the atlas and image to be processed should be aligned. Then, segmentation is done via registration. The whole procedure is described in Fig 1 below. In this paper, we use two sets of data, one is an atlas image and the other is image to be segmented. The atlas [8] was generated using coregistered CT and cryosection images of normal mouse. Using these images the segmentations of the 20 organs were done by the help of a professional mouse anatomist. After defining organ regions every voxel was labeled.



**Fig. 1.** Flowchart of whole procedure: segmentation via registration

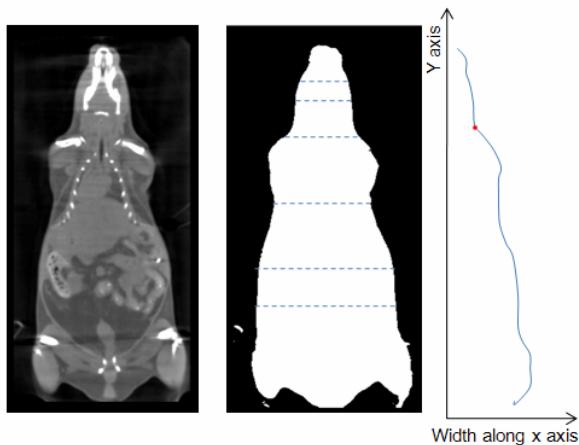
## 2.1 Initial Scale Transformation

Every living object varies in size and shape according to gender, age as well as illness condition. Moreover, experimental animals such as mouse and rats grow faster than human. Therefore, their age represented by weeks after birth differs on each object. Thus, a transformation to fit approximate object size should be performed. When the sizes of two objects are considerably different alignment error can be increased at subsequent registration process.

In this paper, we're only interested in identifying abdominal organs, we firstly restrict the volume of interest to abdomen area. To detect abdomen area, we convert the original image binary with simple thresholding algorithm. Then, we calculate the line width of segmented volume along x axis. Every width value is projected onto y axis

we get profile curve. And we define the separation point between head and abdomen area as a first minimum along this curve (Fig 2).

Now we only concern the abdomen region in the following processes. In cropped abdomen volume, we perform scale transform between atlas and CT images. We represent the object size using the length of principal axes and for scaling, we compute eigen vectors and carry linear transform along these vector directions.



**Fig. 2.** Profile curve to detect abdomen area

## 2.2 Non-rigid Image Registration between Atlas and CT

Image registration is the process of aligning images to make their features correspond. It yields a transformation that relates the coordinate systems of the two sets of images. In this paper labeled CT is transformed into mouse CT so that internal organs' shape and position in mouse CT can be easily recognized by overlapping voxel's label. This intra-modality, inter-subject image matching is done by deformable registration using voxel similarity.

In voxel-based registration the transformation factor can be computed by optimizing some measure directly calculated from the voxel values in the images. We used normalized mutual information (NMI) as a voxel similarity measure. It measures statistical dependency between intensity values of two images [10].

Mutual information is a voxel similarity measure that is expressed as the difference between the sum of the entropies of the individual images,  $H(A)$  and  $H(B)$ , and the joint entropy of the combined image where they overlap,  $H(A,B)$ . The entropy of a single image can be computed from the probability distribution of its intensities, and the joint entropy of two images can be estimated by computing a joint histogram of paired intensities. By interpreting the entropy as measure of uncertainty, mutual information is a measure of which one image explains the other. That means maximizing mutual information results in maximizing the information the image contains about the other. And therefore two images are well aligned when mutual information of geometrically corresponding gray values is maximal.

$$MI(A, B) = H(A) + H(B) - H(A, B) \quad (1)$$

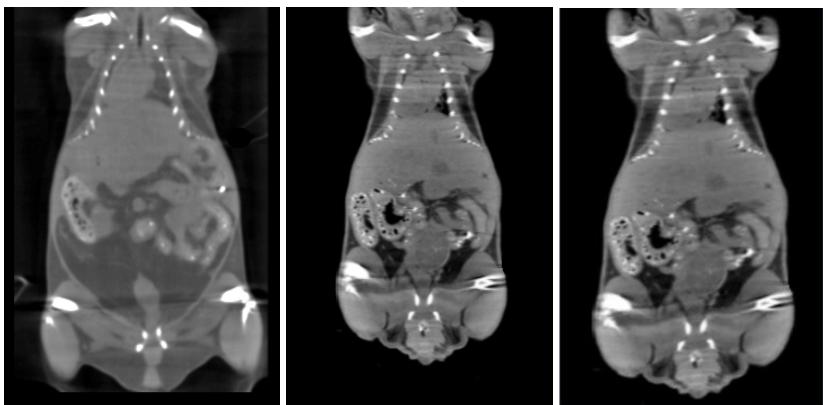
And normalized mutual information proposed by Studholme [11] has proved very robust. It is less insensitive to the overlapping regions of the two images.

$$NMI(A, B) = \frac{H(A) + H(B)}{H(A, B)}. \quad (2)$$

Two images A and B can be registered by searching for an optimal transformation T that maximizes the mutual information between the original image A(x) and the transformed image B(T(x)), where x is a voxel in image A, and the closest voxel in image B is T(x), so that

$$\hat{T} = \arg \max_T I(A(x), B(T(x))). \quad (3)$$

Like mentioned earlier two images are acquired from different mouse so we applied B-spline based deformable transformation to find T that relates the spatial information.



**Fig. 3.** Corresponding slice after 3D registration: a) source, b) target

### 3 Delineation of Multiple Organs in CT

The more accurately the transformation maps the atlas onto the image to be segmented, the more accurate the result of the organ identification. When atlas image is deformed during registration process, underlying contours in labeled CT is overlaid on segmented CT image. However, registration cannot provide perfect matching of internal organs because of individual difference such as subcutaneous fat, or gender, etc. Therefore, in this chapter, we explain fine tuning method for local modification of contours using curve evolution algorithm.

#### 3.1 Geodesic Active Contour Based Segmentation

Due to the noisy nature of CT, streak artifact is more serious when imaging object is small. Therefore, we applied noise reduction filter, an anisotropic diffusion filter to

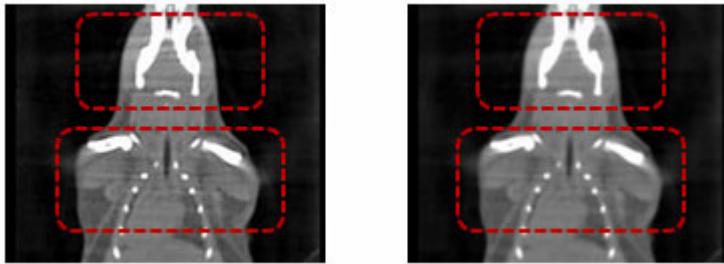
enhance edge area whose gradient is large and to smooth where gradient value is small. The anisotropic diffusion equation

$$I_t = \operatorname{div}(c(x, y, t) \nabla I) \quad (4)$$

has been proposed [5] as a smoothing filter provided where the conduction coefficient  $c$  is a local function of the magnitude of the gradient of the brightness function, i.e.,

$$c(x, y, t) = g(\|\nabla I(x, y, t)\|) \quad (5)$$

where  $g(\cdot)$  is a nonnegative monotonically decreasing function with  $g(0) = 1$ .



**Fig. 4.** Streak artifact in microCT: (a) original image, (b) noise reduced result

When image contrast has enhanced, we compute contours using geodesic active contour model, one of the most efficient algorithm in concave object segmentation. The geodesic active contour model was proposed by Caselles in 1997[4]. Conventional active contour model, introduced by Kass[1], is basically composed of two components, one controls the smoothness of the curve and another attracts the curve towards the boundary. Geodesic active contour additionally introduce an edge detector, which performs the task to draw the curve towards edges in the image.

The main advantage of geodesic active contour model is 1) the sound minimization of energy functional by means of the discredited steepest descent equation. 2) the implicit curve representation allows for topological changes. Hence, partially occluded objects that are split into several parts can be handled without additional efforts.

Most active contour models use the magnitude of the image gradient only in image segmentation. Recently, the directional information of image gradient [2] or the region information that is density distribution [3, Barillot] has been utilized to address the issue that the active contour may get confused in image segmentation and converge to the wrong edge when multiple edges with different directions are present near the object boundary.

$$\frac{\partial \phi}{\partial t} = g(I) \|\nabla \phi\| \left[ \operatorname{div} \left( \frac{\nabla \phi}{\|\nabla \phi\|} \right) + v \right] + \nabla g(I) \cdot \nabla \phi \quad (6)$$

$$g(I) = \frac{1}{1 + \|\nabla G_\sigma * I\|^2} \quad (7)$$

Since ACM are based on intensity gradient variations, GAC is insensitive and incapable of distinguishing between boundaries of similar intensity areas. The original GAC methods only considers topological changes and intensity gradient changes, thus it is sensitive to noisy and inhomogeneous images. Therefore, we suggest providing a few geometric priors. This information is on the approximate vertexes and some points on the boundary.

### 3.2 Segmentation Accuracy

The accuracy of the automatic segmentation via registration method was assessed by quantitatively comparing manual and automatically generated contours in terms of volume and position [14].

Regarding the volume, we calculate the similarity index (SI) for each structure. It defines the ratio between the automatic volume and the manual delineation volume by biologists,

$$SI(s) = \frac{2|V_{\text{manual}}^{(s)} \cap V_{\text{auto}}^{(s)}|}{|V_{\text{manual}}^{(s)}| + |V_{\text{auto}}^{(s)}|} \quad (8)$$

In equation (8),  $V_{\text{manual}}$  represents the number of voxels in structure s according to the manual segmentation, and  $V_{\text{auto}}$  represents the volume in s according to the automatic segmentation.

Regarding the position, the difference of coordinates (x,y,z) of the centers of mass for automatic and manual volume were compared,

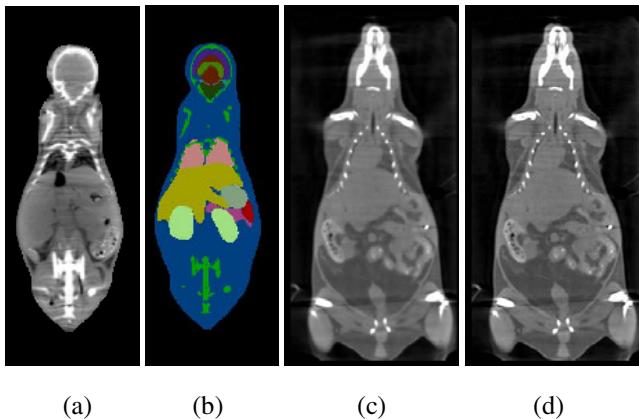
$$\Delta x = |x_{\text{auto}} - x_{\text{manual}}|, \Delta y = |y_{\text{auto}} - y_{\text{manual}}|, \Delta z = |z_{\text{auto}} - z_{\text{manual}}|.$$

## 4 Experimental Results

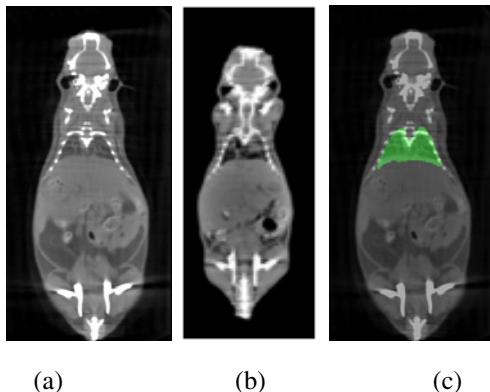
MicroCT image volume is consisted of 380 x 992 x 208 matrix and the size of each voxel is 0.1 cubic millimeters. And we use CT and PET of normal mouse to compare with an atlas and segment organ areas. PET has spatial resolution of 115 x 42 x 140 and its voxel is 0.4 x 0.4 x 0.7 millimeter. CT has 226 x 93 x 465 of 0.2 cubic millimeter voxels. As described here each image has different voxel size and spatial resolution. Therefore prior to performing image matching and segmentation resampling has to be done. Resampling using trilinear interpolation makes all image elements isotropic cubical voxels.

Before curve-based segmentation, anisotropic filter was applied with a conductance of 5, a time step of 0.0625, and 15 iterations (Fig 4.(d)).

If all image pairs aligned we need to overlay and fuse to efficiently visualize VROI, the organ areas. In “Fig. 5” transformed labeled CT are shown in (d) and both external appearance and internal boundaries are deformed to correspond mouse CT (a). And if voxels in lung area in (d) are overlaid on (a), we can easily recognize the lung region as indicated in green in (e). Then, lung area extracted from (a) are superimposed on mouse PET as in “Fig. 5 (f)”. Since there’re no glucose metabolic activities in the lung, if with no aid of coregistration between CT data the boundary of the lung couldn’t be figured out.



**Fig. 4.** Input Images and Filtering Result: (a), (b) labeled CT of anatomical atlas, (c) original image to be segmented, (d) result of noise reduction filtering.



**Fig. 5.** Segmentation result for Heart, a) input image, b) labeled CT, 3) segmentation result in coronal view

## 5 Conclusion and Future Work

We proposed a framework to segment abdominal organs in mouse CT exploiting prior anatomical knowledge from digital atlas. Proposed automatic segmentation framework reduces inter-observer, intra-observer interaction errors which are more serious when segmentation object is small.

Through the alignment between the atlas and CT two image volumes have same spatial correspondence and underlying contours for each organ in labeled CT is used as initial contour in the following segmentation step. In next step, contours are adjusted and locally modified to define organ's boundary for more accurate segmentation result.

Proposed algorithm is appropriate for blob-like organs. However it is rather difficult to be applied to organs which have tubular shape such as intestine and one whose

shape varies according to the animal's sex. We plan to develop proper method to overcome this limitation. Furthermore, we will extend our algorithm to 3D+t image data to analyze quantitative kinetics of physiological changes. This can be also used in oncological study to monitor tumor growth or regression.

## Acknowledgement

This work is financially supported by the Korea Science & Engineering Foundation through the Acceleration Research Program. And we would like to thank Korea Institute of Radiological and Medical Sciences (KIRAMS) for providing small animal images.

## References

1. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. *Intl. J. of Computer Vision*, 321–331 (1988)
2. Mdeical Image Segmentation: Methods and Software, Proceedings of NFSI & ICFBI (2007)
3. Zhu, G., Zhang, S., Zeng, Q., Wang, C.: Directional geodesic active contour for image segmentation. *J. of Electronic Imaging* (2007)
4. Liu, C., Ma, J., Ye, G.: Medical image segmentation by geodesic active contour incorporating region statistical information. In: *Intl. Conf. on Fuzzy Systems and Knowledge Discovery* (2007)
5. Rousson, M., Deriche, R.: Dynamic segmentation of vector valued images. In: *Level Methods in Imaging, Vision and Graphics*. Springer, Heidelberg (2003)
6. Tohlfing, T., Brandt, T., Menzel, R., Maurer, C.R.: Evaluation of atlas selection strategies for atlas-based image segmentation with application to confocal microscopy images of bee brains. *NeuroImage* 21, 1428–1442 (2004)
7. Maintz, J.B., et al.: A survey of medical image registration. *Medical Image Analysis* 2(1), 1–36 (1998)
8. Maroy, R., Boisgard, R., Comtat, C., Frouin, V., Cathier, P., Duchesnay, E., Dolle, F., Nielsen, P.E., Trebossen, R., Tavitian, B.: Segmentation of rodent whole-body dynamic PET images: an unsupervised method based on voxel dynamics. *IEEE Trans. on Med. Img.* 27(3), 342–354 (2008)
9. Jan, M.-L., et al.: A three-Dimensional Registration Method for Automated Fusion of micro PET-CT-SPECT Whole-Body Images. *IEEE Trans. on Med. Img.* 24(7), 886–893 (2005)
10. Chow, P.L., et al.: A Method of Image Registration for Animal, Multi-modality Imaging. *Physics in Medicine and Biology* 51, 379–390 (2006)
11. Rowland, D.J., et al.: Registration of 18f-FDG microPET and small-animal MRI. *Nuclear Medicine and Biology* 32, 567–572 (2006)
12. Shen, D., et al.: Coregistration of Magnetic Resonance and Single Photon Emission Computed Tomography Images for Noninvasive Localization of Stem Cells Grafted in the Infarcted Rat Myocardium. *Mol. Imaging Biol.* 9, 24–31 (2007)
13. Dogdas, B., Stout, D., Chatzioannou, A., Leahy, R.M.: Digimouse: A 3D Whole Body Mouse Atlas from CT and Cryosection Data. *Phys. Med. Biol.* 52(3), 577–587 (2007)

14. Evaluation of an atlas-based automatic segmentation software for the delineation of brain organs at risk in a radiation therapy clinical context. *Radiotherapy and Oncology* 87, 93–99 (2007)
15. Viola, P., Wells, W.: Alignment by maximization of mutual information. *International Journal of Computer Vision* 24(2), 137–154 (1997)
16. Studholme, V., Hill, D., Hawkes, D.: An overlap invariant entropy measure of 3D medical image alignment. *Pattern Recognition* 32(1), 71–86 (1999)
17. Kakadiaris, I.A., Bello, M., Arunachalam, S., Kang, W., Ju, T., Warren, J., Carson, J., Chiu, W., Thaller, C., Eichele, G.: Landmark-driven, atlas-based segmentation of mouse brain tissue images containing gene expression data. In: Proc. Medical Image Computing and Computer-Assisted Intervention, Saint-Malo, France, pp. 192–199 (2004)
18. Suri, J., Wilson, D.L., Laxminarayan, S.: The handbook of medical image analysis: segmentation and registration models. Springer, Heidelberg (2005)
19. Evaluation of atlas-based segmentation of hippocampal in healthy humans. *Magnetic Resonance Imaging* (2009)

# Method of Interest Points Characterization Based C-HOG Local Descriptor

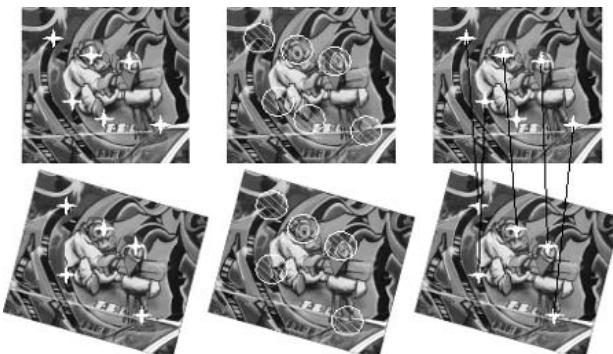
Manuel Grand-brochier, Christophe Tilmant, and Michel Dhome

LAaboratoire des Sciences et Matriaux pour l'Electronique et d'Automatique (LASMEA), UMR 6602 UBP-CNRS, 24 avenue des Landais, 63177 Aubire - France  
`firstname.surname@lasmea.univ-bpclermont.fr`

**Abstract.** This article proposes an approach to detection and description of interest points based C-HOG. The study of two interest point local descriptor methods, the SIFT and the SURF, allows us to understand their construction and extracts the various advantages (invariances, speeds, repeatability). Our goal is to couple these advantages to create a new system (detector and descriptor). The latter must be as invariant as possible for the image transformation (rotations, scales, viewpoints). We will have to find a compromise between a good matching rate and the number of points matched. All the detector and descriptor parameters (orientations, thresholds, analysis pattern, parameters) will be also detailed in this article.

## 1 Introduction

The detection of interest points and local description are two tools used in many fields of vision (pattern recognition, tracking, 3D reconstruction). The detector analyses the image to extract the characteristic points (corners, edges, blobs). The neighborhood study allows us to create a local points descriptor, in order to match them. Figure 1 illustrates these steps.



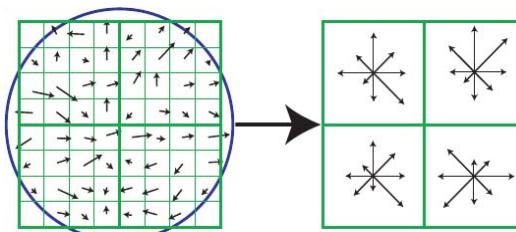
**Fig. 1.** Example of two images analysis: (left to right) interest points detection, description based on local information (neighborhood), matching

For matched interest points, the robustness of various transformations of the image is very important. To be robust to scale, interest points are extracted with a global multi-scales analysis, we considered the Harris-Laplace detector [4,9,11], the Fast-Hessian [2] and the difference of Gaussians [7,8]. The description is based on a local exploration of interest points to represent the characteristics of the neighborhood. In comparative studies [3,10], it is shown that oriented gradients histograms (HOG) give good results. Among the many methods using HOG, we retain the SIFT (Scale Invariant Feature Transform) [7,8] and SURF (Speed Up Robust Features) [2], using a rectangular neighborhood exploration (R-HOG: Rectangular-HOG). We also mention GLOH (Gradient Location and Orientation Histogram) [10] and Daisy [12], using circular geometry (C-HOG: Circular-HOG). We propose to create a system of detection and local description which is robust against the various transformations that can exist between two images (illumination, rotation, viewpoint for example). It should also be as efficient as possible as regards the matching rate. Our method relies on a Fast-Hessian points detector and a local descriptor based C-HOG. We propose to estimate local orientation in order to adjust the descriptor (rotation invariance) and we will normalize (brightness invariance).

Section 2 presents briefly SIFT and SURF, and lists the advantages of each. The various tools we use are detailed in Section 3. To compare our approach to SIFT and SURF, many tests have been carried out. A synthesis of the different results is presented in Section 4.

## 2 Related Work

SIFT [8] consists of a difference of Gaussians (DoG) and R-HOG analysis. The detector is based on an approximation of the Laplacian of Gaussian [6] and interest points are obtained by maximizing the DoG. The descriptor uses a directions histogram to determine the angle of rotation to be applied to the mask analysis. It then uses R-HOG, formed by local gradients in the neighborhood, previously smoothed by a Gaussian (Figure 2). Finally, the descriptor is normalized to be invariant to illumination changes. An extension of SIFT, GLOH [10], has been proposed to increase the robustness. It amounts to the insertion of a grid in



**Fig. 2.** The image on the left is composed of various local gradients and a circle representing the Gaussian smoothing used. The image on the right represents the shape descriptor: a set of HOG calculated on eight classes (eight angles). [8]

log polar localization. The mask analysis of this descriptor is composed of three rings (C-HOG), whose two largest are divided along eight directions. More recently, the descriptor Daisy [2] has been proposed. It is also based on a circular neighborhood exploration. Its mask consists of circles arranged in different scales and oriented in the direction of the interest point gradient.

SIFT has not a fast computational speed. SURF [2] proposes a new approach, whose main objective is to accelerate the various image processing steps. The various tests [2,5] show that the Fast-Hessian has the best repeatability rate. It is based on the Hessian matrix:

$$H(x, y, \sigma) = \begin{bmatrix} L_{xx}(x, y, \sigma) & L_{xy}(x, y, \sigma) \\ L_{xy}(x, y, \sigma) & L_{yy}(x, y, \sigma) \end{bmatrix}, \quad (1)$$

with  $L_{ij}(x, y, \sigma)$  the second derivative in the directions  $i$  and  $j$  of  $L$ . The maximization of its determinant (Hessian) allows us to extract the coordinates of interest points in a given scale. The descriptor is based on Haar wavelets (Figure 3). These estimate the local orientation of the gradient, allowing the construction of the descriptor. The tests [2] show that the results are optimal when studying the sign of the wavelet transform.



**Fig. 3.** The figure on the left represents Haar wavelets used by SURF. The figure on the right is an example of an analysis windows (scale + orientation). [2]

The presented methods use similar tools: multi-scale analysis (Fast-Hessian or DoG), HOG, local smoothing and normalization of the descriptor. For matching they use a minimization of either the Euclidean distance between descriptors (SURF) or the angle between vectors descriptors (SIFT). Many tests [10,5,3] can establish a list of different qualities of each. It follows that the SURF, with its detector, has the best repeatability for viewpoint changes, scale, noise and lighting. It is also faster than the SIFT, however it has a higher precision rate for rotations and scale changes. It has also a higher number of detected points for all transformations. It might be interesting to combine these two methods.

### 3 Method

The detector Fast-Hessian provides a list of interest points, characterized by their coordinates and local scale. Our descriptor is based on the Harris matrix

interpretation, and the construction of C-HOG. Matching is based on minimizing the Euclidean distance between descriptors. These issues will be detailed below.

### 3.1 Detection

The Fast-Hessian relies on the exploitation of the Hessian matrix (equation II), whose determinant is calculated as follows:

$$\det(H(x, y, \sigma)) = \sigma^2(L_{xx}(x, y, \sigma)L_{yy}(x, y, \sigma) - L_{xy}^2(x, y, \sigma)). \quad (2)$$

By looking for local maxima of the determinant, we establish a list of  $K$  points associated with a scale, denoted  $\{(x_k, y_k, \sigma_k); k \in [0; K-1]\}$ , where:

$$(x_k, y_k, \sigma_k) = \underset{\{x, y, \sigma\}}{\operatorname{argmax}}(\det(H(x, y, \sigma))). \quad (3)$$

The number of interest points obtained depends on the space scale explored and thresholding of local maxima.

### 3.2 Description

As with the SIFT and SURF, our method is based on HOG, yet our analysis window will consist of circles. Different tools will also be necessary to adjust and normalize our descriptor.

#### Image preprocessing

Descriptors previously cited use a Gaussian smoothing from the local scale analysis. It gives some weight around the interest point. In the case of an error in the local scale determination, the corresponding smoothing will create errors in matching. In order to minimize this problem, we propose to replace it with a smoothing overall image  $I_0$ , using the median scale of interest points, denoted by  $\sigma_M$ :

$$I(x, y) = I_0 * G_{\sigma_M}(x, y). \quad (4)$$

#### Determining the local orientation gradient

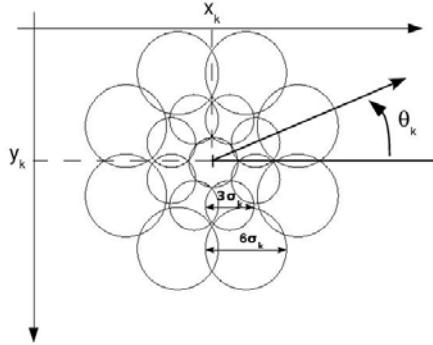
To be as invariant as possible for rotations, estimating the local orientation gradient of the interest point is necessary. This parameter allows us to adjust the HOG, corresponding to give an identical orientation for two corresponding points. For this, we use the Harris matrix, calculated for each point  $(x(k), y(k))$  and defined by:

$$M_H(x_k, y_k) = \begin{bmatrix} \sum_{V(x_k, y_k)} [I_x(x_k, y_k)]^2 & \sum_{V(x_k, y_k)} I_x(x_k, y_k) I_y(x_k, y_k) \\ \sum_{V(x_k, y_k)} I_x(x_k, y_k) I_y(x_k, y_k) & \sum_{V(x_k, y_k)} [I_y(x_k, y_k)]^2 \end{bmatrix}, \quad (5)$$

where  $V(x_k, y_k)$  represents the neighborhood of the interest point,  $I_x$  and  $I_y$  are the first derivatives in  $x$  and  $y$  of image, calculated using the Canny-Deriche operator. The properties of this matrix can study the information dispersion. The local analysis of its eigenvectors ( $v_1$  and  $v_2$ ) associated with corresponding eigenvalues can extract an orientation estimate  $\theta_k = \arctan(\vec{v_1})$ .

### Descriptor construction

Our descriptor relies on a circular neighborhood exploration of the interest point. The seventeen circles used, are divided into three scales (see Figure 4) and are adjusted by an angle  $\theta_k$ .



**Fig. 4.** Mask analysis of our descriptor, centered at  $(x_k, y_k)$  and oriented by an angle  $\theta_k$

The circle diameter is proportional to  $\sigma_k$ , thus accentuating the scale invariance. We construct a HOG eight classes (in steps of 45) for each circle. Our descriptor, we note  $des_I(x_k, y_k)$ , belongs to  $\mathbb{R}^{136}$  (17 circles  $\times$  8 directions). To be invariant for brightness changes, histogram normalization is necessary. We use also a threshold for HOG to remove the high values of gradient.

### 3.3 Matching

The objective is to find the best similarity (corresponding to the minimum distance) between descriptors of two images. Euclidean distance, denoted  $d_e$ , between two descriptors is defined by:

$$d_e(des_{I_1}(x_k, y_k), des_{I_2}(x_l, y_l)) = \sqrt{[des_{I_1}(x_k, y_k)]^T \cdot des_{I_2}(x_l, y_l)} \quad (6)$$

The minimization of  $d_e$ , denoted  $d_{min}$ , provides a pair of points  $\{(x_k, y_k); (x_{\tilde{l}}, y_{\tilde{l}})\}$ :

$$\tilde{l} = \underset{l \in [0; L-1]}{\operatorname{argmin}} (d_e(des_{I_1}(x_k, y_k), des_{I_2}(x_l, y_l))), \quad (7)$$

$$d_{min} = d_e(des_{I_1}(x_k, y_k), des_{I_2}(x_{\tilde{l}}, y_{\tilde{l}})) \quad (8)$$

To simplify the search for this minimum distance, an approximative nearest neighbor search method (a variant of k-d tree) **¶** can be used. The idea is to create a decision tree based on descriptors components of the second image. So for each new descriptor of the first image, all components are tested and the

nearest neighbor is defined. Research is therefore faster, without sacrificing precision. To have a more robust matching, thresholding is applied to this distance, to find a "high" minimum. The pair of points is valid if:

$$d_{min} \leq \alpha \times \min(d_e(des_1(x_k, y_k), des_2(x_l, y_l))), \quad \text{for } l \in [0; N - 1] \setminus \tilde{l} \quad (9)$$

with  $\alpha$  the threshold selection.

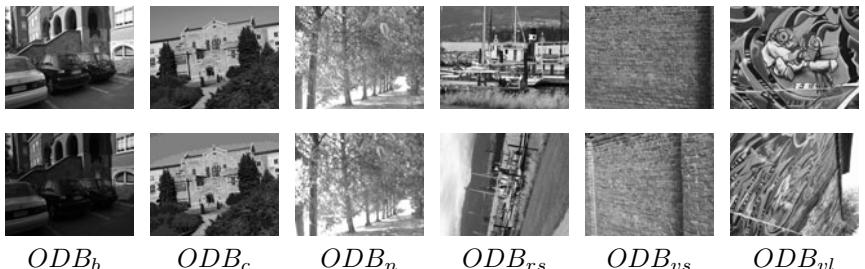
## 4 Results

We are going to compare our method with SIFT and SURF. Indeed it was demonstrated that these two methods give the best results. We propose to study the number of points detected, the matching rate and the precision of each of them.

### 4.1 Databases

To validate our method, we chose two databases:

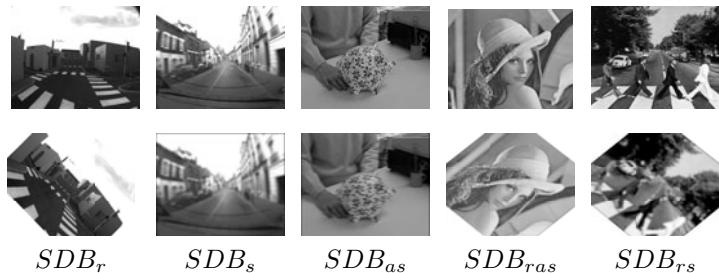
- The first one, noted *ODB* and extracted from the Oxford<sup>1</sup> base, proposes scene transformations with an access to the matrix of homography. Transformations studied are brightness changes ( $ODB_b$ ), modified jpeg compressions ( $ODB_c$ ), blur ( $ODB_n$ ), rotations and scales ( $ODB_{rs}$ ), and small and large angle viewpoint changes (respectively  $ODB_{vs}$  and  $ODB_{vl}$ ). Figure 5 illustrates this database.



**Fig. 5.** Examples of images used for transformations: (left to right) brightness changes, modified jpeg compressions, blur, rotations + scales, viewpoint changes (small angle), and viewpoint changes (large angle)

- A second database, noted *SDB*, composed of a set of synthetic image transformations (Figure 6). These transformations are rotations 45 ( $SDB_r$ ), scales ( $SDB_s$ ), anisotropic scales ( $SDB_{as}$ ), rotations 45 + scales ( $BS_{rs}$ ) and rotations 45 + anisotropic scales ( $BS_{ras}$ ).

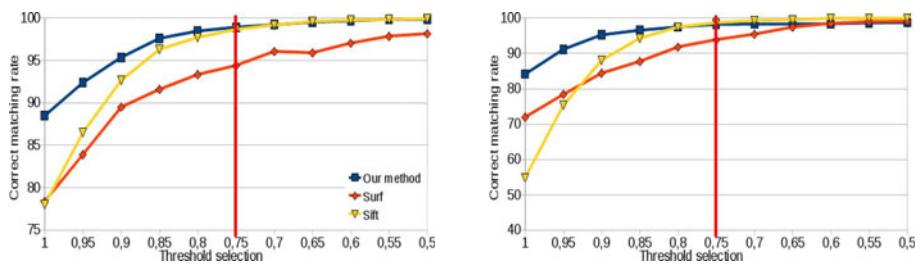
<sup>1</sup> <http://www.robots.ox.ac.uk/~vgg/data/data-aff.html>



**Fig. 6.** Examples of images (Lena, Beatles, Pig, board cameras) used for synthetic transformations

## 4.2 Threshold Selection

The threshold selection  $\alpha$  used in the equation 9 is determined by analysing the curves of Figure 7. These represent the correct matching rate according to this threshold.

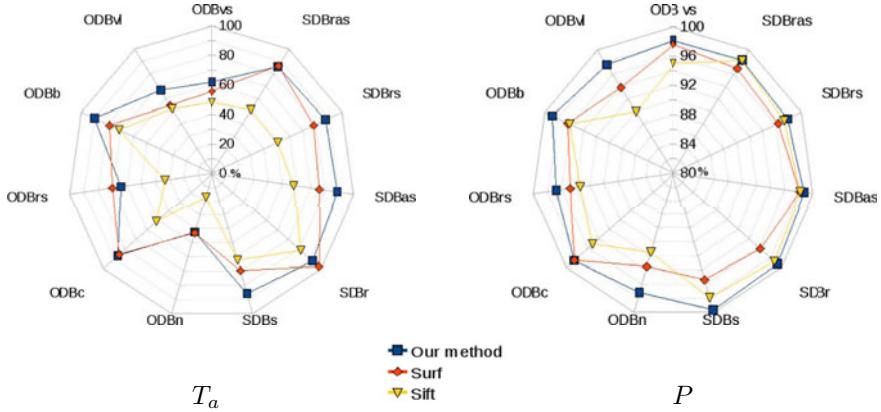


**Fig. 7.** These graphs represent the correct matching rate according to the threshold selection. At left is a viewpoint changes (graffiti 1 → 2) and at right is a rotation + scale (boat 1 → 3).

It is important to specify that the choice of  $\alpha$  has a consequence on the number of interest points detected. If the threshold goes away from 1, the matching becomes more selective and therefore fewer points are used. The problem is to find the best compromise between the correct matching rate and the number of points matched. SIFT recommends a threshold of 0.8 and SURF a threshold of 0.7. By analysing curves of Figure 7, we choose a threshold included between that of SIFT and SURF ( $\alpha = 0.75$ ).

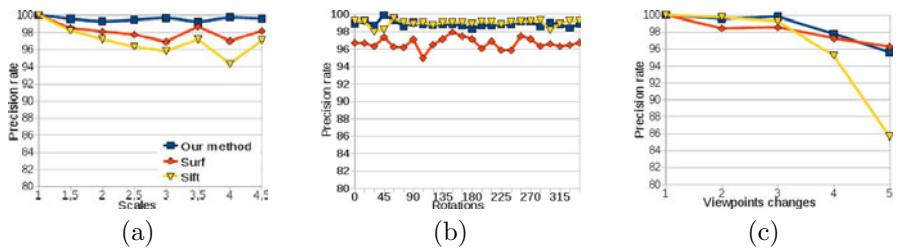
## 4.3 Evaluation Tests and Results

We propose to compare the matching rate  $T_a$ , as well as the precision  $P$  of every method.  $T_a$  is defined by the number of correct matchings divided by the number of possible matchings.  $P$  is defined by the number of correct matchings divided by the number of matchings performed. A synthesis of the results obtained for these two evaluation tests is proposed in Figure 8.



**Fig. 8.** At left is a matching rate and at right is a matching precision

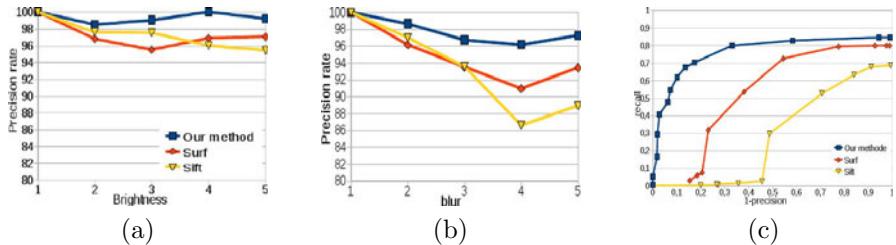
Our method presents results better or as good as SIFT and SURF. Our matching rate remains better than that of the two other methods with the exception of the databases  $ODB_{rs}$  and  $SDB_r$  transformations. Nevertheless the difference between SURF and our method for this type of transformation is lower than 4%. The biggest differences are observed for rotation 45 + scale ( $\approx 10\%$  between our method and SURF and 37% with SIFT) and for large angle viewpoint changes ( $\approx 18\%$  with SURF and SIFT). Our matching precision is also better and remains constantly above 95%. The biggest difference is obtained for large angle viewpoint changes (4% for SURF and 8% for SIFT). To detail the precision curves of different methods, we propose graphs in Figures 9 and 10.



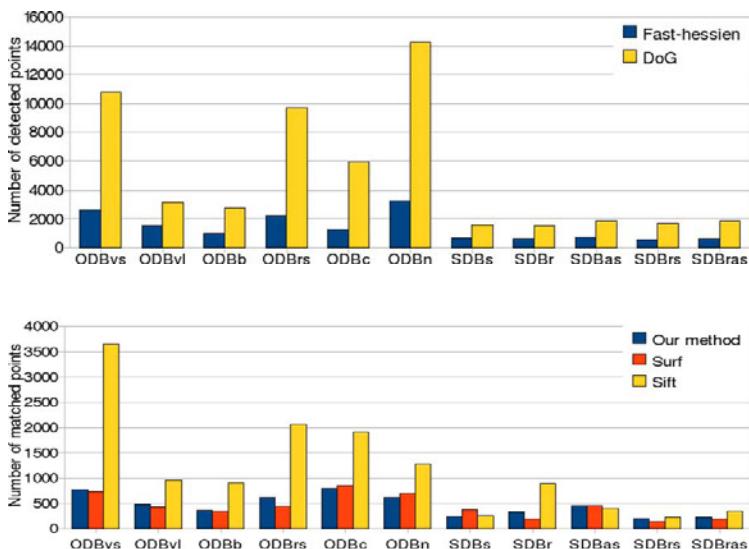
**Fig. 9.** (a) a precision rate for scales changes ( $SDB_s$ ), (b) a precision rate for rotations ( $SDB_r$ ) and (c) a precision rate for viewpoints changes ( $ODB_{vs}$ )

These show a generally higher precision rate for our method (figure 9a and figure 10a,b) or similar (figure 9b and figure 9c). Another observation can be made through the various images transformations, which concerns the stability of our method. Indeed, our curve decreases slower than the SIFT and the SURF, implying greater precision rate constancy. Figure 10c also shows that

our method is stable and more accurate for this type of transformation. However the use of the Fast-Hessian with regard to the DoG has to introduce a different number of points detected. Indeed thresholds selection are not equal, the number of points used will not be the same. Figure 11 present, on one hand, the number of interest points detected by detectors, and on the other hand, the number of points which every method was able to match.



**Fig. 10.** (a) a precision rate for brightness changes ( $ODB_b$ ), (b) a precision rate for blur ( $ODB_n$ ) and (c) a recall versus 1-precision for brightness changes ( $ODB_b: 1 \rightarrow 4$ )



**Fig. 11.** The graphs present, for the three methods studied, (top) the number of detected points and (bottom) the number of matched points (good+false)

Our method and SURF have the same initial number of points (same detector). The number of points matched is more or less equivalent. Concerning the comparison with SIFT, our method has a much smaller number of points detected. Nevertheless, our method shows better results concerning the correct matching rate that SIFT and SURF. Our matching precision is also generally better.

## 5 Conclusion and Prospect

In this article we presented a study of SIFT and SURF. We deducted from it the different advantages of each (repeatability, speed, invariances). Then we proposed a method based on the Fast-Hessian detector and the C-HOG descriptor. The tools (Harris matrix, filtering, threshold) have also been detailed. The evaluation tests presented validate our method. This has a lower number of detected points than SIFT and equal to SURF. However, our matching rate and our precision are superior and more robust.

Our prospects are a generalization of our method, with application to a three-dimensional or spatio-temporal analysis. One application is referred to the treatment of medical imaging. The integration of our method in a vision system is also being studied to improve the tracking and the 3D reconstruction.

## References

1. Arya, S., Mount, D.M., Netanyahu, N.S., Silverman, R., Wu, A.: An optimal algorithm for approximate nearest neighbor searching in fixed dimensions. *J. ACM* 45, 891–923 (1998)
2. Bay, H., Tuytelaars, T., Van Gool, L.: Surf: Speeded up robust features. In: European Conference on Computer Vision (2006)
3. Choksuriwong, A., Laurent, H., Emile, B.: Etude comparative de descripteur invariants d'objets. In: ORASIS (2005)
4. Harris, C., Stephens, M.: A combined corner and edge detector. In: Alvey Vision Conference, pp. 147–151 (1988)
5. Juan, L., Gwun, O.: A comparison of sift, pca-sift and surf. *International Journal of Image Processing* 3(4), 143–152 (2009)
6. Lindeberg, T.: Feature detection with automatic scale selection. *International Journal of Computer Vision* 30(2), 79–116 (1998)
7. Lowe: Object recognition from local scale-invariant features. In: IEEE International Conference on Computer Vision, pp. 1150–1157 (1999)
8. Lowe: Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision* 60(2), 91–110 (2004)
9. Mikolajczyk, K., Schmid, C.: An affine invariant interest point detector. In: European Conference on Computer Vision, vol. 1, pp. 128–142 (2002)
10. Mikolajczyk, K., Schmid, C.: A performance evaluation of local descriptors. *IEEE Pattern Analysis and Machine Intelligence* 27(10), 1615–1630 (2004)
11. Mikolajczyk, K., Schmid, C.: Scale & affine invariant interest point detectors. *International Journal of Computer Vision* 1(60), 63–86 (2004)
12. Tola, E., Lepetit, V., Fua, P.: A fast local descriptor for dense matching. In: IEEE Conference on Computer Vision and Pattern Recognition (2008)

# Stereo-Based Object Segmentation Combining Spatio-Temporal Information

Yingdong Ma and Qian Chen

Centre for Digital Media Computing  
Shenzhen Institutes of Advanced Technology, Shenzhen, China

**Abstract.** In the case of cluttered backgrounds or low quality video input, automatic video object segmentation based on spatial-temporal information is still a problem without a general solution. A new approach is introduced in this work to deal with this problem by using depth information. The proposed approach obtains the initial object masks based on depth density image and motion segmentation. The objects boundaries are obtained by updating object masks using a simultaneous combination of multiple cues, including spatial location, colour, depth and motion, within a maximum likelihood method. The experimental result shows that this method is effective and has good output in cluttered backgrounds.

## 1 Introduction

As one of the most important step towards object-based representation, the problem of extracting video contents at semantic level has received continuously increasing interests. Most of the existing video object segmentation and tracking algorithms are spatial-temporal feature based [1]-[3]. However, due to the limited information recorded at each pixel in a video sequence, these algorithms cannot always guarantee successful semantic objects detection and tracking in the case of cluttered backgrounds or low quality video input.

Combining 3D scene data with other visual features is a step forward to robust object segmentation. Some depth-based segmentation algorithms have been introduced in recent years. While some segmentation methods focus on the usage of depth information only [4], most of the depth-based segmentation approaches incorporating with other spatial and temporal segmentation methods to improve segmentation performance [5]-[7].

Most depth-based segmentation methods rely on a dense disparity map to generate object masks. However, due to the limitation of stereo matching methods, the 3D data generated by these methods are still noisy and have a lower degree of accuracy than that of intensity data. This work aims at developing an accurate and robust video object extraction algorithm by combining depth and other spatial-temporal data. Firstly, a depth density image is generated from the disparity map. Foreground object regions are obtained from the depth density image by means of a region growing method. These regions are then combined with change detection masks which are calculated from multiple consecutive frames with an adaptive thresholding scheme to

form object masks. A Markov Random Field (MRF) model which simultaneously combines multiple cues is employed to get the final object segmentation.

The advantage of using a depth density image is that errors caused by imperfect stereo matching are filtered out by the accumulation effect of the density image. Therefore, do not significantly influence the performance of foreground object detection. In addition, the depth-based segmentation does not need to be very accurate since the object boundaries can be further refined by motion-based segmentation and Bayesian segmentation.

The next section presents the depth density image based object detection algorithm. Section three describes the generation of the initial object masks using depth and motion information. A Bayesian segmentation algorithm is introduced in section four, in which multiple cues are combined to find the object boundaries. Finally, in section five and section six, experimental results and a conclusion of the system are presented.

## 2 Object Detection Using Depth Density Image

To achieve robust object detection, the depth based object detection algorithm consists of two stages. In the first stage, a depth density image is generated by transforming depth information on the XY plane (disparity map) to the XZ plane. The foreground object regions are detected in the second stage by using a region growing method.

In this work, a pair of calibrated CCD cameras is used to get the left and right image sequences. Region-based stereo matching algorithm is selected for its capability of reliable disparity map generation [8].

### 2.1 Depth Density Image Generation

In order to recover the depth information from a disparity map, a depth density image is generated by transforming the depth information on the XY plane (disparity map) to the XZ plane. Points are projected to the XZ plane according to their horizontal position and their grey level, where X is the width of the depth map and the range of Z is [0, 255]. Because an object has similar grey level in the disparity map, the influence of the 3D points of this object is cumulative on the depth density image. The depth density image will contain large values in areas with a high density of 3D points.

Some points in the disparity map correspond to those lying on the ground surface, for example, marks and object shadows on carpet. These points are regarded as noisy points and should be discarded. Assuming a planar ground surface, the camera height is  $C$ , and the lens has a tilt angle  $\theta$  towards the ground plane. The height of points in the disparity map can be computed as:

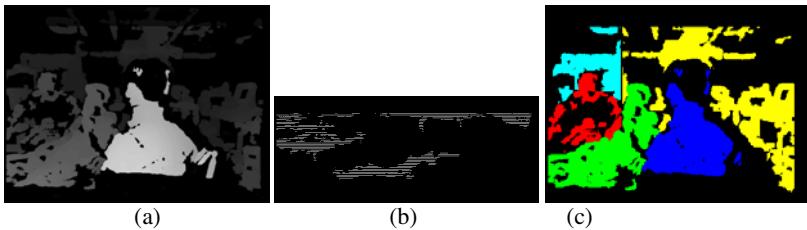
$$H = (Z \times \sin(\theta) - C) / \cos(\theta) \quad (1)$$

For those points with height value greater than H, we consider them as object points above the ground plane.

## 2.2 Depth-Based Object Region Detection

Foreground object regions segmentation is performed based on the depth density image using a region growing method. After a morphological opening and closing operation to remove noisy points and connect nearby points, all 8-orientation-connected points are grouped as one object.

The disparity map is segmented according to the depth density image. The x-coordinates of each connected component in the depth density image shows the location and width of each foreground object. All corresponding points in the disparity map are labelled as different objects according to their grey level and others are labelled as a background image.



**Fig. 1.** Depth-based segmentation. (a) Disparity map. (b) Depth density image. (c) Depth-based object detection.

## 3 Motion Segmentation

Due to the stereo matching ambiguity or depth discontinuities at textureless areas, the object masks obtained from depth-based method are more likely to be inaccurate and therefore, require further refinement. In this work, the motion mask generated by change detection method is used to refine the silhouette of segmented objects. The proposed motion segmentation approach represents changing pixels between two frames in a change detection mask. The determination of these changing pixels is based on an adaptive threshold estimation method. Moving background elimination is achieved using the background image. If a point belongs to the current background image, its frame difference value is zero.

The change detection mask (CDM) is derived from two difference images. The first difference image is obtained from the previous left frame and the current left frame, and the second difference image is generated from the current left frame and the next left frame. If a point is marked as a changing point in the two difference images, it is labelled as a CDM point. In addition, the difference between the current frame and the background image is also taken into account to ensure that stationary foreground objects are segmented.

$$FD_{k,k-1}(x, y) = \begin{cases} 0 & \text{if } (x, y) \in BI_k \text{ \& } (x, y) \in BI_{k-1} \\ |I_k(x, y) - I_{k-1}(x, y)| & \text{otherwise} \end{cases} \quad (2)$$

$$BD_k(x, y) = |I_k(x, y) - BI_k(x, y)| \quad (3)$$

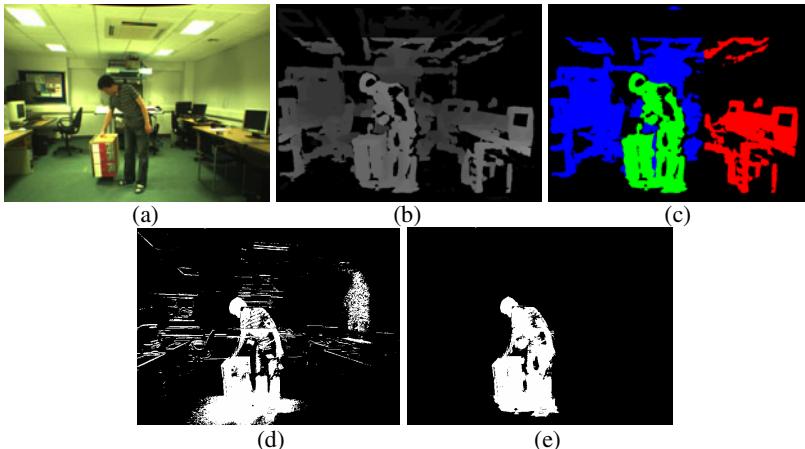
$$CP_{k,k-1}(x, y) = \begin{cases} 1 & \text{if } FD_{k,k-1}(x, y) \text{ or } BD_k(x, y) \geq T_c \\ 0 & \text{otherwise} \end{cases} \quad (4)$$

$$CDM_k(x, y) = \begin{cases} 1 & \text{if } CP_{k,k-1}(x, y) = CP_{k,k+1}(x, y) = 1 \\ 0 & \text{otherwise} \end{cases} \quad (5)$$

where  $FD$  is the frame difference,  $BD$  is the background difference,  $CP$  is the changing point label, and  $CDM$  is the change difference mask.  $BI_k$  and  $BI_{k-1}$  are the current background image and previous background image, respectively.  $T_c$  is the threshold whose value is determined using the least-half-samples (LHS) technique [9] on each pair of images to be processed. Finally, the object masks are calculated as:

$$OM_k(x, y) = \begin{cases} 1 & \text{if } CDM_k(x, y) = 1 \& (x, y) \in ORD_i \\ 0 & \text{otherwise} \end{cases} \quad (6)$$

where  $OM$  is the object mask and  $ORD_i$  is an object region in the disparity map. The interior of each object mask can be filled using a fill-in technique developed in [10]. The combination of depth map segmentation and motion segmentation benefits the shadow cancellation in the background area as shown in Fig.2.



**Fig. 2.** Object mask generation. (a) Original frame. (b) Disparity map. (c) Depth-based segmentation. (d) Change detection mask. (e) Object mask.

## 4 Object Boundary Refinement

The object masks provide the number and location information of the moving objects. However, due to noise and low quality video input, the boundaries of these moving objects are often inaccurate. In this work, a Markov Random Field (MRF) model, which simultaneously combines multiple cues, is employed to refine the object

boundaries. The advantage of using a MRF model consists of the fact that it integrates the spatial relationship within a local area of each pixel and multiple features can be utilized in a MRF model based on the Bayesian segmentation framework.

Assume that there are  $m$  independently moving objects in a frame,  $F_t$ , and each of them has been labelled in the previous step. The object segmentation problem is to find an optimal label  $l(x, y)$  given an optical flow vector  $v(x, y)$ , a pixel colour value, and an initial label (the initial object masks).

Let the probability of a pixel  $(x, y)$  in frame  $F_t$ , belonging to object class  $l_i$  ( $1 \leq i \leq m$ ), be  $P(l_i | f_t(x, y))$ , where  $f_t(x, y)$  is the feature vector of pixel  $(x, y)$  in the current frame. According to the Bayesian rule, the optimal label field  $L = \{L_i | l_i \in [0, \dots, m]\}$  can be estimated by maximizing a posterior probability:

$$P(L | f_t) \propto P(f_t | L)P(L) \quad (7)$$

The first condition pdf  $P(f_t | L)$  reflects how well the segmentation conforms to the current frame. It can be broken down into a product of two probability terms of two feature components: the motion feature and the pixel colour feature. The calculation of the probability is based on an assumption that the distribution of all feature data is a Gaussian function with different means  $\mu_i$  and covariance matrix  $\Sigma_i$ .

$$P(f_t(x, y) | l_i) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_i|^2}} \exp\left\{-\frac{(f_t - \mu_i)^T \Sigma_i^{-1} (f_t - \mu_i)}{2 |\Sigma_i|^2}\right\} \quad (8)$$

where  $d$  is 2 for motion pdf and 3 for colour pdf,  $\mu_i$  and  $\Sigma_i$  are the mean and the covariance matrix of the feature vectors of all the pixels belong to object class  $l_i$ . In this work, the YUV colour space is used for colour distance computation.

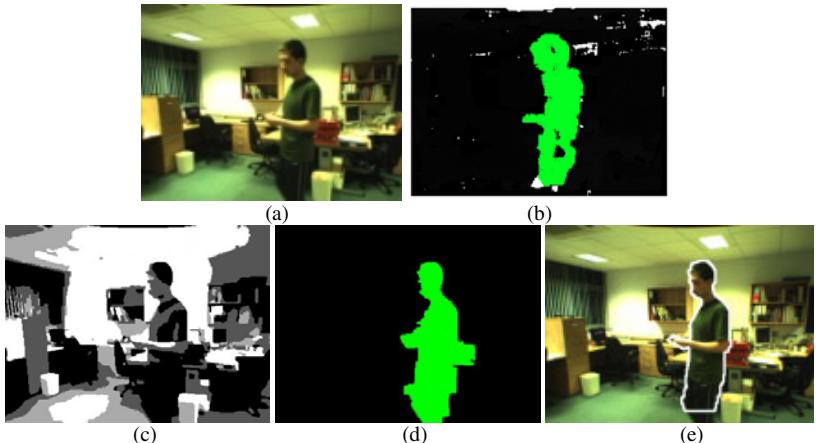
$P(L)$  is the a priori probability of the segmentation field which describes the label distribution of previous depth and motion segmentation results, and is refer to the region labelling component. Its energy can be computed as:

$$U_L(x) = \sum_s \sum_{t \in N_s} V_c(l(x_s), l(x_t)) \quad (9)$$

$$V_c(l(x_s), l(x_t)) = \begin{cases} -\gamma & \text{if } z(x_s) = l(x_t) \\ +\gamma & \text{otherwise} \end{cases} \quad (10)$$

The cost function can be minimized efficiently using the Graph-cuts method developed by Boykov et al. [11].

Since the object mask is close to the object boundary, the iteration number of the algorithm is set as three. When the foreground objects have similar colour to that of the background, the object boundaries obtained from Bayesian segmentation might be inaccurate. The final object boundary is obtained by combining the object mask and the Bayesian segmentation as shown in Fig.3.



**Fig. 3.** Object segmentation. (a) Original frame. (b) Depth-based segmentation. (c) Bayesian segmentation. (d) Object mask from the MRF based segmentation. (e) Object segmentation by combining (b) and (d).

## 5 Experimental Results

Most of the test sequences are obtained using a Digiclops® camera, attached to a personal computer with a 3.00GHz Pentium Xeon processor. The test platform can generate 12 depth frames per seconds (fps), with a resolution of  $320 \times 240$  pixels and 7 fps with a resolution of  $640 \times 480$  pixels. The algorithm performance depends on the number of foreground objects. For a typical office scene with one to three moving objects, the proposed depth, motion, and MRF model based algorithm can achieve 4.2 fps for videos with a resolution of  $320 \times 240$  pixels.

Fig.4 and Fig.5 show the object segmentation results using the proposed depth-based segmentation method. The video sequences in these two figures have resolution of  $320 \times 240$  pixels and  $640 \times 480$  pixels, respectively. The poor quality of the disparity map, the texture-less moving object and the cluttered background are the main challenges in these examples.



**Fig. 4.** Depth, motion and MRF model based object segmentation of frame 1, 7, 11, and 16 (left to right) of a low resolution video sequence

As an algorithm comparison, the video sequence as shown in Fig.5 is used to test other tracking techniques. In Fig.6, the segmentation results are obtained from the depth and active contour model based method proposed by Ntalianis et al. [12].



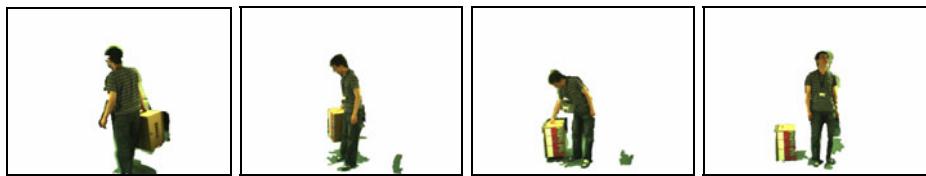
**Fig. 5.** Depth, motion and MRF model based object segmentation of frame 25, 31, 37, 43, 55, and 61 (top to down, left to right) of a test video sequence



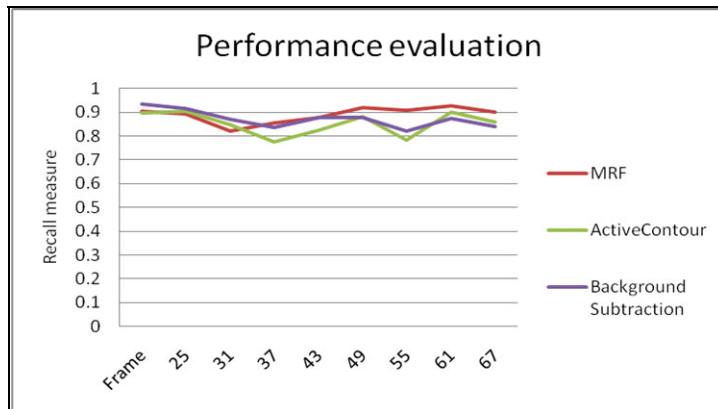
**Fig. 6.** Depth and active contour model based object segmentation [12] of frame 25, 31, 37, 43, 55, and 61 (top to down, left to right) of the test video sequence as shown in Fig.5

This example illustrates that, due to the combination of motion and colour features of each pixel, the MRF model based method has better performance than the active contour model based method. The advantage of active contour model based method is that it has lower computational cost. It achieves 8.6 fps for videos with a resolution of  $320 \times 240$  pixels. Fig.7 gives the background subtraction based segmentation [13] results of the same test video sequence as shown in Fig.5.

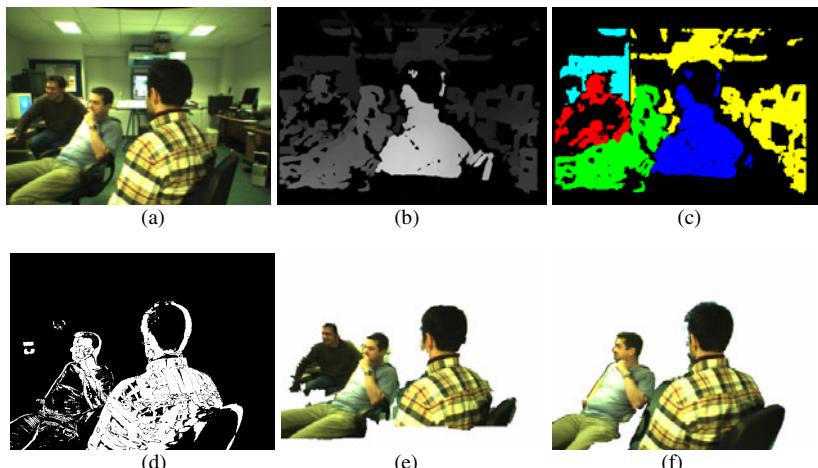
The performance evaluation of the proposed segmentation algorithm, the active contour model based method, and the background subtraction method in terms of the recall measurement is shown in Fig.8. The ground truth of the video sequence is obtained by manual segmentation. From Fig.8 we can observe that the proposed algorithm achieves better segmentation performance than that of the other two methods. The proposed algorithm has lower recall values before frame 31 mainly because of the unavailable depth data near the frame bottom area.



**Fig. 7.** Background subtraction based segmentation [13] of frame 31, 37, 43, and 49 (left to right) of the test video sequence as shown in Fig.5



**Fig. 8.** Performance comparison between the proposed algorithm, the active contour model based method, and the background subtraction method



**Fig. 9.** Object segmentation results with and without depth map. (a) colour frame. (b) Disparity map. (c) Depth-based segmentation. (d) Mmotion mask. (e) Depth-motion-MRF model based segmentation. (f) Motion and colour based segmentation.

Fig.9 illustrates the object segmentation results with and without depth map segmentation. In the case of stationary foreground objects have similar color with the background, spatio-temporal based methods cannot guarantee correct segmentation result as shown in Fig.9.f. However, with the proposed depth-based segmentation method, the correct object masks can be obtained (see Fig.9.e).

## 6 Conclusion

We developed a depth-based video object segmentation algorithm aiming at improving the robustness and accuracy of object segmentation from cluttered background. In the proposed algorithm, the object masks are computed by using depth and motion information. A MRF model is utilized to combine multiple features for further object boundaries refinement.

The novelty of the proposed system mainly consists of the depth density image based object detection and the multi-feature based object boundary refinement, which combines depth and spatial-temporal information for better segmentation performance. Object detection using depth density image is an error resilient algorithm because of its accumulation effect. The depth and motion based object mask generation method provides accurate object shape estimation, which is closed to object boundary and ensures that the MRF model converges quickly.

## References

1. Chien, S., Huang, Y., Hsieh, B., Ma, S., Chen, L.: Fast video segmentation algorithm with shadow cancellation, global motion compensation, and adaptive threshold techniques. *IEEE Trans. on Multimedia* 6, 732–748 (2004)
2. Piroddi, R., Vlachos, T.: A simple framework for spatio-temporal video segmentation and delayering using dense motion fields. *IEEE Signal Processing Letters* 13(7) (2006)
3. Wang, Y., Loe, K., Wu, J.: Spatiotemporal video segmentation based on graphical models. *IEEE Trans. on Image Processing* 14(7) (2005)
4. Parvizi, E., Wu, Q.M.J.: Multiple object tracking based on adaptive depth segmentation. In: Proceedings of the IEEE Conference on Computer and Robot Vision, pp. 273–277 (2008)
5. Nedevschi, S., Bota, S., Tomiuc, C.: Stereo-based pedestrian detection for collision-avoidance applications. *IEEE Trans. on Intelligent Transportation Systems* 10(3) (2009)
6. Cardoso, J.S., Cardoso, J.C.S., Corte-Real, L.: Object-based spatial segmentation of video guided by depth and motion information. In: Proc. IEEE workshop Motion and Video Computing, WMVC 2007 (2007)
7. Cigla, C., Alatan, A.A.: Object segmentation in multi-view video via colour, depth and motion cues. In: Proc. 15th IEEE International Conference on Image Processing, pp. 2724–2727 (2008)
8. Brown, M.Z., Burschka, D., Hager, G.D.: Advances in computational stereo. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 25(8) (2003)
9. Ong, E.P., Tye, B.J., Lin, W.S., Etoh, M.: An efficient video object segmentation scheme. In: Proc. International Conference on Acoustics, Speech, and Signal Processing, vol. 4, pp. 3361–3364 (2002)

10. Meier, T., Ngan, K.N.: Video segmentation for content-based coding. *IEEE Transactions on Circuits and Systems for Video Technology* 9(8) (1999)
11. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimisation via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 29, 1222–1239 (2001)
12. Ntalianis, K.S., Doulamis, A.D., Doulamis, N.D., Kollias, S.D.: Unsupervised VOP segmentation of stereo-captured video sequences (2008)
13. Spagnolo, P., Orazio, T.D., Leo, M., Distante, A.: Moving object segmentation by background subtraction and temporal analysis. *Image and Vision Computing* 24(5), 411–423 (2006)

# Fast Motion Estimation Based on Search Range Adjustment Using Neighboring MVDs

Hyun-Soo Kang and Jae-Hyeung Park

College of ECE, ChungBuk National University, Chungju, Korea  
`{hskang,jh.park}@cbnu.ac.kr`

**Abstract.** In this paper, we propose a new adaptive search range motion estimation method for H.264/AVC where search ranges are adjusted by the probabilities of motion vector differences (MVDs). The MVDs are modeled as a discrete Laplace distribution and then its parameter is estimated by the maximum likelihood estimator. The MVDs of neighboring blocks are employed as the samples for the estimation. With the estimated distribution, the search ranges which include the correct MVDs for a prefixed probability are analytically determined. Since the proposed method handles the search ranges instead of search point sampling patterns, it provides very flexible and hardware-friendly approach in motion estimation. Experimental results show that it is very similar to the optimal method (full search algorithm) in PSNR but gives significant reduction in the computational complexity.

## 1 Introduction

Motion estimation (ME) has been widely adopted by many video coding standards such as H.264/AVC [1] because of excellent elimination of temporal redundancy in images. However, motion estimation is very intensive in computation. To reduce the computational complexity, many fast algorithms, such as three-step search [2], four-step search [3], diamond search [4-5], and hexagon-based search [6-8],[11], have been proposed. These algorithms achieve less computational complexity by sampling of the search points in a search range with their inherent search patterns. Although these fast algorithms considerably reduce the computational complexity, they often fall in local minima to cause some degradation. Furthermore, due to their sequential behavior, these algorithms are not hardware-friendly.

On the other hand, adaptive search range (ASR) methods are hardware-friendly because there is no conditional branch after search ranges are fixed and they can be realized by regular structures such as systolic arrays. They can also save the memory bandwidth which is the number of hardware clocks required for reading the pixels in a search range from external frame memories to internal memories of a motion estimation module. The clock cycles saved by adopting the ASR methods can be assigned to the motion estimation module in order to reduce its hardware complexity. In addition, since their search areas are rectangular and all pixels in the areas are available, the ASR methods can be easily combined with the search point sampling based methods mentioned above.

The ASR methods which are based on arithmetic computations for motion vectors (MVs) of neighboring blocks have been proposed [9-12]. In [11], an ASR method is employed prior to hexagonal searching so that it may reduce the number of search points. Lee et al. [13] introduced an ASR method based on motion estimation errors of neighboring blocks. Z. Chen et al. [14] presented an ASR method based on an MV estimation of a current block where the MV of the collocated block was considered as an estimate of the MV of the current block and a motion vector difference (MVD) was computed by using the estimated vector and a motion vector predictor (MVp).

In this paper, a new ASR method using the probability density function (PDF) of MVDs is proposed as follows. Firstly, the PDF is estimated by the maximum likelihood estimator. Then, considering the PDF, a search range is determined by a given probability that a correct MVD is included in the search range.

## 2 Estimation of the Distribution of Motion Vector Differences

In H.264/AVC, the center position of a search area in motion estimation is indicated by MVp. When a search range is set to  $SR$ , therefore, the corresponding MVD is within  $\pm SR$  as the search area is restricted within  $MVp \pm SR$ . As the search range is generally fixed as an input parameter to a video encoder, computing power may be wasted because the search range can be much larger than motions of input image sequences. Thus, if we have any indications that the MVD is within  $\pm k$ , the search area can be reduced to  $MVp \pm k$ . In this sense, if the MVD distribution is known, we can obtain the probability that the MVD falls in a given search range. Reversely, given the probability, we can fix the search range which contains the MVD with the probability. For the purpose of fixing the search range, in this paper, the MVD distribution is estimated by the maximum likelihood estimation method.

Since the MVp of H.264/AVC is a good predictor for MV prediction,  $x$  and  $y$  components of MVDs may follow Laplace distributions. Fig.1 shows the distributions of  $x$  and  $y$  components which are resulted by H.264/AVC encoding for Foreman (CIF) 100 frames. As the search range is identified by integer pixel unit, the values have been rounded off to the nearest integer numbers. Fig. 1 exhibits that the MVD components can be well approximated by Laplace distributions.

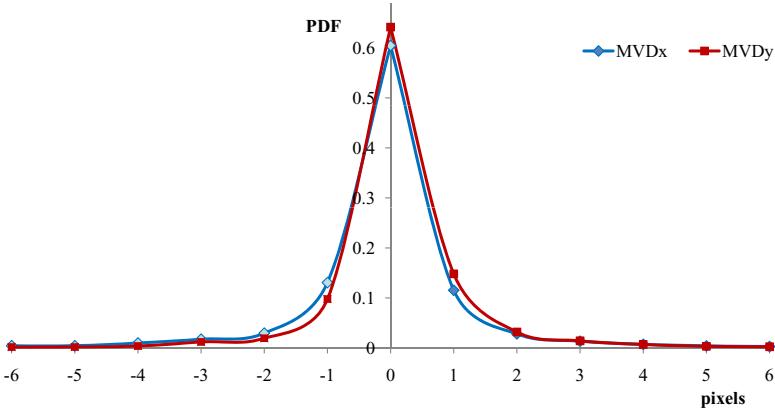
Considering that MVD data is a discrete valued signal, the original Laplace distribution is slightly modified into:

$$f_Z(z) = \tanh(\alpha/2) \cdot e^{-\alpha|z|} \quad (1)$$

such that

$$\sum_{z=-\infty}^{\infty} f_Z(z) = 1 \quad (2)$$

In Eq. (1),  $Z \in \mathbf{I}$  is a discrete random variable which corresponds to  $x$  or  $y$  components of MVD vectors, and  $\alpha$  is a positive constant. Although the PDF



**Fig. 1.** Distributions of  $x$  and  $y$  components of MVDs in Foreman CIF sequence

is given by an integer pixel unit representation for convenience, it can be easily converted to a quarter pixel unit expression by scaling.

As a result, the  $x$  and  $y$  components of MVD vectors, namely MVDx and MVDy, are assumed to follow:

$$f_X(x) = \tanh(\alpha_x/2) \cdot e^{-\alpha_x|x|} \quad (3)$$

$$f_Y(y) = \tanh(\alpha_y/2) \cdot e^{-\alpha_y|y|} \quad (4)$$

where  $X$  and  $Y$  are random variables which correspond to MVDx and MVDy, respectively. Additionally, assume that the  $x$  and  $y$  components are independent of each other. This assumption is reasonable because MVDx and MVDy are independently predicted error signals as well as each of them are uncorrelated by oneself due to good prediction. When MVDx and MVDy are independent, we have a joint PDF:

$$f_{XY}(x, y) = \tanh(\alpha_x/2) \tanh(\alpha_y/2) \cdot e^{-(\alpha_x|x| + \alpha_y|y|)} \quad (5)$$

In case of  $\alpha_x = \alpha_y$ , we have more concise PDF

$$f_{XY}(x, y) = \tanh^2(\alpha/2) \cdot e^{-\alpha(|x| + |y|)}, \text{ where } \alpha \equiv \alpha_x = \alpha_y \quad (6)$$

Having the PDF of MVDs, we describe how to find the parameters,  $\alpha_x$  and  $\alpha_y$ . Assuming  $N$  independent and identically distributed samples (MVD vectors),  $\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N$ , an estimate  $\hat{\boldsymbol{\alpha}}$  of  $\boldsymbol{\alpha}$  can be obtained by the maximum likelihood estimation, i.e.

$$\begin{aligned} \hat{\boldsymbol{\alpha}} &= \max_{\boldsymbol{\alpha}} l(\boldsymbol{\alpha} | \mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N) \\ &= \max_{\boldsymbol{\alpha}} \prod_{i=1}^N \tanh(\alpha_x/2) \tanh(\alpha_y/2) \cdot e^{-(\alpha_x|x_i| + \alpha_y|y_i|)} \end{aligned} \quad (7)$$

where  $\hat{\boldsymbol{\alpha}} = (\hat{\alpha}_x, \hat{\alpha}_y)$ ,  $\boldsymbol{\alpha} = (\alpha_x, \alpha_y)$ ,  $\mathbf{s}_i = (x_i, y_i)$ , and  $l(\cdot)$  is a likelihood function.

As maximizing a logarithmic form of the likelihood function, i.e.  $\ln[l(\alpha|\mathbf{s}_1, \mathbf{s}_2, \dots, \mathbf{s}_N)]$ , with respect to  $\alpha_x$  and  $\alpha_y$ , we have:

$$\hat{\alpha}_x = \begin{cases} \sinh^{-1}(1/\hat{\mu}_x), & \hat{\mu}_x \neq 0 \\ 0, & \hat{\mu}_x = 0 \end{cases} \quad (8)$$

$$\hat{\alpha}_y = \begin{cases} \sinh^{-1}(1/\hat{\mu}_y), & \hat{\mu}_y \neq 0 \\ 0, & \hat{\mu}_y = 0 \end{cases} \quad (9)$$

$$\text{where } \hat{\mu}_x = \frac{1}{N} \sum_{i=1}^N |x_i|, \quad \hat{\mu}_y = \frac{1}{N} \sum_{i=1}^N |y_i|$$

In case of  $\alpha_x = \alpha_y$ , similarly, we have:

$$\hat{\alpha} \equiv \hat{\alpha}_x = \hat{\alpha}_y = \begin{cases} \sinh^{-1}(1/\hat{\mu}), & \hat{\mu} \neq 0 \\ 0, & \hat{\mu} = 0 \end{cases} \quad (10)$$

$$\text{where } \hat{\mu} = \frac{1}{2N} \sum_{i=1}^N |x_i| + |y_i|$$

These equations explain that the estimated parameters are related to the average of the absolute values of MVD components.

Now let's more investigate the special case of  $\alpha_x = \alpha_y$ . In the case, as the PDF is identical to  $x$  and  $y$  directions, a square shaped search range which is preferred in hardware implementation is resulted for a given error probability, which is called missing probability in the following description and will be explained in detail. Besides, it can reduce a undesirable problem which occurs in estimation of  $\alpha_x$  and  $\alpha_y$  for  $\alpha_x \neq \alpha_y$ . As a lot of MVD components are null, either  $\hat{\mu}_x$  or  $\hat{\mu}_y$  is frequently concluded to be zero, which leads the search range to be zero so that poor motion estimation is caused. To relieve this problem, considering the special case can be a good option because the probability of  $\hat{\mu}$  being zero is less than that of either  $\hat{\mu}_x$  or  $\hat{\mu}_y$  being zero. Thus, the investigation into the special case will be described in the remainder.

Provided that the distribution is well estimated by the equations above, we can find the probability that an MVD is fallen within a given search range. Consider an event  $\{X > k \text{ or } Y > k\}$  that corresponds to the event where a correct MVD is beyond bound of the search range  $k$ . Then, we need to restrict the probability of the event under a certain value for good motion estimation. Accordingly, introducing the constraint that  $P\{X > k \text{ or } Y > k\} \leq \epsilon^2$ , where  $\epsilon^2$  is called missing probability, we can choose  $k$ . The constraint gives:

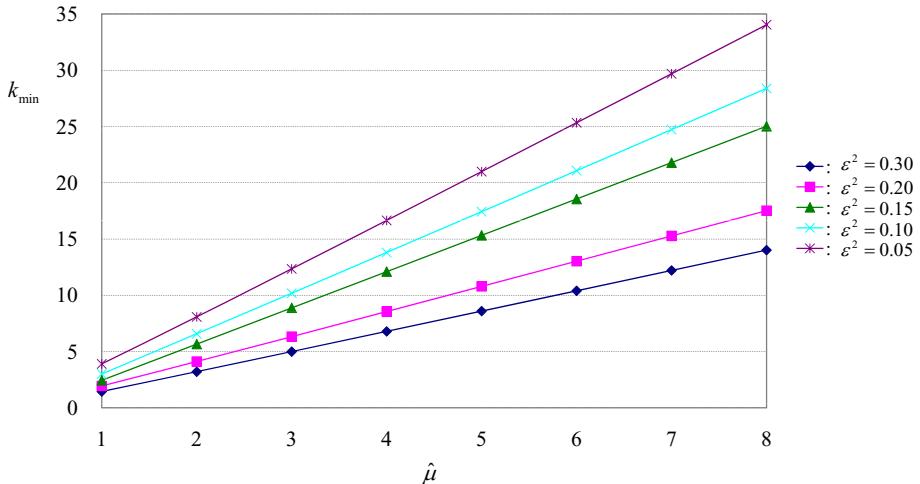
$$k \geq -1 - \frac{1}{\alpha} \ln \left[ \frac{\epsilon}{2} (1 + e^{-\alpha}) \right] \equiv k_{\min} \quad (11)$$

Substituting the estimated parameter  $\hat{\alpha}$  above into Eq. (11), the search range,  $k_{\min}$  for  $\alpha_x = \alpha_y$ , can be estimated when given the missing probability.

$$\hat{k}_{\min} = -1 - \frac{1}{\hat{\alpha}} \ln \left[ \frac{\epsilon}{2} (1 + e^{-\hat{\alpha}}) \right] \quad (12)$$

The relation in Eq. (12) was plotted in Fig. 2. For instance, if  $\varepsilon^2 = 0.1$  and  $\hat{\mu} = 2$ , then  $\hat{k}_{min} = 6.6$ . This reveals that a correct MV is in the range of  $MVp \pm 6.6$  with the probability of 90%. As seen in Fig. 2, the relation appears in a nearly linear behavior which can be approximated by the first order polynomials and hence Eq. (12) can be computationally simplified.

Applying the analytic results above, conclusively, the search range can be managed by the missing probability so that the performance of motion estimation may meet our desire.



**Fig. 2.** Search range ( $k_{min}$ ) according to mean values of samples ( $\hat{\mu}$ )

### 3 Proposed Method

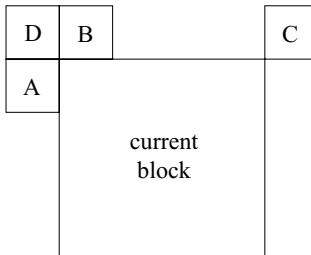
In the previous section, we have described the PDF of MVD vectors. In practice,  $N$  samples for estimating  $\hat{\alpha}$  should be carefully chosen so that the PDF may be more precisely estimated. There may be many options in choosing  $\hat{\alpha}$  when using motion information of the neighboring blocks. For precise estimation, we should adopt as many samples as possible. However, it is not likely that all samples are good for representing the PDF. Taking into account memory requirement and empirical results, we selected a set of four samples.

$$\mathbf{S} = \{\mathbf{s}_1, \mathbf{s}_2, \mathbf{s}_3, \mathbf{s}_4\} = \{MVD_A, MVD_B, MVD_C, MVD_{col}\} \quad (13)$$

where  $MV_A$ ,  $MV_B$ ,  $MV_C$ , and  $MV_{col}$  are motion vectors of the left block (A), the upper block (B), the upper-right block (C), and the collocated block in the previous frame, respectively. If the block C is unavailable, it is replaced with the upper-left block D. Fig. 3 shows the definitions of the neighboring blocks which are in accordance with H.264/AVC.

Since there is little correlation between MVD vectors, we may not ensure that the set  $\mathbf{S}$  is a good one. However, this is not about how to predict the MVD of a current block using the MVDs of neighboring blocks, but how to estimate the distribution of the MVD of the current block. This may be similar to an estimation problem that one estimates the distribution of noise at a time instant. Probably, the samples near the time instant would be selected to estimate the distribution of the noise at the time instant, assuming that the noise is piece-wise stationary. In this sense, the set  $\mathbf{S}$  can be a good candidate. Then, the proposed algorithm is summarized as:

- (1) Set the missing probability  $\epsilon^2$ .
- (2) Compute  $\hat{\mu}$  and  $\hat{k}_{min}$  using Eq. (10) and Eq. (12) if more than three out of four samples are available, or set  $\hat{k}_{min}$  to the original  $SR$  ( $SR_{org}$ ) which is an input parameter to video encoder.
- (3) Obtain the final search range  $k = \min(\max(\hat{k}_{min}, f), SR_{org})$  which is to guarantee searching for at least  $\pm f$ . In our experiments, we set  $f = 3$ .
- (4) Go back to step 2 for the next block.



**Fig. 3.** Definitions of neighboring blocks

## 4 Experimental Results

For performance evaluation, the proposed method was embodied into JM16.2 [16]. Coastguard, Foreman and Stefan with CIF format (352x288), and City, Crew and Soccer with 4CIF format (704x576), were tested for evaluation. We selected the sequences of larger than CIF so that they may contain various motion vectors. The encoding parameters are as follows: number of frames-100, frame rate-30Hz, 16x16, 16x8, 8x16, 8x8 block mode on, RD optimization off, picture types-IPPP, and number of reference frames-1.

For comparison, we measured BDPSNR [17] applying QP=8, 18, 28, 38 without rate control where QP stands for quantization parameter and QP values are selected according to [8]. BDPSNR is the average gain of PSNRs over a few fixed bit-rates against the reference method. For a given QP, the different methods produce different PSNRs and bit-rates, which makes fair comparisons

difficult. For the purpose of the fair comparisons of different methods, it has been developed and widely being adopted in evaluation of image coding systems.

Table 1 shows the performance of the proposed method against the full search algorithm (FSA) on JM16.2. Table 2 provides the details of Table 1. In the tables, the computational complexities are relative to the FSA. For instance, 10% means that the computational complexity of the proposed method is 10% of FSA complexity.

Table 1 demonstrates that the proposed method nearly approaches the FSA in PSNR according to increase of  $\epsilon^2$ , while the complexity is proportional to it. When considering the results, the proposed method gives computational reduction of about 90% for CIF sequences with the search range of 16. In particular, the proposed method is very effective for higher resolution images that require wide search ranges. It prevents computing power from being dissipated in searching for such wide ranges, since the search ranges in the proposed method are adaptively controlled by motion characteristics of input sequences. This is verified by the results in rows where SR=32 in Table 1 and Table 2. The results say that the proposed method gives significant reduction in the computational complexity for large search ranges. The encoders for general purposes cannot help wasting the computing power because they should encompass input sequences with large motions. In this sense, the proposed method is a good candidate to solve such problem. For the same reason, it provides further gain for image sequences with small motions.

**Table 1.** BDPSNR and computational complexity (CPX) of the proposed method against the full search algorithm (JM16.2)

image	SR*	$\epsilon^2 = 0.3$		$\epsilon^2 = 0.2$		$\epsilon^2 = 0.1$		$\epsilon^2 = 0.05$	
		BDPSNR	CPX	BDPSNR	CPX	BDPSNR	CPX	BDPSNR	CPX
coastguard	16	-0.011dB	9.04%	-0.011dB	9.05%	-0.007dB	9.13%	-0.011dB	9.22%
foreman	16	-0.049dB	9.11%	-0.046dB	9.30%	-0.030dB	10.61%	-0.017dB	11.84%
Stefan	16	-0.135dB	9.09%	-0.121dB	9.23%	-0.074dB	10.55%	-0.059dB	12.06%
city	16	-0.012dB	6.77%	-0.015dB	6.79%	-0.012dB	6.98%	-0.012dB	7.23%
	32	-0.020dB	3.51%	-0.023dB	3.51%	-0.020dB	3.60%	-0.016dB	3.72%
soccer	16	-0.128dB	7.50%	-0.107dB	8.32%	-0.064dB	11.62%	-0.047dB	13.53%
	32	-0.120dB	3.69%	-0.105dB	3.98%	-0.060dB	5.94%	-0.038dB	7.37%
crew	16	-0.034dB	9.45%	-0.028dB	12.70%	-0.016dB	23.74%	-0.012dB	29.63%
	32	-0.031dB	4.78%	-0.024dB	6.86%	-0.016dB	15.56%	-0.013dB	20.31%

\*SR = search range value used as an encoding parameter to JM16.2

In the other hand, the proposed method has a potential that it gives a great amount of reduction in complexity when it is combined with the search point sampling based methods. It can be easily combined with them because its search areas are rectangular in shape and all pixels in the areas are available.

**Table 2.** Details of the results of the proposed method and JM16.2 (full search algorithm)

image	SR	QP	JM16.2		$\epsilon^2 = 0.3$		$\epsilon^2 = 0.2$		$\epsilon^2 = 0.1$	
			Bits	PSNR	Bits	PSNR	Bits	PSNR	Bits	PSNR
Coast-guard	16	8	12385.8	53.42	12382.6	53.43	12383.2	53.43	12384.4	53.43
		18	5426.8	44.16	5427.1	44.16	5424.9	44.16	5425.2	44.16
		28	1659.7	35.52	1659.2	35.52	1658.9	35.51	1658.1	35.51
		38	228.8	27.85	231.1	27.83	230.9	27.83	230.0	27.84
Foreman	16	8	9996.7	53.23	10000.6	53.23	10000.0	53.23	9996.6	53.23
		18	2797.0	44.31	2799.4	44.31	2797.6	44.31	2796.1	44.31
		28	509.5	37.32	515.2	37.32	514.5	37.32	511.9	37.32
		38	127.9	31.22	131.9	31.12	132.1	31.12	130.4	31.11
Stefan	16	8	13017.4	53.39	13055.3	53.40	13047.9	53.39	13027.1	53.39
		18	5380.7	44.84	5434.1	44.83	5430.9	44.83	5408.5	44.83
		28	1581.6	36.64	1615.8	36.62	1612.2	36.62	1597.8	36.63
		38	285.0	28.40	296.4	28.34	296.0	28.36	292.5	28.35
City	16	8	48891.5	53.44	48859.9	53.44	48863.0	53.44	48865.3	53.44
		18	19470.7	44.26	19462.7	44.26	19462.6	44.26	19460.4	44.26
		28	2826.8	36.07	2836.4	36.07	2831.6	36.07	2835.2	36.07
		38	402.4	29.28	404.3	29.23	406.8	29.22	403.1	29.22
	32	8	48918.8	53.44	48863.0	53.44	48864.1	53.44	48863.3	53.44
		18	19481.4	44.26	19461.9	44.26	19465.7	44.26	19463.3	44.26
		28	2820.7	36.07	2837.0	36.07	2837.4	36.07	2833.8	36.07
		38	400.9	29.31	405.1	29.22	405.7	29.22	404.6	29.22
Crew	16	8	44619.6	53.37	44637.1	53.37	44622.5	53.37	44608.9	53.37
		18	16937.7	44.66	17023.5	44.66	17001.8	44.66	16968.0	44.66
		28	3176.9	36.70	3264.1	36.71	3246.9	36.71	3213.0	36.70
		38	601.7	30.11	678.2	30.11	666.7	30.11	640.6	30.10
	32	8	44652.8	53.37	44630.1	53.38	44624.5	53.37	44609.5	53.37
		18	16949.1	44.66	17021.0	44.66	17002.0	44.66	16964.5	44.66
		28	3174.3	36.71	3255.3	36.71	3241.5	36.71	3212.7	36.70
		38	602.1	30.10	673.0	30.11	666.9	30.11	636.8	30.09
Soccer	16	8	45184.1	53.37	45064.4	53.37	45066.4	53.37	45105.9	53.37
		18	16348.8	44.84	16331.2	44.84	16326.1	44.84	16321.0	44.84
		28	2827.1	38.21	2867.1	38.21	2858.5	38.21	2845.8	38.21
		38	633.2	32.72	650.5	32.68	648.8	32.68	641.9	32.68
	32	8	45265.8	53.37	45061.7	53.37	45068.7	53.37	45120.0	53.37
		18	16378.4	44.84	16332.5	44.84	16329.7	44.84	16330.2	44.84
		28	2829.7	38.22	2867.3	38.21	2859.4	38.22	2848.0	38.22
		38	633.6	32.73	651.8	32.68	648.4	32.67	643.8	32.68

## 5 Conclusion

We have proposed a new adaptive search range method where the search ranges are constrained by the missing probability of MVDs. The PDF of MVDs was modeled and then its parameter was estimated. Being aware of the PDF, we defined the missing probability as a constraint for restricting the search ranges. With the missing probability, we analytically derived the search ranges to satisfy the constraint. As a result, we have obtained the formula that the search ranges are proportional to the absolute mean of the MVD samples. Then we introduced a new motion estimation method. In the aspect of the parameter estimation, the proposed method considers the MVDs of neighboring blocks as samples for the estimation. Experimental results revealed that the proposed method results in significant reduction in computation, while conserving picture quality close to the optimal method (FSA).

## Acknowledgement

This work was supported by the grant of the Korean Ministry of Education, Science and Technology (The Regional Core Research Program / Chungbuk BIT Research-Oriented University Consortium).

## References

1. ITU-T VCEG and ISO/IEC MPEG, Advanced video coding for generic audiovisual services, ITU-T Recommendation H.264 and ISO/IEC 14496-10 (MPEG-4 AVC) (May 2003)
2. Jain, J., Jain, A.: Displacement measurement and its application in interframe image coding. *IEEE Trans. Communications* 29(12), 1799–1808 (1981)
3. Po, L.-M., Ma, W.-C.: A novel four-step search algorithm for fast block motion estimation. *IEEE Trans. Circuits and Systems for Video Technology* 6(3), 313–317 (1996)
4. Zhu, S., Ma, K.-K.: A new diamond search algorithm for fast block matching motion estimation. *IEEE Trans. Image Processing* 9(2), 287–290 (2000)
5. Cheung, C., Po, L.: A Novel Cross-Diamond Search Algorithm for Fast Block Motion Estimation. *IEEE Trans. Circuits and Systems for Video Technology* 12(12), 1168–1177 (2002)
6. Zhu, C., Lin, X., Chau, L.P.: Hexagon-based search pattern for fast block motion estimation. *IEEE Trans. Circuits and Syst. Video Technol.* 12(5), 349–355 (2002)
7. Chen, Z., Zhou, P., He, Y.: Fast integer and fractional pel motion estimation for JVT, JVT-F017r, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (December 2002)
8. Yi, X., Zhang, J., Ling, N., Shang, W.: Improved and simplified fast motion estimation for JM, JVT-P021, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (July 2005)
9. Hong, M.-C., Oh, H.H.: Range decision for motion estimation of VCEG-N33, JVT-B022, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (February 2002)

10. Hong, M.-C., Kim, C.-W., Seok, K.: Further improvement of motion search range, JVT-D117, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (July 2002)
11. Xu, X., He, Y.: Modification of dynamic search range for JVT, JVT-Q088, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (October 2005)
12. Song, T., Ogata, K., Saito, K., Shimamoto, T.: Adaptive search range motion estimation algorithm for H.264/AVC. In: Proc. of International Symposium on Circuits and Systems, pp. 3956–3959 (2007)
13. Lee, S.W., Park, S.M., Kang, H.S.: Fast motion estimation with adaptive search range adjustment. Optical Engineering 46(4), 040504-1–040504-3 (2007)
14. Chen, Z., Song, Y., Ikenaga, T., Goto, S.: A macroblock level adaptive search range algorithm for variable block size motion estimation in H.264/AVC. In: Proc. of Int. Sym. on Intelligent Signal Processing and Comm. Sys., pp. 598–601 (2007)
15. Lim, K.-P., Sullivan, G., Wiegand, T.: Text description of joint model reference encoding methods and decoding concealment methods, JVT-N046, Joint Video Team (JVT) of ISO/IEC MPEG & ITU-T VCEG (January 2005)
16. H.264/AVC Reference Software JM16.2,  
<http://iphom.hhi.de/suehring/tm1/download/>
17. Andersson, K., Sjoberg, R., Norkin, A.: Reliability measure for BD measurements, ITU-T SG16 Q.6 Document, VCEG-AL22 (July 2009)

# Towards Computational Understanding of Skill Levels in Simulation-Based Surgical Training via Automatic Video Analysis

Qiang Zhang and Baoxin Li

Computer Science & Engineering, Arizona State University,  
Tempe, AZ, 85283

**Abstract.** Analysis of motion expertise is an important problem in many domains including sports and surgery. Recent years, surgical simulation has emerged at the forefront of new technologies for improving the education and training of surgical residents. In simulation-based surgical training, a key task is to rate the performance of the operator, which is done currently by senior surgeons. This is deemed as a costly practice and researchers have been working towards building automated systems to achieve computational understanding of surgical skills, largely through analysis of motion data captured by video or data of other modalities. This paper presents our study on understanding a fundamental issue in building such automated systems: how visual features computed from videos capturing surgical actions may be related to the motion expertise of the operator. Utilizing domain-specific knowledge, we propose algorithms for detecting visual features that support understanding the skill of the operator. A set of video streams captured from resident surgeons in two local hospitals were employed in our analysis. The experiments revealed useful observations on potential correlations between computable visual features and the motion expertise of the subjects, hence leading to insights into how to build automatic system for solving the problem of expertise evaluation.

## 1 Introduction

For centuries, the marvel of human movement has been a source of significant research. Understanding of human motion is especially important in fields such as biomechanics, dance, computer animation, and ergonomics. A key problem that is of significant interest to several domains is the analysis of motion expertise. In domains such as dance, sports, and even surgery, motion of experts differs considerably from novices. This has been the basis of several video based motion analysis systems for sports analysis. However using such systems requires significant amount of planning, calibration, and customized development, rendering it difficult to extend the systems to domains such as surgery wherein medical educationists are not technically trained. The motivation of this paper is to develop computational algorithms that support the development of an intuitive and simple-to-use video-based system in the domain of simulation-based surgical training for motion expertise analysis.

The traditional process of training resident surgeons has been primarily based on interactive and direct instruction of supervising surgeons. Surgical motions have increasingly become complex, requiring significant psychomotor and cognitive skills. As the surgeons have moved from open surgery to laparoscopic surgery to now robotic surgery, the dexterity requirements have grown manifold. This makes surgery education and training even more challenging. And thus conventional training purely relying on a senior surgeon's instruction is not only typically very slow, but also costly and hard to generalize since objectively quantifiable assessment metrics are often missing. Recent years, surgical simulation has emerged at the forefront of new technologies for improving the education and training of surgical residents (e.g., [3] [4] [15] [16]). In theory, data captured from the training simulation can be used to evaluate surgical skills. However, in practice, there has been little substantial advance in this front largely due to the lack of efficient and effective computational approaches to inference of surgical skills from such captured raw data. One challenge is to automatically rate the proficiency level of a resident surgeon ([5]). This factor has currently gained added initiatives as the American college of surgeons seeks to develop a national skills curriculum ([www.acs.org](http://www.acs.org)). The state of the art is the work by Lin et al [8] that segments the captured actions into some pre-defined units and the skill evaluation remains unsolved. For a large part, the challenge is due to the lack of a computational understanding of surgical skills in terms of computable features. This is the problem that our study set out to address.



(a)



(b, c and d)

**Fig. 1.** (a) The setup of the system: an FLS trainer and two external cameras capturing the hand movements. The on-board camera is connected to the display that shows what happens inside the box. (b) One sample frame captured by the on-board camera. (c) and (d) Sample frames from two hand-view cameras.

Specifically, we explore the above problem in the context of simulation-based laparoscopic surgery, which has emerged as an alternative (and in some cases, a replacement) to traditional open surgery. Based on the FLS (Fundamentals of Laparoscopic Surgery) training system [4] (Fig. 1), which has been widely adopted by hospitals in the U.S., we employ two additional cameras to capture the hand motion of the subject. Adding to the video that is captured by the camera of the FLS system, we have a total of three video streams: two capturing the hand movements (i.e. Hand view as shown in Fig. 1c and d) and one capturing the tool movement (i.e. Tool view as shown in Fig. 1b). Our task is to analyze the videos so as to find potential correlation models between the computable visual features and the motion expertise of the subjects. To this end, we first propose a novel algorithm for segmenting the tool-view video into segments that correspond to meaningful physical action units. Then we extract motion features from such segments from videos of the resident surgeons with varying level of surgical skills. This allows us to study the potential correlation between the features and the proficiency of the subject. As we know, this is the first work that tries to evaluate the proficiency level of surgical operation through visual features.

In Section 2, we briefly review relevant studies in the literature. In Section 3, we present the proposed algorithms for segmenting the video and for computing the motion features. Then in Section 4, experimental results and analysis with real videos are reported. Section 5 concludes the paper with a brief discussion on future work.

## 2 Related Work

Objective evaluation of surgical skills has been a topic of research for many years ([6][10][15]). According to [2][18], there is high correlation between the proficiency level and the motion parameters observed, such as duration, number of movements and length of path. This provides the theoretical foundation for building the system for objective evaluation of surgical skills, according to the features collected. A technique proposed by [5] called task deconstruction was implemented in a recent system by [14]. They use Markov Models to model a sequence of force patterns or positions of the tools. Specially designed tools measure hand-tool and tool-tissue interactions through sensors. They showed that their Markov Models were suitable for decomposing a task (such as suturing) into basic gestures, and then the proficiency of the complex gesture could be analyzed. While this study offers an intriguing approach to expertise analysis, it requires an expert surgeon to provide specifications for building the topology of the model; hence it cannot be easily generalized to new procedures.

The work of [8] relies on the use of robot-assisted Minimally Invasive Surgical Systems (MISS), such as Intuitive Surgical's da Vinci, to provide the quantitative motion and video data. In that work, automatic techniques for detecting and segmenting surgical gestures have been developed. Useful features are selected from the original feature set obtained from the da Vinci system; these features are then normalized and projected onto a lower-dimensional feature space with

Linear Discriminant Analysis [13], and a Bayesian classifier is then built to recognize different motion gestures in the lower-dimensional feature space.

The aforementioned systems require special devices, such as data gloves, which require modification of the current system. In addition, wearing data gloves may interfere with the performance of the subjects. For those reasons, a video-based system is preferred, in which one or more cameras are used to capture the movements of the subjects or objects. The key task is then to extract visual features from the videos for proficiency evaluation.

This problem is related to video based action recognition, which has been under research for many years and some good surveys can be found in [17] [11] [12]. However, most of those methods are designed for classifying different actions, whose visual motion is apparently different. On the other hand, in the above problem of proficiency evaluation, the subjects are required to perform the same task and the difference of the visual motion due to their proficiency is subtle. In addition, subjects in the same proficiency level may show very different movements, due to the variation of personal habit. Thus typical action recognition methods do no directly apply for proficiency evaluation.

### 3 Proposed Approach

As discussed in Section 1, we assume that we have 3 cameras capturing the motion of the subject, two capturing the motion of the hands (giving two hand-view videos) and the third capturing the motion of the tool (giving the tool-view video). Our analysis will be based on three videos from these cameras. In this section, we first present our approach for segmenting the tool-view video into segments that correspond to meaningful physical actions of the subject. Then we propose algorithms for extracting features from the tool views. These features will be further used in our analysis in the next Section. Our tool-view-centric approach was motivated by the fact that human experts mainly rely on the tool-view in evaluating the performances of a resident surgeon. In current stage, our approach works offline.

#### 3.1 Segmenting the Tool-View Video

When a subject performs an operation on the FLS system, a task may include several subtasks. For example, in the “peg transferring” operation, the subject needs to first lift an object with a grasper first in one’s non-dominant hand, then transfer the object midair to the dominant hand, and then place the object on a peg on the other side of the board. This process needs to be repeated a few times. Therefore, to facilitate any further analysis of the motion of the subject from the captured video of the tool and the objects movements, it is desired to first segment the tool-view video into much shorter clips that correspond to some underlying actions. However, in the videos captured from existing FLS systems, segmentation of the video based on motion analysis or explicit object tracking is challenging due to a few factors. For example, multiple objects with similar

colors are present in the field of view, which easily confuse a feature tracker that relies on color-based cues. Also, the tool and the object movement will cast shadows and occlusions in the field of view. In addition, the movement of the objects and the tools are three-dimensional and non-rigid in nature. All adding uncertainties to a tracking algorithm.

To get around the above problems, we utilize two other videos capturing the hand movements to achieve the segmentation of the tool-view video. That is, assuming that all the three videos are synchronized, we will attempt to infer the information for motion-based segmentation in the hand-view videos and then use the result to segment the synchronized tool-view video. While in theory, using both hand views may improve the robustness of the approach, in the current study, we experimented with only one hand view for this purpose. Our basic strategy is to compute the motion trajectories from the chosen hand view and then analyze the motion to segment the tool-view video. From the setup presented in Fig. 1, we can make the following assumptions, which simplify the analysis: (i) the subjects wear gloves with known, distinct colors (For simplicity, the gloves region is manually marked out for initialization. But in future work, we can build a color model for the gloves to enable fully automatical initialization); and (ii) the two hands of the subject are separate from each other during the operation. With these assumptions, we propose the following tracking methods for the hand view:

---

**Algorithm 1.** Tracking two hands in the hand view
 

---

Input: A surgical video in the hand view.

Output: Two trajectories  $l$  and  $r$  storing the positions of mass centers of two hands respectively.

Initialize: Build a 3-component Gaussian model for pixels of gloves in HSV color space:  $\mu = \frac{1}{|G|} \sum_{i \in G} x_i$  and  $\sigma = \sqrt{\frac{1}{|G|} \sum_{i \in G} (x_i - \mu)^2}$ , where  $G$  means pixels in gloves area;

1. Convert current frame into HSV space;
2. Calculate the probability of the each pixel classified as gloves pixel with following equation:

$$P(x_{H,S,V}) = p(x_H)p(x_S)p(x_V) \quad (1)$$

$$\text{where } p(x) = \frac{1}{\sqrt{2\pi}\sigma} e^{-\frac{(x-\mu)^2}{2\sigma^2}};$$

3. Binarize the probability by following equation: where  $\mu_p = \frac{1}{|I|} \sum_{i \in I} P(x(i))$  and  $I$  is current frame;
  4. Removing the noises and hole by morphological operation (i.e. erode and dilate);
  5. Clustering the coordinate of pixels where  $\hat{P}(x(i)) = 1$  with K means ( $K = 2$ ). The 2 cluster centers represent the mass centers of the two hands;
  6. Save the 2 cluster centers in  $l$ ,  $r$  respectively and repeat the above steps until the end of the video;
-

With Algorithm 1, we can obtain the trajectory of two hands  $l \in \mathbb{R}^{2*N}$  and  $r \in \mathbb{R}^{2*N}$ , where  $N$  is the number of frames.

In order to remove the effect of translation, without loss of generality, the trajectories are centered to their centroid so that their first moments are zero. The remaining scale normalization is achieved by  $\tilde{x}(:, i) = \frac{x(:, i)}{\|x(:, i)\|}$ , where  $x = [l^T, r^T]^T \in \mathbb{R}^{4*N}$  and  $x(:, i)$  means the  $i$ th column of  $x$ , i.e. the coordinates of mass centers of two hands in  $i$ th frame. After normalization, we calculate a Self Similarity Matrix (SSM) [7]  $M$  with following equations:

$$M(i, j) = \|\tilde{x}(:, i) - \tilde{x}(:, j)\|^2 \quad (2)$$

SSM represents the similarity between action representation (trajectory here) of all pairs of frames.  $M(i, j)$  being small means the positions of two hands in frame  $i$  and  $j$  are similar. Fig. 2a shows the figure of SSM of an example. The region  $S$ , where  $M(i, j)$  is small for all  $(i, j) \in S$  (i.e. the blue regions near the diagonal of Fig. 2a), indicates that the positions in this period is similar, which can be viewed as a segment. So segmentation of video is formulated as detecting these regions. The method is summarized in the following algorithm:

**Algorithm 2.** Segment the hand-view video based on SSM

---

Input: Self Similarity Matrix  $M$ .

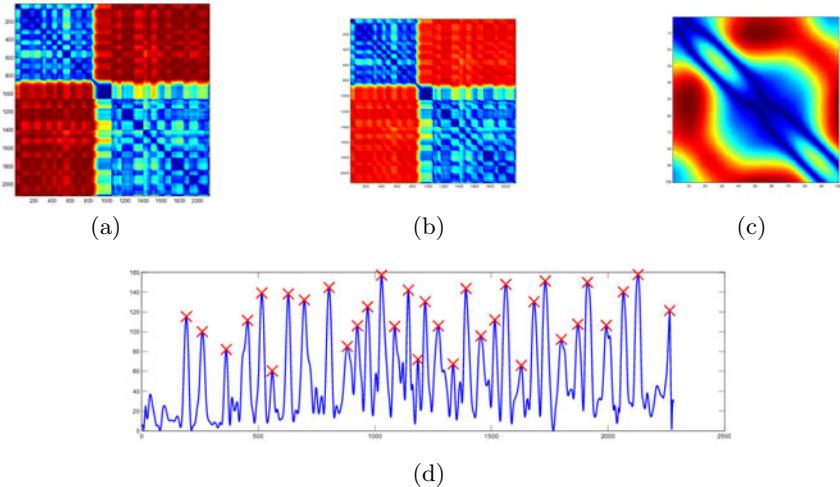
Output: Segmentation positions  $W$ .

1. Build a Gabor filter  $f$  with  $\theta = \frac{3\pi}{4}$ ,  $\sigma_x = \sigma_y = 16$  and  $freq = 33$ ;
2. Calculate  $\hat{M} = M * f$ , where  $*$  is the convolution operation (Fig. 2b and c);
3. Extract the diagonal elements  $d$  from the matrix  $\hat{M}$ ;
4. Thresholding  $x$  by  $\tilde{d}(i) = 0$  if  $d(i) < \bar{d}$ , where  $\bar{d}$  is the mean of  $d$ ;
5. Detecting the positions of local maximal for thresholded  $\tilde{d}$   
 $w = \{i | \tilde{d}(i) - \tilde{d}(i-1) \geq 0 \text{ and } \tilde{d}(i+1) - \tilde{d}(i) \leq 0\}$ ;
6. For each element in  $w$ , if  $w_{j+1} - w_j$  is small, remove the one with smaller value in  $\tilde{d}$ ;
7. The remaining  $w$  are the segmentation positions, which are shown as crosses in Fig. 2(d);

The gabor filter is desgiend to detect the discoutinuity in the diagnoal of SSM.

### 3.2 Feature Extraction in the Tool View

Our hypothesis is that, after segmenting the tool-view video into segments of meaningful action units, it will become easier to analyze the correlation between computed motion features and the motion expertise of the subject, compared with considering the original video on the whole. In this subsection, we describe our method for computing some features for the video segments in the tool view.



**Fig. 2.** (a) SSM for an example. (b) The convolution result of Gabor filter and SSM. (c) Enlarged version of area in highlighted part of (b). The diagonal  $d$  of Self Similarity Matrix  $M$  shown in (a). (d) The red crosses indicate the segmentation positions  $W$  we find with Algorithm 2. This segmentation will be carried over the tool view, assuming the views are synchronized. In our dataset, the segments obtained from the above algorithm are usually 50 to 150 frames long.

We first apply the KLT tracker [9] in each segment to extract the trajectories of points on the tools and the objects. To alleviate the difficulties due to confusing background and shadows etc, we utilize the fact that each segment is relatively short and the motion is relative homogeneous within a segment and propose the following method for computing motion trajectories in the tool view:

---

**Algorithm 3.** Extract trajectories in the segments of tool view

---

Input: A tool-view segment; Number of trajectories  $K$ .

Output:  $K$  trajectories.

For each frame in the video:

1. Convert every frame  $I$  to HSV space and represent the frame as  $\tilde{I} = I_S I_V$ ;
2. Threshold  $\tilde{I}$  by  $\hat{I}(i) = 0$  if  $\tilde{I}(i) <= \bar{I}$ , where  $\bar{I}$  is the mean of  $\tilde{I}$ ;
3. Use morphological operation (erode and dilate) to remove the noise and hole in  $\hat{I}$ ;
4. Calculate the absolute difference  $D$  between the first frame and the last frame and binarize it with

$$\hat{D}(i) = \begin{cases} 1 & \text{if } D(i) \geq \bar{D} \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

where  $\bar{D}$  is the mean of  $D$ ;

5. Detect the feature points with KLT tracker and select those in the region where  $\tilde{D}(i) = 1$ ;
  6. Start KLT tracker with the feature points we selected;
  7. Remove the trajectory whose average motion magnitude is smaller than 1 pixel/frame;
  8. Sort the trajectories according to their length and return the longest  $K$  trajectories, if any;
- 

For every trajectory  $X \in \mathbb{R}^{2*T}$  of a segment, we consider two kinds of feature: the average motion magnitude  $M$  and the jitter  $A$ :

$$M = \frac{1}{T} \sum_i^T |V(:, i)| \text{ and } A = \sqrt{\frac{1}{T} \sum_i^T (V(:, i) - \tilde{V}(:, i))^2} \quad (4)$$

where  $V = \frac{dX(t)}{dt} \in \mathbb{R}^{2*T}$  is the velocity and  $\tilde{V}$  is a smoothed version of  $V$  after moving-average filtering. Finally, each trajectory is represented by a feature vector  $f = [M, A] \in \mathbb{R}^2$ . Each surgical video is represented as a bag of these feature vectors.

## 4 Experiments and Analysis

Videos captured from two local hospitals were used in our experiments to evaluate the segmentation approach and to assess whether the segmentation procedure and the subsequent feature extraction can help generate numeric features that may correlate to the motion expertise of the underlying subject. In the following, we first briefly describe the data used our experiments and then present some analysis results.

We used 12 set of videos for the “peg transfer” operation by 12 different subjects with 2 proficiency levels: 6 of the resident surgeons are deemed as experts who are very skilled with the task while the other 6 are considered as novices who are yet to gain better skills with the task. To give some more detail about the operation, we describe the task below. The peg transfer task requires the subjects to lift (i.e. “Pick”) six objects (one by one) with a grasper by the non-dominant hand, transfer the object midair to the dominant hand, and then place (i.e. “Drop”) the object on a peg on the other side of the board. Once all six objects are transferred, the process is reversed, and the objects are to be transferred back to the original side of the board. The timing for this task starts when the subjects grasped the first object and ends upon the release of the last peg. Some details of the videos used in our experiments, including the segments generated by our algorithm, are given in Tab. □

It was found that the segments generated by the proposed algorithm indeed correspond to some physically-meaningful action units. For example, typically, a segment corresponds to one of the following actions: picking-up, dropping, transferring, or attempts of these actions. Note that, if a subject made several picking

**Table 1.** The detail of the data we used. “duration” means the number of the frames in the video. “Movements” means the number of movements, i.e. segments obtained by Algorithm 2. “comment” indicates the proficiency level of residents, where “E5” means high proficiency level and “N” for low proficiency level. Frame rate is 30 FPS.

ID	1	2	3	4	5	6	7	8	9	10	11	12
Duration	2240	2190	3637	2759	3791	3722	5313	5354	5468	5747	4833	4735
Segments	34	27	39	29	47	29	61	70	62	71	54	48
Comments	E	E	E	E	E	E	N	N	N	N	N	N

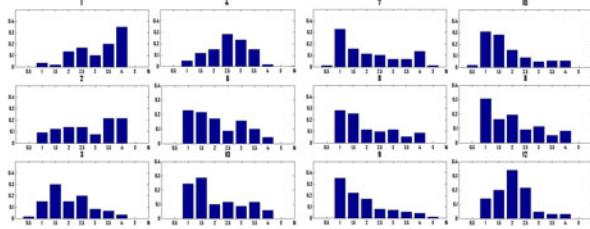
attempt before successfully lifting the object, the video may be segmented into multiple segments. This explains why in the above table the videos from the experts typically contains fewer segments than the novices. Also, an expert may be able to pick up one object while dropping another object simultaneously, hence reducing the number of computed segments.

In this paper, instead of building a classifier for proficieency level based on some features, we analyzed the relationship between the proficiency level and the features we extracted, including: (1) the duration and the number of detection segments which were used in [2][18]; (2) the histogram of average motion magnitude and motion jitter for each type of the segments (as defined above, the segments are manually labeled into “picking-up”, “dropping”, or “transferring”). In the future work, we plan to do it automatically). The definition of the the features are elaborated below:

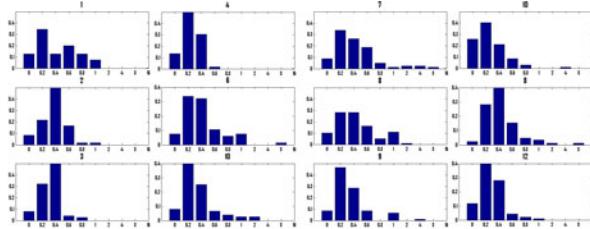
1. Duration: the time to complete the task (2nd column in Tab. I);
2. Number of segments obtained by Algorithm 2 (3rd column in Tab. I);
3. Histogram of average motion magnitude: considering the average motion magnitude M of every trajectory in each type of segments of a video as a data points and compute a histogram. Fig. 4 shows the histograms for the “transferring” segments (i.e. only considering those segments which are determined to be a “transferring” action after visual inspection);
4. Histogram of motion jitterness: considering the jitterness A of every trajectory in each type of segments of a video as a data points and compute a histogram. Fig. 5 shows the histograms for the “picking” action;

From the features computed from the videos, we had the following observations:

1. According to Tab 1, the proficiency level is highly correlated with the duration time and number of the movements that the subjects take to complete the task, which is also verified in [2][18]. For subjects who are novices (i.e. 7 to 12), they take more time and movements, since they would typically need multiple attempts (e.g. taking several tries to pick up an object), and make errors (e.g. losing the objects during transferring) and thus need corrections. Note that, while it may appear to be trivial to find that an expert needs shorter time to complete an action, our intention is for explicitly identifying this fact so that it can be correlated with other feature, hence enabling more profound analysis tasks such as how to provide feedback to a subject



**Fig. 3.** The histogram of average motion magnitude for the “transferring” segments, for each of the 12 videos in Table 1 respectively. The histogram is normalized. The X axis is motion magnitude with the unit pixel/frame.



**Fig. 4.** The histogram of jitterness for the “picking” action, for each of the 12 videos in Table 1 respectively. The histogram is normalized. The X axis is jitterness with unit pixel/frame.

by giving some actionable instructions. For example, telling a subject to act quick may not be as helpful as tell him/her to try to stabilize his/her hands, if we do find a correlation between the expertise and the motion jitterness.

2. According to Fig. 4, subjects at higher proficiency level (i.e. 1 to 6) tends to move faster than subjects who are novice during the “transferring” action, since the histograms of the former group have big masses on the right side of the histogram (i.e. higher average motion magnitude). We can also find intuitive explanation for this observation: The “transferring” action is the most difficult part since most of the time is spent here and errors are most likely to happen during this period. Thus subjects who are novices tend to move more slowly for the “transferring” action;
3. If we divide the histograms in Fig. 5 by 0:8, we can find that the histograms of most subjects at higher proficiency level have less distribution on the right-hand side (i.e. 0:8) than that of novice subjects. This indicates that, subjects at higher proficiency level tend to move more smoothly during the “picking” action. Intuitively, this may be explained by the fact that a novice may be more susceptible to fumbling while trying to pick up the object.

## 5 Conclusion and Future Work

We presented our approach for processing and analyzing videos captured in simulation-based surgical training, based on the widely-adopted FLS platform.

Automatic algorithms were proposed for segmenting the videos and for computing some important motion features from the videos. The algorithms were evaluated with real videos capturing actual movements of resident surgeons in two local hospitals. The analysis on the relationship between the proficiency level and the feature extracted revealed some clear trends which can also find physically meaningful interpretations. The current results were based on 12 videos, and our immediate future work is to expand the analysis to a large data set involving more human subjects. This work contributes to a better computational understanding of motion expertise in the domain of simulation-based surgical training, which will in turn be helpful in building a fully automatic system for evaluating the skills and for providing feedback in such training. This is our long-term future work.

**Acknowledgement.** The authors were partially supported during this work by an NSF grant (Award # 0904778), which is greatly appreciated.

## References

1. Fundamentals of Laparoscopic Surgery
2. Aggarwal, R., Grantcharov, T., Moorthy, K., Milland, T., Papasavvas, P., Dosis, A., Bello, F., Darzi, A.: An evaluation of the feasibility, validity, and reliability of laparoscopic skills assessment in the operating room. *Annals of surgery* 245(6), 992–999 (2007)
3. Aggarwal, R., Undre, S., Moorthy, K., Vincent, C., Darzi, A.: The simulated operating theatre: comprehensive training for surgical teams. *Quality and Safety in Health Care* 13(suppl. 1), i27 (2004)
4. Eversbusch, A., Grantcharov, T.P.: Learning curves and impact of psychomotor training on performance in simulated colonoscopy: a randomized trial using a virtual reality endoscopy trainer. *Surgical endoscopy* 18(10), 1514–1518 (2004)
5. Gallagher, A.G., Ritter, E.M., Champion, H., Higgins, G., Fried, M.P., Moses, G., Smith, C.D., Satava, R.M.: Virtual Reality Simulation for the Operating Room. Proficiency-Based Training as a Paradigm Shift in Surgical Skills Training. *Annals of Surgery* 241, 364–372 (2005)
6. Healey, A.N., Undre, S., Vincent, C.A.: Developing observational measures of performance in surgical teams. *Qual. Saf. Health Care* 13, 33–40 (2004)
7. Junejo, I., Dexter, E., Laptev, I., Pérez, P.: Cross-view action recognition from temporal self-similarities. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008, Part II. LNCS*, vol. 5303, pp. 293–306. Springer, Heidelberg (2008)
8. Lin, H.C., Shafran, I., Yuh, D., Hager, G.D.: Towards automatic skill evaluation: Detection and segmentation of robot-assisted surgical motions. *Computer Aided Surgery* 11(5), 220–230 (2006)
9. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: International Joint Conference on Artificial Intelligence, vol. 3, p. 3. Citeseer (1981)
10. Mayes, S., Deka, J., Kahol, K., Smith, M., Mattox, J., Woodwards, A.: Evaluation Of Cognitive And Psychomotor Skills Of Surgical Residents at Various Stages in Residency. In: 5th Annual Meeting of American College of Obstetricians and Gynecologists (2007)

11. Mitra, S., Acharya, T.: Gesture recognition: A survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews* 37(3), 311–324 (2007)
12. Poppe, R.: A survey on vision-based human action recognition. *Image and Vision Computing* 28(6), 976–990 (2010)
13. Riffenburgh, R.H., Clunies-Ross, C.W.: Linear discriminant analysis. PhD thesis, Virginia Polytechnic Institute (1957)
14. Rosen, J., Brown, J.D., Chang, L., Sinanan, M.N., Hannaford, B.: Generalized approach for modeling minimally invasive surgery as a stochastic process using a discrete markov model. *IEEE Transactions on Biomedical Engineering* 53(3), 399–413 (2006)
15. Satava, R.M., Gallagher, A.G., Pellegrini, C.A.: Surgical competence and surgical proficiency: definitions, taxonomy, and metrics. *Journal of the American College of Surgeons* 196(6), 933–937 (2003)
16. Sutherland, L., Middleton, P., Anthony, A., Hamdorf, J., Cregan, P., Scott, D., Maddern, G.J.: Surgical Simulation: A Systematic Review. *Annals of Surgery* 243, 291–300 (2006)
17. Wang, J.J., Singh, S.: Video analysis of human dynamics—a survey. *Real-Time Imaging* 9(5), 321–346 (2003)
18. Wanzel, K.: Visual-spatial ability correlates with efficiency of hand motion and successful surgical performance. *Surgery* 134(5), 750–757 (2003)

# Biomedical Image Retrieval in a Fuzzy Feature Space with Affine Region Detection and Vector Quantization of a Scale-Invariant Descriptor

Md Mahmudur Rahman, Sameer K. Antani, and George R. Thoma

U.S. National Library of Medicine,  
National Institutes of Health, Bethesda, MD, USA  
`{rahmanmm,santani,gthoma}@mail.nih.gov`

**Abstract.** This paper presents an approach to biomedical image retrieval by detecting affine covariant regions and representing them with an invariant fuzzy feature space. These regions refer to a set of pixels or interest points which change covariantly with a class of transformations, such as affinity. A vector descriptor based on Scale-Invariant Feature Transform (SIFT) computed from the intensity pattern within the region. These features are then vector quantized to build a codebook of keypoints. By mapping the interest points extracted from one image to the keypoints in the codebook, their occurrences are counted and the resulting histogram is called the “bag of keypoints” for that image. Images are finally represented in fuzzy feature space by spreading each region’s membership values through a global fuzzy membership function to all the keypoints in the codebook. The proposed feature extraction and representation scheme is not only invariant to affine transformations but also robust against quantization errors. A systematic evaluation of retrieval results on a heterogeneous medical image collection has shown around 15-20% improvement in precision at different recall levels for the proposed fuzzy feature-based representation when compared to individual color, texture, edge, and keypoint-based features.

## 1 Introduction

The significance of medical imaging is understood and maintaining archives is technically feasible. In recent years, rapid advances of software and hardware technology in medical domain facilitate the generation and storage of large collections of images by hospitals and clinics every day [1]. Medical images of various modalities constitute an important source of anatomical and functional information for the diagnosis of diseases, medical research and education. In a heterogeneous medical collection with multiple modalities, images are often captured with different views, imaging and lighting conditions, similar to the real world photographic images. Distinct body parts that belong to the same modality frequently present great variations in their appearance due to changes in pose, scale, illumination conditions and imaging techniques applied. Ideally, the representation of such images must be flexible enough to cope with a large

variety of visually different instances under the same category or modality, yet keeping the discriminative power between images of different modalities.

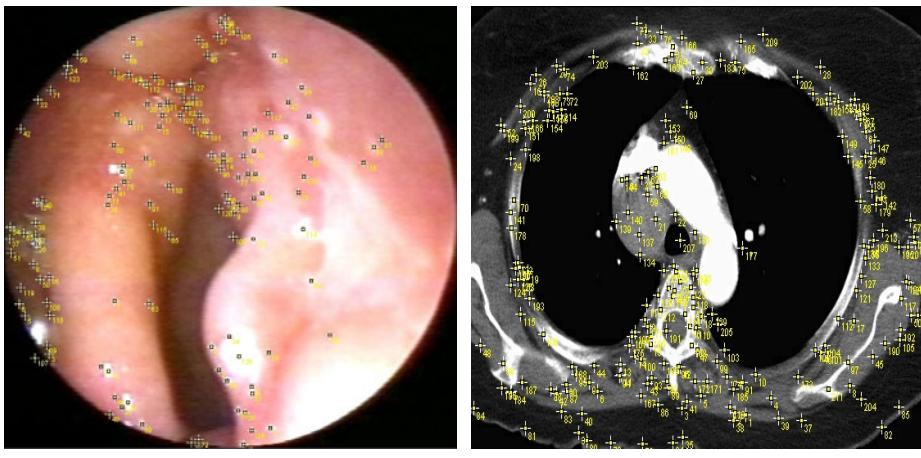
Recent advances in computer vision and pattern recognition techniques have given rise to extract such robust and invariant features from images, commonly termed as affine region detectors [2]. The regions simply refers to a set of pixels or interest points which are invariant to affine transformations, as well as occlusion, lighting and intra-class variations. This differs from classical segmentation since the region boundaries do not have to correspond to changes in image appearance such as color or texture. Often a large number, perhaps hundreds or thousands, of possibly overlapping regions are obtained. A vector descriptor is then associated with each region, computed from the intensity pattern within the region. This descriptor is chosen to be invariant to viewpoint changes and, to some extent, illumination changes, and to discriminate between the regions. The calculated features are clustered or vector quantized (features of interest points are converted into visual words or keypoints) and images are represented by a bag of these quantized features (e.g., bag of keypoints) so that figures are searchable similarly with “bag of words” in text retrieval.

The idea of “bag of keypoints”-based image representation has already been applied to the problem of texture classification and recently for generic visual categorization with promising results [7,8]. For example, the work described in [8] presents a computationally efficient approach which has shown good results for objects and scenes categorization. Besides, being a very generic method, it is able to deal with a great variety of objects and scenes. However, the main limitation of keypoint-based approaches is that the quality of matching or correspondence (i.e., covariant region to keypoints) is not always exact. During the image encoding process, a region in general is classified or matched to a single keypoint only and the rest are simply overlooked or ignored. Hence, the correspondence of an image region to a keypoint is basically “*one-to-one*” due to the nature of hard classification. In reality, there are usually several keypoints with almost as closely match as the one detected for a particular image region. Although, two regions will be considered totally different if they match to different keypoints even though they might be very similar or correlated to each other.

To overcome the above limitation, this paper presents a “bag of keypoints”-based image representation in a fuzzy feature space by applying a soft annotation scheme. In this approach, the SIFT features are extracted at first from the covariant regions and then vector quantized to build a visual vocabulary of keypoints by utilizing a Self-Organizing Map (SOM)-based clustering. The images are presented in a fuzzy feature space by spreading each region’s membership values through a global fuzzy membership function to all the keypoints in the codebook during the encoding and consequent feature extraction process. The organization of the paper is as follows: Section 2 describes the keypoint-based feature representation approach and an image representation scheme is a fuzzy feature space is presented in Section 3. Experiments and analysis of the results are presented in Sections 4 and 5. Finally, Section 6 provides our conclusions.

## 2 “Bag of Keypoints”-Based Feature Representation

A major component of this retrieval framework is the detection of interest points in scale-space, and then determine an elliptical region for each point. Interest points are those points in the image that possess a great amount of information in terms of local signal changes [2]. In this study, the Harris-affine detector is used as interest point detection methods [3]. In this case, scale-selection is based on the Laplacian, and the shape of the elliptical region is determined with the second moment matrix of the intensity gradient [4]. Fig. 1 shows the interest points (cross marks) detected in two images of different modalities from the medical collection.



**Fig. 1.** Images from the medical collection marked (white crosses) with interest points detected by the affine region detector

A vector descriptor which is invariant to viewpoint changes and to some extent, illumination changes is then associated with each interest point, computed from the intensity pattern within the point. We use a local descriptor developed by Lowe [5] based on the Scale-Invariant Feature Transform (SIFT), which transforms the image information in a set of scale-invariant coordinates, related to the local features. SIFT descriptors are multi-image representations of an image neighborhood. They are Gaussian derivatives computed at 8 orientation planes over a  $4 \times 4$  grid of spatial locations, giving a 128-dimension vector. Recently in a study [2] several affine region detectors have been compared for matching and it was found that the SIFT descriptors perform best. SIFT descriptor with affine covariant regions gives region description vectors, which are invariant to affine transformations of the image. A large number of possibly overlapping regions are obtained with the Harris detector. Hence, a subset of the representative region vectors is then selected as a codebook of keypoints by applying a SOM-based clustering algorithm [9].

For each SIFT vector of interest point in an image, the codebook is searched to find the best match keypoint based on a distance measure (generally Euclidean). Based on the encoding scheme, an image  $I_j$  can be represented as a vector of keypoints as

$$\mathbf{f}_j^{\text{KV}} = [\hat{f}_{1j} \cdots \hat{f}_{ij} \cdots \hat{f}_{Nj}]^T \quad (1)$$

where each element  $\hat{f}_{ij}$  represents the normalized frequency of occurrences of the keypoints  $c_i$  appearing in  $I_j$ .

This feature representation captures only a coarse distribution of the keypoints that is analogous to the distribution of quantized color in a global color histogram. As we already mentioned, image representation based on the above hard encoding scheme (e.g., to find only the best keypoints for each region) is very sensitive to quantization error. Two regions in an encoded image will be considered totally different if their corresponding keypoints are different even though they might be very similar or correlated to each other. In the following section, we propose an effective feature representation scheme to overcome the above limitation.

### 3 Image Representation in a Fuzzy Feature Space

There are usually several keypoints in the codebook with almost as good match as the best matching one for a particular covariant region. This scheme considers this fact by spreading each region's membership values through a global fuzzy membership function to all the keypoints in the codebook during the encoding and consequent feature extraction process. The vector  $\mathbf{f}^{\text{Keypoint}}$  is viewed as a keypoint distribution from the probability viewpoint. Given a codebook of size  $N$ , each element  $f_{ij}$  of the vector  $\mathbf{f}_j^{\text{Keypoint}}$  of image  $I_j$  is calculated as  $f_{ij} = l_i/l$ . It is the probability of a region in the image encoded with label  $i$  of keypoint  $c_i \in C$ , and  $l_i$  is the number of regions that map to  $c_i$  and  $l$  is the total number of regions detected in  $I_j$ .

According to the total probability theory [10],  $f_{ij}$  can be defined as follows

$$f_{ij} = \sum_{k_j=1}^l P_{i|k_j} P_k = \frac{1}{l} \sum_{k_j=1}^l P_{i|k_j} \quad (2)$$

where  $P_k$  is the probability of a region selected from image  $I_j$  being the  $k_j$ th region, which is  $1/l$ , and  $P_{i|k_j}$  is the conditional probability of the selected  $k_j$ th region in  $I_j$  maps to the keypoint  $c_i$ . In the context of the keypoint-based vector  $\mathbf{f}^{\text{keypoint}}$ , the value of  $P_{i|k_j}$  is 1 if the  $k_j$ th region is mapped to  $c_i$  or 0 otherwise. Due to the crisp membership value, this feature representation is sensitive to quantization errors.

In such a case, fuzzy set-theoretic techniques can be very useful to solve uncertainty problem in classification tasks [11][14]. This technique assigns an observation (input vector) to more than one class with different degrees instead of a definite class by crisp classification. In traditional two-state classifiers, an

input vector  $\mathbf{x}$  either belongs or does not belong to a given class  $A$ ; thus, the characteristic function is expressed as [11]

$$\mu_A(\mathbf{x}) = \begin{cases} 1 & \text{if } \mathbf{x} \in A \\ 0 & \text{otherwise.} \end{cases}$$

In a fuzzy context, the input vector  $\mathbf{x}$ , belonging to the universe  $X$ , may be assigned a characteristic function value or grade of membership value  $\mu_A(\mathbf{x})$  ( $0 \leq \mu_A(\mathbf{x}) \leq 1$ ) which represents its degree of membership in the fuzzy set  $A$ .

Many methods could be adapted to generate membership from input observations. These include the histogram method, transformation of probability distributions to possibility distributions, and methods based on clustering [11,14]. For example, fuzzy-c-means (FCM) [14] is a popular clustering method, which embeds the generation of fuzzy membership function while clustering. Few schemes have been proposed to generate fuzzy membership functions using SOM [12,13], where the main idea is to augment the input feature vector with the class labeling information. However, without any class label information (as in our case), it might be difficult to generate such fuzzy membership functions. Due to this, we perform a two-step procedure, where in the first step we generate the proper clusters (e.g., keypoints in the codebook) based on the SOM clustering and next the fuzzy membership values are generated according to the generated clusters in the first step as follows [14]:

The membership degree  $\mu_{ik_j}$  of a region vector  $\mathbf{x}_{k_j} \in \Re^d$ ,  $k = 1, 2, \dots, l$ , of the  $k_j$ th region in  $I_j$  to keypoint vectors  $\mathbf{c}_i$ ,  $i = 1, 2, \dots, N$  is:

$$\mu_{ik_j} = \frac{\frac{1}{\|\mathbf{x}_{k_j} - \mathbf{c}_i\|^2}}{\sum_{n=1}^N \frac{1}{\|\mathbf{x}_{k_j} - \mathbf{c}_n\|^2}}^{\frac{2}{m-1}} \quad (3)$$

The higher the distance of an input SIFT vector from a keypoint vector, the lower is its membership value to that keypoint based on (3). It is to be noted that when the distance is zero, the membership value is one (maximum) and when the distance is infinite, the membership value is zero (minimum). The values of  $\mu_{ik_j}$  lies in the interval  $[0, 1]$ . The fuzziness exponent  $\frac{2}{m-1}$  controls the extent or spread of membership shared among the keypoints.

In this approach, during the image encoding process, the fuzzy membership values of each region to all keypoints are computed for an image  $I_j$  based on (3), instead of finding the best matching keypoint only. Based on the fuzzy membership values of each region in  $I_j$ , the *fuzzy keypoint vector* (FKV) is represented as  $\mathbf{f}_j^{\text{FKV}} = [\hat{f}_{1_j}, \dots, \hat{f}_{i_j}, \dots, \hat{f}_{N_j}]^T$ , where

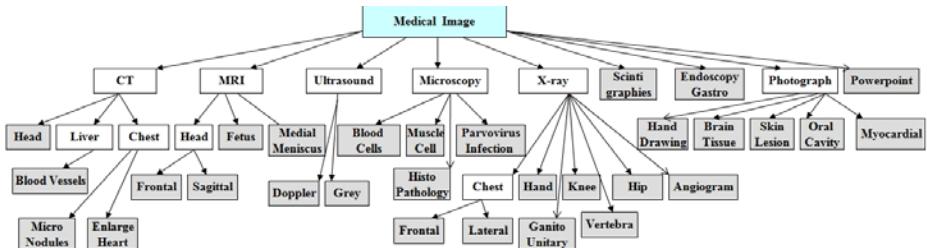
$$\hat{f}_{i_j} = \sum_{k=1}^l \mu_{ik_j} P_k = \frac{1}{l} \sum_{k=1}^l \mu_{ik_j}; \quad \text{for } i = 1, 2, \dots, N \quad (4)$$

The proposed vector essentially modifies probability as follows. Instead of using the probability  $P_{i|k_j}$ , we consider each of the regions in an image being related

to all the keypoints in the codebook based on the fuzzy-set membership function such that the degree of association of the  $k_j$ -th region in  $I_j$  to the keypoint  $c_i$  is determined by distributing the membership degree of the  $\mu_{ik_j}$  to the corresponding index of the vector. In contrast to the keypoint-based vector (e.g.,  $\mathbf{f}^{\text{Keypoint}}$ ), the proposed vector representation (e.g.,  $\mathbf{f}^{\text{FKV}}$ ) considers not only the similarity of different region vectors from different keypoints but also the dissimilarity of those region vectors mapped to the same keypoint in the codebook.

## 4 Experiments

The image collection for experiment comprises of 5000 bio-medical images of 32 manually assigned disjoint global categories, which is a subset of a larger collection of six different data sets used for medical image retrieval task in ImageCLEFmed 2007 [17]. In this collection, images are classified into three levels as shown in Fig. 2. In the first level, images are categorized according to the imaging modalities (e.g., X-ray, CT, MRI, etc.). At the next level, each of the modalities is further classified according to the examined body parts (e.g., head, chest, etc.) and finally it is further classified by orientation (e.g., frontal, sagittal, etc.) or distinct visual observation (e.g. CT liver images with large blood vessels). The disjoint categories are selected only from the leaf nodes (grey in color) to create the ground-truth data set.

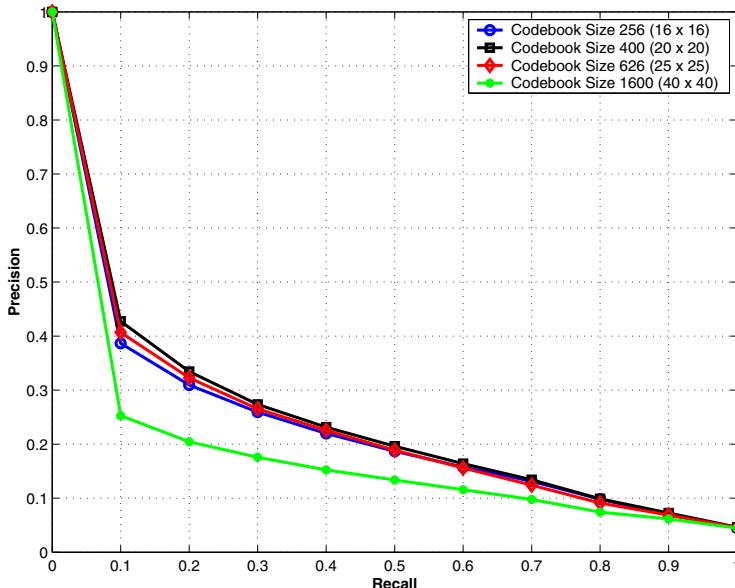


**Fig. 2.** Classification structure of the medical image data set

To build the codebook based on the SOM clustering, a training set of images is selected beforehand for the learning process. The training set used for this purpose consists of 10% images of the entire data set (5000 images) resulting in a total of 500 images. For a quantitative evaluation of the retrieval results, we selected all the images in the collection as query images and used *query-by-example (QBE)* as the search method. A retrieved image is considered a match if it belongs to the same category as the query image out of the 32 disjoint categories at the global level as shown in Fig. 2. Precision (percentage of retrieved images that are also relevant) and recall (percentage of relevant images that are retrieved) are used as the basic evaluation measure of retrieval performances [6]. The average precision and recall are calculated over all the queries to generate the precision-recall (PR) curves in different settings.

## 5 Results

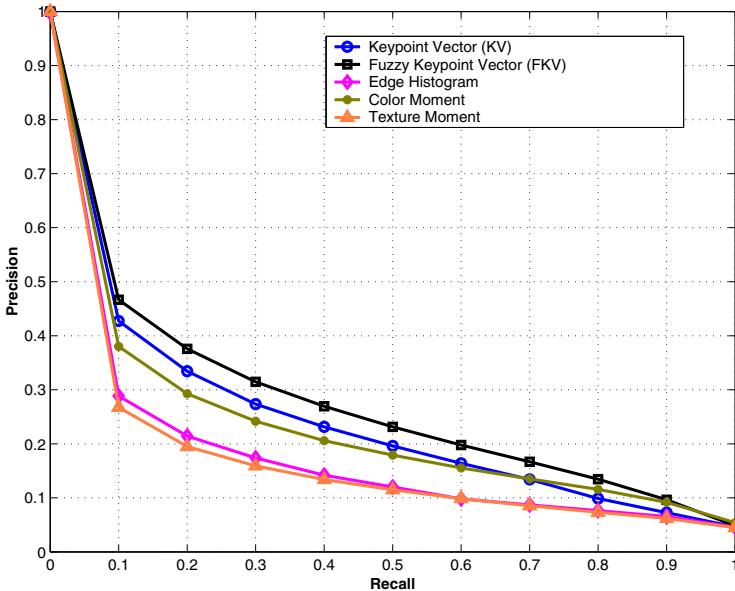
To find an optimal codebook that can provide the best retrieval accuracy in this particular image collection, the SOM is trained at first to generate two-dimensional codebook of four different sizes as 256 ( $16 \times 16$ ), 400 ( $20 \times 20$ ), 625 ( $25 \times 25$ ), and 1600 ( $40 \times 40$ ) units. After the codebook construction process, all the images in the collection are encoded and represented as “bag of keypoints” as described in Section 2. For training of the SOM, we set the initial learning rate as  $\alpha = 0.07$  due to its better performance.



**Fig. 3.** PR-graphs of different codebook sizes

Fig. 3 shows the PR-curves on four different codebook sizes. It is clear from Fig. 3 that the best precision at each recall level is achieved when the codebook size is 400 ( $20 \times 20$ ). The performances are degraded when the sizes are further increased, as a codebook size of 1600 ( $40 \times 40$ ) showed the lowest accuracies among the four different sizes. Hence, we choose a codebook of size 400 for the generation of the proposed keypoints-based feature representation and consequent retrieval evaluation.

Fig. 4 shows the PR-curves of the keypoints-based image representation by performing the Euclidean distance measure in the “bag of keypoints”-based feature space (e.g., “KV”) and the proposed fuzzy keypoints-based feature space (e.g., “FKV’). The performances were also compared to three low-level color, texture, and edge related features to judge the actual improvement in performances of the proposed methods. The reason of choosing these three low-level



**Fig. 4.** PR-graphs of different feature spaces

feature descriptors is that they present different aspects of images. For color feature, the first (mean), second (standard deviation) and third (skewness) central moments of each color channel in the RGB color space are calculated to represent images as a 9-dimensional feature vector. The texture feature is extracted from the gray level co-occurrence matrix (GLCM). A GLCM is defined as a sample of the joint probability density of the gray levels of two pixels separated by a given displacement and angle [15]. We obtained four GLCM for four different orientations (horizontal  $0^\circ$ , vertical  $90^\circ$ , and two diagonals  $45^\circ$  and  $135^\circ$ ). Higher order features, such as energy, maximum probability, entropy, contrast and inverse difference moment are measured based on each GLCM to form a 5-dimensional feature vector and finally obtained a 20-dimensional feature vector by concatenating the feature vector for each GLCM. Finally, to represent the shape feature, a histogram of edge direction is constructed. The edge information contained in the images is processed and generated by using the Canny edge detection (with  $\sigma = 1$ , Gaussian masks of size = 9, low threshold = 1, and high threshold = 255) algorithm [16]. The corresponding edge directions are quantized into 72 bins of  $5^\circ$  each. Scale invariance is achieved by normalizing this histograms with respect to the number of edge points in the image.

By analyzing the Fig. 4, we can observe that the performance of the keypoints-based feature representation (e.g., “KV”) is better when compared to the global color, texture, and edge features in term of precision at each recall level. The better performances are expected as the keypoints-based feature representation is more localized in nature and invariant to viewpoint and illumination changes.

In addition, we can observe that the fuzzy feature-based representation (e.g., “FKV”) approach performed better when compared to the similarity matching in the normalized keypoints-based feature space. In general, we achieved around 15-20% improvement in precision at different recall levels for the fuzzy feature-based representation when compared to individual color, texture, edge, and keypoint-based features. Overall, the improved result justifies the soft annotation scheme by spreading each region’s membership values to all the keypoints in the codebook. Hence, the proposed fuzzy feature representation scheme is not only invariant to affine transformations but also robust against the distribution of the quantized keypoints. For generation of the fuzzy feature, we consider the value of  $m = 2$  of the fuzziness exponent due to its better performance in the ground truth dataset.

## 6 Conclusions

We have investigated the “bag of keypoints” based image retrieval approach in medical domain with a fuzzy annotation scheme. In this approach, images are represented by spreading each region’s membership values through a global fuzzy membership function to all the keypoints in the codebook. The proposed fuzzy feature representation scheme is invariant to affine transformations, as well as occlusion, lighting and intra-class variations and robust against quantization errors. Experimental results with improved precision at different recall levels in a medical image collection justify the validity of the proposed feature representation approach.

## Acknowledgment

This research is supported by the Intramural Research Program of the National Institutes of Health (NIH), National Library of Medicine (NLM), and Lister Hill National Center for Biomedical Communications (LHCNCBC). We would like to thank the ImageCLEFmed [17] organizers for making the database available for the experiments.

## References

1. Müller, H., Michoux, N., Bandon, D., Geissbuhler, A.: A Review of Content-Based Image Retrieval Systems in Medical Applications Clinical Benefits and Future Directions. *International Journal of Medical Informatics* 73(1), 1–23 (2004)
2. Mikolajczyk, K., Tuytelaars, T., Schmid, C., Zisserman, A., Matas, J., Schaffalitzky, F., Kadir, T., Van Gool, L.: A Comparison of Affine Region Detectors. *International Journal of Computer Vision* 65, 43–72 (2005)
3. Mikolajczyk, K., Schmid, C.: An affine invariant interest point detector. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) *ECCV 2002. LNCS*, vol. 2350, pp. 128–142. Springer, Heidelberg (2002)

4. Baumberg, A.: Reliable feature matching across widely separated views. In: Proc. of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 774–781 (2000)
5. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
6. Yates, R.B., Neto, B.R.: Modern Information Retrieval. Addison Wesley, Reading (1999)
7. Lazebnik, S., Schmid, C., Ponce, J.: Sparse texture representation using affine-invariant neighborhoods. In: Proc. International Conference on Computer Vision & Pattern Recognition, pp. 319–324 (2003)
8. Csurka, G., Dance, C., Willamowski, J., Fan, L., Bray, C.: Visual categorization with bags of keypoints. In: Proc. Workshop on Statistical Learning in Computer Vision, pp. 1–22 (2004)
9. Kohonen, T.: Self-Organizing Maps. Springer, New York (1997)
10. Fukunaga, K.: Introduction to Statistical Pattern Recognition, 2nd edn. Academic Press, London (1990)
11. Bezdek, J.C., Pal, S.K.: Fuzzy Models for Pattern Recognition: Methods that Search for Structures in Data. IEEE Press, NY (1992)
12. Mitra, S., Pal, S.K.: Self-organizing neural network as a fuzzy classifier. IEEE Trans. Syst. Man Cybernet. 24(3), 385–399 (1994)
13. Yang, C.C., Bose, N.K.: Generating fuzzy membership function with self-organizing feature map. Pattern Recognition Letters 27(5), 356–365 (2006)
14. Bezdek, J.C., Pal, M.R., Keller, J., Krisnapuram, R.: Fuzzy Models and Algorithms for Pattern Recognition and Image Processing. Kluwer Academic Publishers, Boston (1999)
15. Haralick, R.M., Shanmugam, K., Dinstein, I.: Textural features for image classification. IEEE Trans. System, Man, Cybernetics SMC-3, 610–621 (1973)
16. Canny, J.: A computational approach to edge detection. IEEE Trans. Pattern Anal. Machine Intell. 8, 679–698 (1986)
17. Müller, H., Deselaers, T., Deserno, T.M., Kalpathy-Kramer, J., Kim, E., Hersh, W.: Overview of the ImageCLEFmed 2007 Medical Retrieval and Annotation Tasks. In: Peters, C., Jijkoun, V., Mandl, T., Müller, H., Oard, D.W., Peñas, A., Petras, V., Santos, D. (eds.) CLEF 2007. LNCS, vol. 5152, pp. 472–491. Springer, Heidelberg (2008)

# Model Distribution Dependant Complexity Estimation on Textures

Agustín Mailing<sup>1,2,\*</sup>, Tomás Crivelli<sup>1</sup>, and Bruno Cernuschi-Frías<sup>1,2</sup>

<sup>1</sup> Facultad de Ingeniería, Universidad de Buenos Aires, Argentina

<sup>2</sup> IAM-CONICET, Buenos Aires, Argentina

**Abstract.** On this work a method for the complexity of a textured image to be estimated is presented. The method allow to detect changes on its stationarity by means of the complexity with respect to a given model set (distribution dependant). That detection is done in such a way that also allows to classify textured images according to the whole texture complexity. When different models are used to model data, the more complex model is expected to fit it better because of the higher degree of freedom. Thus, a naturally-arisen penalization on the model complexity is used in a Bayesian context. Here a nested models scheme is used to improve the robustness and efficiency on the implementation. Even when MRF models are used for the sake of clarity, the procedure it is not subject to a particular distribution.

**Keywords:** Complexity, Textured Images, Classification.

## 1 Introduction

One of the most important problems on information theory is the choice of the model order for a given observed data. Lets assume that the observed data comes from an image which has to be classified according to its complexity. The *complexity* of that image depends not only on its nature (satellite image, landscape photograph, etc. ) but also on the context we are working. That is, it can be thought in terms of the number of human-recognizable things present in the image as well as the width of its spectrum, etc., ie. the complexity measure differs in problems according to which aspects we are interested to measure. Owing to this work is aimed to texture analysis, we will think the data complexity as the complexity of its probabilistical description. The fact that textures are often quite similar to noise implies that the complexity in the sense of Kolmogorov or Rissanen [1] is always high because it is considered (almost) random and its description length is as large as the data sequence itself. In our probabilistic scheme instead, a random iid. sequence is very simple to be described statistically.

Then, a set of models with different complexities is selected and that which fits better the observed data (image) is chosen. Here a problem arise : in general, the

---

\* The authors would thanks to Universidad de Buenos Aires, CONICET Argentina, and ANPCyT(PICT 209-2006) Argentina.

most complex model in the set will fit better the data. Thus, in order to preserve the main principle, a penalty over the model complexity has to be introduced in such a way that models with less complexity can be chosen but avoiding artifacts and inconsistency.

Some examples of these penalties are present in Rissanen's MDL [2], the Akaike information criterion (AIC) [3] or the Bayes information criterion (BIC) [4].

We use an extension to the results of [5] and [6] to non-independent Gaussian Markov Random Fields where the penalty over the model complexity is not arbitrarily imposed. Moreover, the same scheme can be used with more complicated distributions instead of the simple GMRF used here for the sake of clarity.

Unlike [7] a nested models set is used here in a particular way, allowing the whole texture to be considered as the data set instead of clustering before an independent classification. This nested model set has some other noticeable advantages that will be remarked below.

## 2 Complexity in a Bayesian Context

Being  $\omega_1, \omega_2, \dots, \omega_c$  a set of  $c$  classes and  $\mathbf{X} = \{x_p\}_1^n$  an image of  $n$  pixels the Bayesian decision rule for equal prior probabilities classes is [8],

$$\mathbf{X} \in \omega_k \quad \text{if} \quad p(\mathbf{X} | \omega_k) \geq p(\mathbf{X} | \omega_l) \quad \forall l \neq k. \quad (1)$$

Under the assumption that for each class  $\omega_k$  the model is completely defined by its parameter vector  $\boldsymbol{\alpha}^{(k)} = [\alpha_1^{(k)}, \alpha_2^{(k)}, \dots, \alpha_{d_k}^{(k)}] \in \Xi^{(k)}$  of dimension  $d_k$  the likelihood can be written as

$$p(\mathbf{X} | \omega_k) = \int_{\Xi^{(k)}} p(\mathbf{X} | \boldsymbol{\alpha}^{(k)}, \omega_k) p(\boldsymbol{\alpha}^{(k)} | \omega_k) d\boldsymbol{\alpha}^{(k)}. \quad (2)$$

Let  $\hat{\boldsymbol{\alpha}}^{(k)}$  the ML estimate of  $\boldsymbol{\alpha}^{(k)}$  for the class  $\omega_k$  using  $n$  points. The log-likelihood

$$L_k = \log p(\mathbf{X} | \boldsymbol{\alpha}^{(k)}, \omega_k) \equiv \log p(\mathbf{X} | \hat{\boldsymbol{\alpha}}^{(k)}) \quad (3)$$

in Taylor expansion series around the ML estimate  $\hat{\boldsymbol{\alpha}}^{(k)}$  is

$$\begin{aligned} \log p(\mathbf{X} | \boldsymbol{\alpha}^{(k)}) &= \log p(\mathbf{X} | \hat{\boldsymbol{\alpha}}^{(k)}) \\ &- \frac{1}{2} (\boldsymbol{\alpha}^{(k)} - \hat{\boldsymbol{\alpha}}^{(k)})^T \mathbf{H}(\hat{\boldsymbol{\alpha}}^{(k)}) (\boldsymbol{\alpha}^{(k)} - \hat{\boldsymbol{\alpha}}^{(k)}) + \dots, \end{aligned} \quad (4)$$

where  $-\mathbf{H}(\hat{\boldsymbol{\alpha}}^{(k)})$  is the  $d_k \times d_k$  Hessian matrix w.r.t the parameter vector  $\boldsymbol{\alpha}^{(k)}$ . That means that the  $i, j$ -element is

$$H_{ij} = -\frac{\partial^2}{\partial \alpha_i^{(k)} \partial \alpha_j^{(k)}} \log p(\mathbf{X} | \boldsymbol{\alpha}^{(k)})|_{\hat{\boldsymbol{\alpha}}^{(k)}}. \quad (5)$$

Because the expansion is done around the MLE the first order term disappears by definition. In addition, the higher order terms can mostly be ignored on images because as  $n$  increases  $p(\mathbf{X} | \boldsymbol{\alpha}^{(k)})$  becomes asymptotically Gaussian [6] whenever the model is correctly specified and  $\mathbf{H}(\cdot)$  positive definite.

$$p(\mathbf{X} | \boldsymbol{\alpha}^{(k)}) \approx p(\mathbf{X} | \hat{\boldsymbol{\alpha}}^{(k)}) e^{-\frac{1}{2}(\boldsymbol{\alpha}^{(k)} - \hat{\boldsymbol{\alpha}}^{(k)})^T \mathbf{H}(\boldsymbol{\alpha}^{(k)} - \hat{\boldsymbol{\alpha}}^{(k)})} \quad (6)$$

On the other hand, as usual [8] for large values of  $n$  the likelihood does concentrate near to the ML value given the estimation will converge to the true parameter vector [5]. Thus,

$$\begin{aligned} p(\mathbf{X} | \boldsymbol{\alpha}^{(k)}) &\approx \\ p(\mathbf{X} | \hat{\boldsymbol{\alpha}}^{(k)}) (2\pi)^{d_k/2} \det \{\mathbf{H}\}^{-1/2} \delta(\boldsymbol{\alpha}^{(k)} - \hat{\boldsymbol{\alpha}}^{(k)}), \end{aligned} \quad (7)$$

where  $\delta$  is the Dirac delta. Now, using the identity  $\det \mathbf{H} = \det \left\{ \frac{n\mathbf{H}}{n} \right\} = n^{d_k} \det \left\{ \frac{\mathbf{H}}{n} \right\}$ , replacing in (2) and integrating,

$$L_k \approx \log p(\mathbf{X} | \hat{\boldsymbol{\alpha}}^{(k)}) - \frac{d_k}{2} \log n / 2\pi - \frac{1}{2} \log \det \frac{\mathbf{H}}{n}. \quad (8)$$

where the term  $\log p(\hat{\boldsymbol{\alpha}}^{(k)} | \omega_k)$  was deprecated for large  $n$  in comparison to the other terms.

At this point, it is important to note in equation (8) that  $d_k$  has appeared giving an explicit relation of  $L_k$  to the number of parameters  $d_k$ , in other words, an explicit penalization on the *complexity* of the model in the likelihood calculation.

## 2.1 Complexity Penalization for GMRF Distribution

We will consider a MRF distribution of the form  $p(\mathbf{X}; \boldsymbol{\alpha}) = \exp Q(\mathbf{X}; \boldsymbol{\alpha}) / Z(\boldsymbol{\alpha})$  where  $Q(\cdot)$  is a linear function of the parameters  $\boldsymbol{\alpha}$ , in particular a GMRF [9]. With this constraint, the most cases of interest are contemplated, and

$$\frac{\partial^2}{\partial \alpha_i \partial \alpha_j} Q(\cdot) = 0. \quad (9)$$

Thus, avoiding to write the subscript  $k$  for the sake of clarity we get

$$H_{ij} = -\frac{\partial^2}{\partial \alpha_i \partial \alpha_j} \log p(\mathbf{X} | \boldsymbol{\alpha})|_{\hat{\boldsymbol{\alpha}}} = \frac{\partial^2}{\partial \alpha_i \partial \alpha_j} \log Z(\boldsymbol{\alpha})|_{\hat{\boldsymbol{\alpha}}}. \quad (10)$$

Can be noticed that  $\mathbf{H}$  approaches the Fisher [10] information matrix (which can also be found for example in AIC) by observe that the  $i, j$ -element of the FIM is  $-E[\frac{\partial^2}{\partial \alpha_i \partial \alpha_j} \log p(\mathbf{X} | \boldsymbol{\alpha})]|_{\boldsymbol{\alpha}^{true}}$  given that  $\hat{\boldsymbol{\alpha}} \rightarrow \boldsymbol{\alpha}^{true}$ .

Finally we want to obtain the log-likelihood (8) for a GMRF defined by

$$Q(\mathbf{X}; \boldsymbol{\alpha}) = \sum_p a x_p - b x_p^2 + \sum_{\{p,q\}} h_{p,q} x_p x_q, \quad (11)$$

where the pairs  $\{p, q\}$  are neighbouring pixels in the image. Here an eight neighbourhood is considered with interaction in four directions (namely  $\{h_{vert}, h_{horiz}, h_{diag}, h_{anti-diag}\}$ ).

Given that the correlation between neighbours in a Gibbs distributed image decays exponentially with distance, can be shown for the evaluation of  $\log \det \left\{ \frac{\mathbf{H}}{n} \right\}$  that each element of  $\mathbf{H}$  it is of order  $O(n)$ . Then  $\log \det \left\{ \frac{\mathbf{H}}{n} \right\}$  becomes of order  $O(1)$  and so depreciable compared to the other terms (see [7] for details).

In a future work, it is planned to observe the role of  $\mathbf{H}$  when other distributions [6] are used and that term does not become depreciable. At last, the log-likelihood remains,

$$L_k \approx \log p(\mathbf{X}_k | \hat{\boldsymbol{\alpha}}^{(k)}) - (d_k/2) \log n/2\pi. \quad (12)$$

With this equation  $L_k$  can be approximately calculated for a Gaussian MRF-distributed image. Note that the same approximation is valid for any neighbourhood, whenever its size is small compared to the whole image. As the term  $(d_k/2) \log n/2\pi$  does only depend on  $d_k$  it results useless to the likelihood test if the classes have the same number of parameters, provided that the parameters were estimated with the same number of samples. On the other hand, it is worth noting that without this term comparing the likelihood for two classes with different number of parameters becomes unfair, because the more parameters a model has, the better it fits the data. This means that in a ML scheme the more *complex* model will always be chosen. Hereafter, we call this term the *complexity term*.

### 3 Nested Models and Complexity for GMRF Textures

In order to estimate the complexity of a GMRF texture the following window scheme is proposed. The whole textured image is partitioned so that each partition can be also partitioned in the same way, and thus giving a simple nested model set (Fig. 1a). In that way if the  $N$  partitions over the window  $\mathbf{X}_{(k)}$  are considered independent its (partitioned) likelihood  $L_k^\#$  becomes,

$$L_k^\# = \sum_{m=1}^N \{\log p(\mathbf{X}_{k,m} | \hat{\boldsymbol{\alpha}}^{(k,m)}) - (d_k/2) \log n^{(k,m)}/2\pi\}, \quad (13)$$

and thus,

$$L_k^\# = \sum_{m=1}^N L_{k,m}. \quad (14)$$

Each sub-window (thinking of the whole texture as a sub-window as well) is modelled by some distribution associated with a likelihood value. The both likelihood values  $L_k$  and  $L_k^\#$ , corresponding to a partitioned and non-partitioned

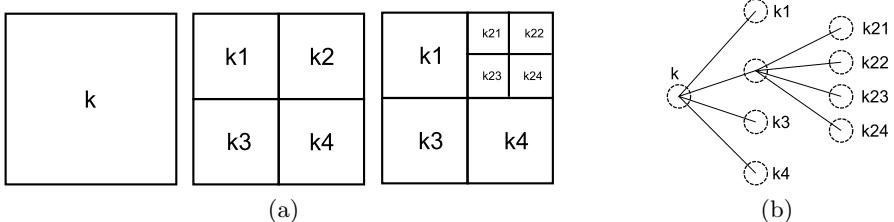
windows respectively, can now be compared and then decide whether to partition it or not. Here it is important to note that in  $L_k$  only one distribution is considered for the whole set  $\mathbf{X}_k$  instead of the  $N$  distributions taken into account for  $L_k^\#$ . Also, as we will consider only one distribution at time, ie. GMRF with fixed number of parameters, the  $d_k$  is constant for  $k$ , and so  $d_k = d$ .

For example, if no further partitions are made over any of the  $N$  partitions  $((k, 1), (k, 2), (k, \dots), (k, N))$ , the comparison for window  $k$  becomes,

$$\begin{aligned} L_k &\leq L_k^\#, \\ L_k &\leq \sum_{m=1}^N L_{k,m}, \\ L_k &\leq \sum_{m=1}^N \{\log p(\mathbf{X}^{k,m} | \hat{\boldsymbol{\alpha}}^{(k,m)}) - (d/2) \log n^{(k,m)} / 2\pi\}, \\ L_k &\leq \sum_{m=1}^N \log p(\mathbf{X}^{k,m} | \hat{\boldsymbol{\alpha}}^{(k,m)}) - \sum_{m=1}^N (d/2) \log n^{(k,m)} / 2\pi, \end{aligned} \quad (15)$$

where in  $L_k$  we will have  $n_k = \sum_{m=1}^N n^{(k,m)}$ .

As can be noticed, the first term of (15) is a classical likelihood provided that  $\mathbf{X}_{k,m}$  are independent, while the second term penalizes the increase on the number of parameters as a consequence of the partition of  $\mathbf{X}_k$ . As follows, without the second term the decision will be ever to partition  $\mathbf{X}_k$  because *non-partitioned* (ie.  $L_k$ ) is just a particular partition (ie.  $L_k^\#$ ).



**Fig. 1.** (a) Three possible partitions for the  $k$ -th sub-window ( $\mathbf{X}_k$ ). To the right the complexity increases (1,4 and 7). (b) The tree for the right-hand model of (a).

In addition, as the sub-windows are taken independently and the models nested, the same scheme can also be applied in the same fashion to every sub-windows at any level of the partition. That is, the process of calculating the likelihood can be thought as a tree (Fig. 1b) with  $N$  branches, where the more ramification implies more complexity and the leaves are the smallest partition admissible. In the actual implementation, that partition depends on the parameter estimation algorithm: the more efficient is the algorithm a finer partition can be done. This process can be recursively done by means of the following simple algorithm,

```

function get_likelihood(  $\mathbf{X}_k$  : sub-window ) : scalar;
     $L_k = 0;$ 

     $L_k = \text{likelihood\_estimator}(\mathbf{X}_k);$ 

    if  $\mathbf{X}_k$  is not partitionable
        return  $L_k$  ;
    end

     $L_k^\# = 0;$ 
    for  $m = 1$  to  $N$ 
         $L_k^\# = L_k^\# + \text{get\_likelihood}( \mathbf{X}_{k,m} )$  ;
    end

    if  $L_k^\# \geq L_k$ 
        return  $L_k^\#$  ;
    else
        return  $L_k$ ;
    end
end function

```

Note that in a similar fashion the complexity can be calculated as well.

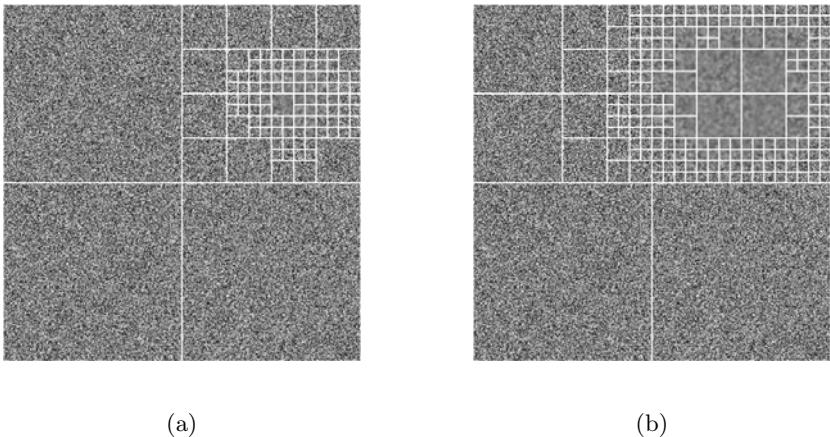
### 3.1 Experimental Results

According to the model used in section 2.1, a six parameter ( $d = 6$ ) GMRF model is assumed and a pseudo-maximum-likelihood estimator is used to estimate its parameters. This is done in that way because the calculation of the partition function  $Z(\boldsymbol{\alpha}^{(k)}) = \int_{\mathbf{X}_k} \exp Q(\mathbf{X}_k; \boldsymbol{\alpha}^{(k)}) d\mathbf{X}_k$  is intractable through a classical estimator. In order to calculate  $\log p(\mathbf{X}_k | \hat{\boldsymbol{\alpha}}^{(k)}) = Q(\mathbf{X}_k; \hat{\boldsymbol{\alpha}}^{(k)}) - \log Z(\hat{\boldsymbol{\alpha}}^{(k)})$  the value of the second term can be approximated by numerical simulation [11] or by integration of the series expansion of  $\exp Q(\mathbf{X}_k; \hat{\boldsymbol{\alpha}}^{(k)})$ : here we use the latter.

Note that the complexity measure as well as the way the image is partitioned will be strongly dependant on the probability distribution model chosen.

In the comparison of figures 2a and 2b can be noticed that two different textures are present, with essentially different degree of correlation between pixels. In figure 2a the region of different statistic concentrates the most of the complexity. When the different region increases in its size (from 2a to 2b) the region of more complexity concentrates in the *border* between the textures because a more complex model is necessary to model the transition.

The figure 3 shows that sometimes images that are not distributed according to the modeling distribution can even offer interesting results on its complexity estimation (w.r.t. GMRF distribution).



**Fig. 2.** When the statistics change in the texture (a) to the texture (b) the complexity does it concentrate at the region of change (ie. texture border)



**Fig. 3.** The application of the method in a non-GMRF distributed textured image but with interesting results

The nested models scheme allows a considerable simplification in the likelihood calculation owing to the *decision stage* is at each branch level (see the algorithm). This means that for each  $k$  once  $L_k$  is compared to  $L_k^\#$  a decision can be taken, thus, simplifying the computation of  $L$  and the complexity for the whole image.

## 4 Conclusion

In this work an application is shown where the complexity of a textured image is estimated according to a particular distribution. Here, for the sake of

clarity, a Gaussian MRF distribution is used for calculus and experimental results. However, several distributions can be also an election, and more than one model can be used at the same time in the likelihood comparison, although some re-calculation would be possibly necessary.

The proposed method allows to compute the complexity of the overall texture, allowing the complexity (w.r.t a distribution) between different textured images to be compared and thus to take an action. Moreover, as can be noticed (see the experimental results) the complexity can be estimated also in a local fashion since more complex regions (possibly borders between different textures) are deeper in the tree.

Even when the task of detecting non-uniform regions in textures can sometimes be done through wavelets, the proposed method is worthy of consideration because it achieves a more probabilistic insight given a distribution and a model. Moreover, provided the distributions are properly chosen and valid, the image texture can be statistically the same (at least similar) as one obtained through simulation methods using the previously estimated parameters. That is, the textured image can be statistically compressed by saving the tree(Fig. [IB](#)) and the corresponding parameters.

A future work is to completely understand how the FIM matrix influences the complexity estimation when the election of the distribution does not allow its neglection. Also it would be important to compensate the drawback caused by the less consistence on the estimation on the leaves compared to the major branches.

One of the most important aspects is that even when some approximation were done, the penalization term is not artificially imposed, so that the method has more theoretical insight implying the more understanding can be achieved, and possibly the richer will be any future work.

## References

1. Li, M., Vitanyi, P.: An introduction to Kolmogorov complexity and its applications. Springer, Heidelberg (1993)
2. Rissanen, J.: Hypothesis selection and testing by the MDL principle. *The Computer Journal* (1999)
3. Akaike, H.: A new look at the statistical model identification. *IEEE Trans. on Automatic Control* 19, 716–723 (1974)
4. Schwartz, G.: Estimating the dimension of a model. *Annals of Statistics* 6, 461–464 (1978)
5. Bolle, R., Cooper, D.: Bayesian recognition of local 3D shape by approximating image intensity functions with quadric polynomials. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 6, 418–429 (1984)
6. Kashyap, R.: Optimal choice of AR and MA parts in autoregressive moving average models. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 4, 99–104 (1982)
7. Crivelli, T., Mailing, A., Cernuschi-Friás, B.: Complexity-based border detection for textured images. In: *IEEE International Conference on Acoustics, Speech and Signal Processing* (2010)

8. Duda, R.O., Hart, P.E., Stork, D.G.: *Pattern Classification*, 2nd edn. Wiley-Interscience, Hoboken (2000)
9. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. *Journal of the Royal Statistical Society. Series B* 36, 192–236 (1974)
10. Cover, T., Thomas, J.: *Elements of Information Theory*. Wiley-Interscience, Hoboken (2006)
11. Potamianos, G., Goutsias, J.: Stochastic approximation algorithms for partition function estimation of gibbs random fields. *IEEE Trans. on Information Theory* 43, 1948–1965 (1997)

# Integrating Multiple Uncalibrated Views for Human 3D Pose Estimation

Zibin Wang and Ronald Chung

Department of Mechanical and Automation Engineering  
The Chinese University of Hong Kong  
`{zbowang, rchung}@mae.cuhk.edu.hk`

**Abstract.** We address the problem of how human pose in 3D can be estimated from video data. The use of multiple views has the potential of tackling self-occlusion of the human subject in any particular view, as well as of estimating the human pose more precisely. We propose a scheme of allowing multiple views to be put together naturally for determining human pose, allowing hypotheses of the body parts in each view to be pruned away efficiently through consistency check over all the views. The scheme relates the different views through a linear combination-like expression of all the image data, which captures the rigidity of the human subject in 3D. The scheme does not require thorough calibration of the cameras themselves nor the camera inter-geometry. A formulation is also introduced that expresses the multi-view scheme, as well as other constraints, in the pose estimation problem. A belief propagation approach is used to reach a final human pose under the formulation. Experimental results on in-house captured image data as well as publicly available benchmark datasets are shown to illustrate the performance of the system.

## 1 Introduction

The ability of determining a full human-body pose in space from video data opens the door to understanding human actions, and has applications ranging from visual surveillance, to motion capture in the film industry, and to human-computer interaction. However, the problem is arguably one of the most difficult ones: full body kinematics is of high dimensionality, limb depths are ambiguous, people can put on various clothing to impose various appearances, and there are often self-occlusions to any single viewpoint. Although there have been works in the literature on the subject, in the form of single-view based [3, 4, 5, 8, 12] and multi-view based [17, 18, 19, 20] ones, a robust solution is yet to be developed. Obviously, the use of multiple views could enhance robustness of the solution toward uncertainties, as more data are available about the same pose. It also has the potential of allowing self-occlusions to be handled better, especially if the available views are from significantly different viewpoints. This work aims at developing a framework that can extend single-view solution to a multi-view one in a natural way. Specifically, we introduce a simple mechanism of embracing multiple views' data for 3D pose determination, which requires no sophisticated calibration of the cameras ahead of time.

Developments of single-view based human detection [7, 9] and pose estimation algorithms [8] have shown considerable success on video retrieval and visual surveillance. In some examples, multiple persons could be detected, and 3D human pose could be recovered from a single video stream [3]. However, in general, single view methods suffer severely from the problem of self-occlusions, especially upon those human poses in which the limbs are not fully stretched out.

Multi-view solutions could address the problem of self-occlusions better, but then the presence of multiple streams of data also introduce new difficulty – that of how the multiple video streams could be associated to one another for the goal of determining the same 3D pose. Existing works on multi-view systems [18, 19, 20] require fixed cameras and pre-known intrinsic and extrinsic camera parameters to establish correspondences between views. By means of the inter-camera geometry (which in the case of two cameras can be expressed as a matrix termed the fundamental matrix, and in the case of three cameras a tensor called the trifocal tensor), different video streams can be related. A few multi-view systems that use a large number of cameras in a controlled environment could obtain reliable results and have reached commercial maturity [21]. However, such systems have difficulty in being extended to uncontrolled environments where the cameras could be arbitrarily positioned and moved around in accordance with need. This work aims at coming up with a multi-view solution that does not require the cameras to be fixed and well calibrated beforehand.

On the other hand, the application domain varies, and so does the demanded accuracy in pose estimation. While outdoor surveillance requires to track multiple people in only coarse accuracy [9], the film industry could employ more than 10 cameras in a controlled environment to capture every minute detail of body movements [20]. In human computer interaction, which is the application domain this work assumes, intermediate precision but fast processing is demanded. In this particular context, we can adopt the affine camera model to describe each camera. By so doing, as to be described in this report, the image projection process of each view and the data association between views can be much simplified.

## 2 Related Work

There is a wide range of approaches to human pose estimation [1, 2]. Most of the recent approaches assume a tree structure for the human body articulation. These algorithms could be categorized to two: the top-down approach that looks for an entire person in a single frame [3, 4, 6, 12], and the bottom-up approach that looks for candidate body parts in the image data and find the assemblies of them that might be persons [5, 7, 9, 10].

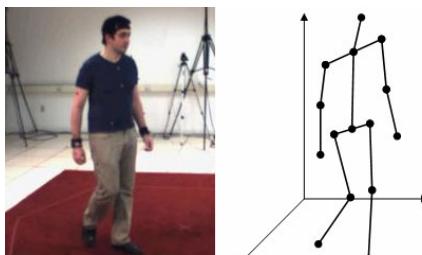
The top-down approach generally makes use of an overall optimization scheme to minimize the overall matching error between the model and the image data. Sminchisescu and Triggs [12] have investigated the application of stochastic sampling to estimate 3D pose from monocular image sequences. Lee et al. [3] combined a probabilistic proposal map representing pose likelihoods with a 3D model to recover 3D pose from a single image. Other inference approaches include Data Driven Belief Propagation [4] and particle filter [6]. Bergtholdt et al. provide a general paradigm for estimating the human pose in images using complete graph [16]. The bottom-up

approaches usually combine human body parts under constraints related to the body kinematic articulation. One merit of the bottom up approach is that by detecting human parts separately, the complexity of the problem can be much reduced. Felzenszwalb et al. proposed a pictorial structures framework that uses linear time to solve for the best pose configuration [5]. This framework is further applied and extended to 2D pose estimation [7] [8] [9] and 3D pose estimation [10] [11]. Another approach, commonly known as the discriminative methods [13] [14] [15], is to learn pose configurations from training data and find the human pose by appearance based features. Since it is considerably difficult to resolve the inherent ambiguity in monocular human motion, monocular estimation of complex 3D human movement remains an open problem.

Multiple cameras can provide more views that help solve depth ambiguity and the occlusion problem. Oliver Bernier et al. locate the 3D position of the hands and faces by utilizing the depth map of the upper limb and the skin color, and infer the upper-body pose in 3D using a belief network [17]. Gupta et al. proposed a part based approach that incorporates a variety of constraints including multiple views, occluded image evidence and appearance symmetry [18]. These works however require some restrictive assumptions, such as the possibility of removing the background or pre-knowing the cameras' parameters. Another approach [19] for estimating the 3D human pose from multiple cameras ( $>10$ ) is based on segmenting the visual hull based on prior knowledge of the shapes of the body parts, their relative sizes, and the possible configurations. S. Corazza et al. have aligned a 3D visual hull body model to an observed visual hull constructed from silhouettes, and randomly search the limbs [20]. Although the approach has reached commercial maturity [21], they can only be used for studio-like applications since they require static backgrounds and tight fitting clothes, and are too sensitive to background subtraction errors.

### 3 Human Model and Problem Formulation

We use a human model that could be described as a puppet of 10 elliptical cylinders each representing a body part. The body parts are interconnected to one another through a system of 12 skeletons (which include the shoulder and hip skeletons), as illustrated in Fig. 1. The skeletons help define which two body parts are connected through a common end-point.



**Fig. 1.** The 3D human model we use in our method: the connections between body parts are defined by a system of 12 skeletons

A graph is used to express the body articulation. Body parts are represented by a collection of nodes  $\mathbf{X} = \{\mathbf{X}_i : i = 1, 2, \dots, 10\}$ , in which  $\mathbf{X}_i = (x_i, y_i, z_i, \alpha_i, \beta_i, \gamma_i, s_i)$  corresponds to the desired results for the  $i$ th body part, where  $x_i, y_i, z_i$  denote its 3D position,  $\alpha_i, \beta_i, \gamma_i$  denote its orientation in space in the roll-pitch-yaw angles, and  $s_i$  denotes the scale of the body. We denote the set of observations in the image space of the 10 body parts as  $\mathbf{Z} = \{\mathbf{Z}_i, i = 1, 2, \dots, 10\}$ . The joint posterior distribution that relates  $\mathbf{Z}$  to  $\mathbf{X}$  under the graph is:

$$P(\mathbf{X} | \mathbf{Z}) \propto \prod_{(i,j) \in \mathcal{E}} \phi_{ij}(\mathbf{X}_i, \mathbf{X}_j) \prod_{i \in v} \phi_i(\mathbf{Z}_i | \mathbf{X}_i, A_i) \quad (1)$$

In the above,  $\phi_{ij}$  is the connection constraint between node  $i$  and node  $j$ , which is modeled as the Euclidean distance between their connected joint points;  $\phi_i$  is the likelihood function for node  $i$ ;  $A_i$  is the 3D appearance of the part;  $v$  is the set of indices of the nodes in the above graph; and  $\mathcal{E}$  is the set of index-pairs of all the connected nodes. This way, the pose estimation problem is formulated as a Bayesian inference problem of estimating the marginal posterior distribution  $P(\mathbf{X} | \mathbf{Z})$ . Note that in the multi-view scenario, the observation  $\mathbf{Z}$  refers to the integrated information of all the views.

## 4 Multiple Image Stream Association

Here we describe a simple mechanism of allowing multiple image streams of the same human subject, each captured by a distinct camera at an arbitrary viewpoint, to be put together in a simple way to let the human pose be estimated more robustly and accurately.

Affine camera is a description of the image projection process of a camera when the overall depth of the imaged object (relative to the camera) is substantially larger than the thickness of the object (in the direction of the camera). With the description, image projection of the object is about a bundle of parallel projection rays from all points of the object. In our intended application, it is configured that the entire body of the human subject is visible in all cameras. Thus the object distances to all cameras are substantial, and affine camera will be an adequate model to describe each camera.

Consider a network of  $N$  cameras, with the first camera (Camera 1) designated as the reference camera, and its coordinate frame  $X$ - $Y$ - $Z$  as the reference coordinate frame for all 3D positions in the imaged space. Under the affine camera model, any point  $\mathbf{P} = [X, Y, Z]^T$  in 3D will project to the reference camera as:

$$\begin{bmatrix} x_1 \\ y_1 \end{bmatrix} = \mathbf{J}_1 \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (2)$$

where  $\mathbf{J}_1$  is the projection matrix, which is a  $2 \times 4$  matrix, of Camera 1.

For any other camera, say the  $i$ th camera, the projected point of  $\mathbf{P}$  is:

$$\begin{bmatrix} x_i \\ y_i \end{bmatrix} = \mathbf{G}_i \begin{bmatrix} \mathbf{R}_i & \mathbf{t}_i \\ 0_{1 \times 3} & 1 \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{J}_i \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (3)$$

where  $\mathbf{G}_i$  is the projection matrix of Camera  $i$ , and  $(\mathbf{R}_i, \mathbf{t}_i)$  represent the rotation and translation transformation of camera  $i$  with respect to the reference camera.

Equations (2) and (3) illustrate that the image projection of any 3D point in space to any camera can be captured by merely a  $2 \times 4$  matrix  $\mathbf{J}_i$ . Putting together the image observations of the same object point in the first  $(N-1)$  cameras, we have:

$$\begin{bmatrix} x_1 & y_1 & x_2 & y_2 & \dots & \dots & x_{N-1} & y_{N-1} \end{bmatrix}^T = \begin{bmatrix} \mathbf{J}_1 \\ \mathbf{J}_2 \\ \dots \\ \mathbf{J}_{N-1} \end{bmatrix} \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \quad (4)$$

where the first  $2(N-1) \times 1$  matrix on the left represents all image observations and is referred to as the observation matrix  $\mathbf{O}_{1:N-1}$ , and the  $2(N-1) \times 4$  matrix on the right represents the intrinsic and extrinsic parameters of all cameras and is referred to as the projection matrix  $\mathbf{J}_{1:N-1}$ .

With (4), the projection of the same 3D point  $\mathbf{P}$  to the remaining camera –  $N$ th camera – could be expressed as:

$$\begin{bmatrix} x_N \\ y_N \end{bmatrix} = \mathbf{J}_N \cdot \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} = \mathbf{J}_N \cdot (\mathbf{J}_{1:N-1}^{-1} \cdot \mathbf{O}_{1:N-1}) = (\mathbf{J}_N \cdot \mathbf{J}_{1:N-1}^{-1}) \cdot \mathbf{O}_{1:N-1} = \mathbf{T} \cdot \mathbf{O}_{1:N-1} \quad (N \geq 3) \quad (5)$$

where  $\mathbf{J}_{1:N-1}^{-1}$  is the pseudo-inverse of the projection matrix, and the  $2 \times 2(N-1)$  matrix  $\mathbf{T} = \mathbf{J}_N \cdot \mathbf{J}_{1:N-1}^{-1}$  captures the information of all the  $N$  cameras parameters. The existence of the pseudo-inverse matrix  $\mathbf{J}_{1:N-1}^{-1}$  requires that  $N \geq 3$ .

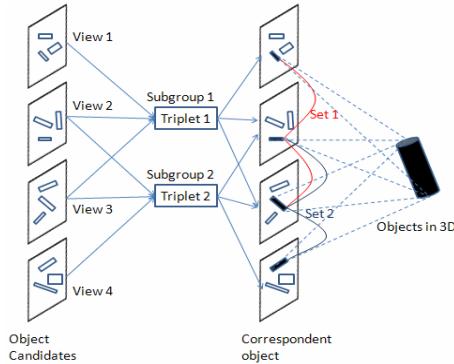
If the matrix  $\mathbf{T}$  is rewritten in the form of  $\mathbf{T} = [\mathbf{T}_1 \ \mathbf{T}_2 \ \dots \ \mathbf{T}_i \ \dots \ \mathbf{T}_{N-1}]$ , where each  $\mathbf{T}_i$  is a  $2 \times 2$  matrix, (5) could be reformulated as:

$$\begin{bmatrix} x_N \\ y_N \end{bmatrix} = [\mathbf{T}_1 \ \mathbf{T}_2 \ \dots \ \mathbf{T}_i \ \dots \ \mathbf{T}_{N-1}] \cdot \begin{bmatrix} (x_1, y_1)^T \\ (x_2, y_2)^T \\ \dots \\ (x_{N-1}, y_{N-1})^T \end{bmatrix} = \sum_{i=1}^{N-1} \mathbf{T}_i \begin{bmatrix} x_i \\ y_i \end{bmatrix} \quad (N \geq 3) \quad (6)$$

From (6), any 3D point  $\mathbf{P}$ 's the projection to the  $N$ th view could be represented as a linear combination-like expression of its projections to other (at least 2) views, with the combination coefficients being not scalars but be  $2 \times 2$  matrices. We refer to these  $\mathbf{T}_i$ 's as the coefficient matrices. In other words, the point-to-point correspondences

over different views could be simply established by means of a linear combination-like expression as long as the coefficient matrices  $\mathbf{T}_i$  are known.

Equation (6) represents a convenient way of associating all views through rigidity of the imaged object in 3D. However, involving all  $N$  views in (6) for each 3D point might not be computationally efficient. In our work, to gain computational efficiency we divide all  $N$  views into triplets of views, i.e., sets of 3, and apply (6) to each triplet to associate the three views in it. That way, for each triplet,  $N$  is 3, and the above pseudo-inverse becomes exact inverse. Different triplets, say triplets of Views {1,2,3} and Views {2,3,4}, are then related by the common views (in this case Views 2 and 3) they have, as illustrated by Fig. 2. The underlying meaning of (6) is that, while two views specify 3D information, three views bear certain redundancy and must exhibit certain rigidity connection among them.



**Fig. 2.** Data Association over different triplets of views: correspondences between different triplets of views (say triplets of Views {1,2,3} and Views {2,3,4}) could be established across the views that the triplets share in common (Views 2 and 3)

## 5 A Bottom-up Framework upon Multiple Views

A bottom-up framework is used to estimate the human pose using multiple views. The bottom-up approach first looks, in each distinct view, for possible candidates for each body part. In the multi-view scenario, after finding part candidates in each view respectively, correspondences of such body part hypotheses would be established across different views before their being assembled to whole-body candidates. Since each part candidate is represented by two end points, we just use the two end points in each view to associate the corresponding body part candidates over different views. We apply the two end points, one set from each view, to Equation (6) described in Section 4. Only those body parts that have support from all views by satisfying (6) would stay in the filtering process. One merit of this step is that a large amount of false part candidates hypothesized in each distinct view can be pruned away, allowing the inference process to operate in a more efficient manner.

### 5.1 Part Detection and Data Association

We use the boundary evidence and appearance of a body part as the features for body part detection, same as in [11]. The image patches  $Patch_i^t = \{x_i^t, y_i^t, \mathbf{p}_i^t, l_i^t\}$  in a view that receive much edgel support and with similar color in the part appearance are detected as candidates in parallel with those in all other views, where  $x_i^t, y_i^t, \mathbf{p}_i^t$  and  $l_i^t$  denote the x, y coordinates, part orientation, and part length of the  $i^{th}$  body part in the view  $t$  respectively. Such part candidates are then associated together as part correspondences using (6).

If each set of the above corresponding part candidates are considered as coming from a real 3D part in space, the image evidence of that 3D part is the integration of the image evidence in all views. We set the image evidence of the 3D part in (1) as:

$$\phi_i(\mathbf{Z}_i | \mathbf{X}_i, \mathbf{A}_i) = \prod_{t=1}^N \phi_t(\mathbf{Z}_i^t | \bar{x}_j^t) \quad (7)$$

where  $\bar{x}_j^t = x_j^t, y_j^t, \mathbf{p}_j^t, l_j^t$  represents the 2D part configuration in view  $t$  for part  $i$ .

### 5.2 Articulation Constraint

The articulation constraint of the human body is modeled in our system as a soft constraint between hypothesized body parts in each view, in the sense that body parts that should be neighbors in a human body should have common end-points in both 3D and all image projections. We use a Gaussian distribution to model the Euclidean distance between the joint points of any two parts that should be neighbors. Each 3D connection could be expressed as 2D connections in  $N$  views, as shown in Fig.3. The articulation constraint in (1) can be expressed as:

$$\phi_{jk}(X_j, X_k) \propto F_{jk}(X_j, X_k, S) = \prod_{t=1}^N f_{jk}(\bar{x}_j^t, \bar{x}_k^t, S) = \prod_{t=1}^N \exp\left(\frac{-(j_{jk}(\bar{x}_j^t, S) - j_{kj}(\bar{x}_k^t, S))^2}{2\sigma_t^2}\right) \quad (8)$$

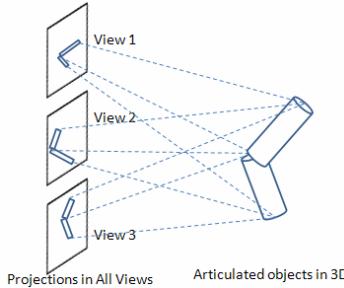
where  $S$  is the body scale,  $\sigma_t$  is the parameter that defines the tightness of the connection between every two adjacent body parts in view  $t$ ,  $\bar{x}_j^t, \bar{x}_k^t$  are the 2D part configuration in view  $t$  for part  $j$  and  $k$ , and  $j_{kj}(x_k, S)$  is the joint of body part  $k$  that is supposed to be connected to part  $j$ .

### 5.3 Put Together All Terms and the Belief Propagation Process

Expressing (1) further by the use of (7) and (8), we can put together all constraints as:

$$P(X | Z) \propto \prod_{(j,k) \in \epsilon} \prod_{t=1}^N \exp\left(\frac{-(j_{jk}(\bar{x}_j^t, S) - j_{kj}(\bar{x}_k^t, S))^2}{2\sigma_t^2}\right) \times \prod_{k \in v} \prod_{t=1}^N \phi_k(Z_k^t | \bar{x}_k^t) \quad (9)$$

where  $\bar{x}_i^t$  are 2D candidate configuration after data association.



**Fig. 3.** Articulation Constraint: body parts that are neighbors in a human body should share common end-points in both 3D and each view

A standard factor graph belief propagation algorithm (sum-product) is applied to infer the human pose. We then extend the inference approach of [8] to multiple views by substituting the body part candidates in one view with body part correspondences in multiple views. Since the part dependencies are modeled after Gaussian distributions, the expensive summations necessary in the propagation can be efficiently computed using Gaussian convolutions.

## 6 Experiments

In the experiments, we tested the proposed multi-view data association scheme on hand labeled images captured ourselves, and evaluated the whole human pose estimation algorithm using the publicly available benchmark dataset “Human Eva datasets” [22], which provides synchronized images and motion capture data.

To evaluate the influence of the angle between views and the view numbers to the *data association scheme*, we applied the system to images taken around a table with littered objects on it, as shown in Fig.4. The pictures were taken around the table at an imaging distance of 2 meters and in an angle step of  $15^\circ$ , the resolution of the images being  $660 \times 1000$ . 10 cylindrical objects, each with their two end points labeled, were in the scene. We pre-calibrated the coefficient matrices with four corresponding points across all views and set the point error as:

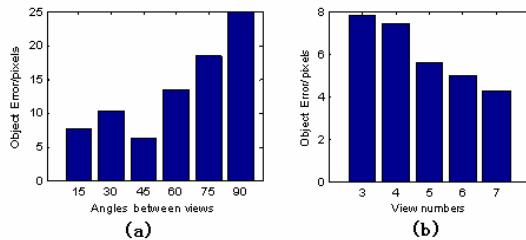
$$\text{Error} = D\left(\begin{bmatrix} x_N \\ y_N \end{bmatrix}, \sum_i^{N-1} \mathbf{T}_i \begin{bmatrix} x_i \\ y_i \end{bmatrix}\right) \quad (N \geq 3) \quad (10)$$

where  $\mathbf{T}_i$  are the coefficient matrices and  $D(\cdot)$  is the Euclidean distance. For an object in the views, the average error of the two end-points is set to be the object error. In evaluating the influence of angles between views, three cameras were used; and when evaluating the influence of view numbers, the angle between views was set to be  $30^\circ$ . The evaluation of the various view angles and view numbers is shown in Fig. 5.

Fig.5 (a) suggests that the proper angle between views should be kept under  $45^\circ$ , or else the error will soar. Fig. 5(b) shows that the object error decreases when more



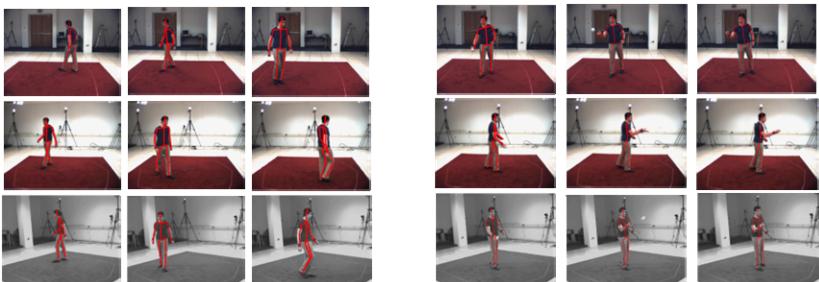
**Fig. 4.** A set of cylindrical objects imaged from different angles



**Fig. 5.** Evaluation results of Multi-view Object Filter. (a) The object error plot against different view angles. (b) The object error plot against different view numbers.

views are added. This conclusion illustrates the success of the data association scheme: by a simple linear combination-like relation, more views could be added into the system and improve the performance.

We then use the Human Eva benchmark datasets, without the use of any background subtraction, to evaluate the whole human pose estimation system. Instead of using the pre-calibrated camera parameters supplied by the datasets, the system used 4 correspondences over the views to pre-determine the coefficient matrices in the linear-combination like expression that related all the views. Some of the image data used are shown Fig. 6. The sequences taken by the cameras C1, C2 and C3 are colored images while the sequences taken by BW1, BW2, BW3 and BW4 are gray level images. For the gray level images, part templates were used to reduce the searching space by locating the person in a smaller window. We used subject S2 in C1, C2,



**Fig. 6.** Estimation results on the Human Eva benchmark datasets, using 3 cameras at a time. The first row is C1 images, the second row is C2 images and the last row is the BW4 images; the first 3 columns are from ThrowCatch\_1 and the last 3 columns are from Walking\_2.

BW4 to evaluate the algorithm when N=3, and added BW3 into the system when N=4. Fig.6 shows the estimation results with 3 cameras, and Table 1 shows the quantitative performance comparison between one of the latest developments [10] and our method, upon 3 and 4 views. Since the view numbers used are different, we just list the errors of C1 and C2 of our results.

In Table 1, our results could reach similar performance as [10] even though we used simpler and coarser part detectors as compared with that in [10]. In the future work we will embed the trained part detector of [10] into our system for more elaborate evaluation. Another observation is that the errors decreased when more view was added to the system, as anticipated.

**Table 1.** Quantitative performance evaluation on the Human Eva dataset: the mean error and the standard deviation of the relative 2Doint positions of different views in pixels are reported

Subj./Cam.	2D Mean (std) [10]	2D Mean (std) / 3 views	2D Mean (std) / 4views
S2/C1	10.49 (2.70)	14.74 (3.62)	13.22 (3.47)
S2/C2	10.72 (2.44)	14.81 (3.86)	13.46 (3.32)

## 7 Conclusion

We describe a multi-view system that is capable of putting together a number of constraints for determining human pose. A novel element of the system is a scheme that allows image data from multiple views to be working together in reaching a consistent human pose interpretation. The scheme takes the form of a linear combination-like expression over the image data in the views. By the use of the scheme, body part hypotheses in each view needs the support of other views to not be pruned away. Even though in hypothesizing body part candidates from each view we inevitably employed a particular mechanism, the multi-view scheme does not bear much dependence on the single-view part extraction mechanism; it is a general one that can be used with other single-view mechanisms. Experimental results with our own data and with publicly available benchmark data show that, as expected, the use of more views generally improve the performance, as long as the views are not too far apart to bear too little overlap in what they picture (and thus there is a meaning in their working together). We also show how the multi-view scheme can be integrated naturally to a multi-constraint formulation framework, and how a simple probability inference method can be used with the formulation to arrive at a human pose interpretation. The system works well on complex environment even without background subtraction.

## Acknowledgment

This work is affiliated with the Microsoft-CUHK Joint Laboratory for Human-centric Computing and Interface Technologies. The work described was partially supported by the Chinese University of Hong Kong 2009-2010 Direct Grant (Project No. 2050468).

## References

1. Moeslund, T., Granum, E.: A Survey of Computer Vision-Based Human Motion Capture. *Computer vision and Image Understanding* 81(3), 231–268 (2001)
2. Moeslund, T., Hilton, A., Krüger, V.: A survey of advances in vision-based human motion capture and analysis. *Computer vision and Image Understanding* 103, 90–126 (2006)
3. Lee, M.W., Cohen, I.: Proposal maps driven mcmc for estimating human body pose in static images. In: CVPR, vol. 2, pp. 334–341 (2004)
4. Hua, G., Yang, M.H., Wu, Y.: Learning to estimate human pose with data driven belief propagation. In: CVPR, vol. 2, pp. 747–754 (2005)
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Pictorial structures for object recognition. *IJCV* 61(1), 55–79 (2005)
6. Peursum, P., Venkatesh, S., West, G.: A Study on Smoothing for Particle-Filtered 3D Human Body Tracking. *IJCV* 87(1-2) (2010)
7. Ramanan, D., Forsyth, D., Zisserman, A.: Tracking People by Learning Their Appearance. *TPAMI* 29(1), 65–81 (2007)
8. Andriluka, M., Roth, S., Schiele, B.: Pictorial Structures Revisited: People Detection and Articulated Pose Estimation. In: CVPR (2009)
9. Ferrari, V., Marin, M., Zisserman, A.: Progressive search space reduction for human pose estimation. In: CVPR (2008)
10. Andriluka, M., Roth, S., Schiele, B.: Monocular 3D Pose Estimation and Tracking by Detection. In: CVPR (2010)
11. Wang, Z., Chung, R.: Articulated Human Body: 3D Pose Estimation using a Single Camera. In: ICPR (2010)
12. Sminchisescu, C., Triggs, B.: Estimating articulated human motion with covariance scaled sampling. *International Journal of Robotics Research* 22(6), 371–391 (2003)
13. Agarwal, A., Triggs, B.: 3D human pose from silhouettes by relevance vector regression. In: CVPR, pp. 882–888 (2004)
14. Elgammal, A., Lee, C.: Inferring 3d body pose from silhouettes using activity manifold learning. In: CVPR, pp. 681–688 (2004)
15. Dalal, N., Triggs, B.: Histogram of oriented gradients for human detection. In: CVPR (2005)
16. Bergtholdt, M., Kappes, J., Schmidt, S., Schnörr, C.: A Study of Parts-Based Object Class Detection Using Complete Graphs complete graph. *IJCV* 87(1-2) (2010)
17. Bernier, O., Cheung-Mon-Chan, P., Bouguet, A.: Fast nonparametric belief propagation for real-time stereo articulated body tracking. *Computer Vision and Image Understanding* 113, 29–47 (2009)
18. Gupta, A., Mittal, A., Davis, L.S.: Constraint Integration for Efficient Multiview Pose Estimation with Self-Occlusions. *TPAMI* 30(3) (2008)
19. Cheung, K.M., Baker, S., Kanade, T.: Shape-from-silhouette of articulated objects and its use for human body kinematics estimation and motion capture. In: CVPR, vol. 1, pp. 77–84 (2003)
20. Corazza, S., Mundermann, L., Gambaretto, E., Ferrigno, G., Andriacchi, T.: Markerless motion capture through visual hull, articulated ICP and subject specific model generation. *IJCV* 87(1-2) (2010)
21. Sigal, L., Black, M.J.: Guest Editorial: State of Art in Image- and Video-Based Human Pose and Motion Estimation. *IJCV* 87(1-3) (2010)
22. Sigal, L., Black, M.J.: Humaneva: Synchronized video and motion capture dataset for evaluation of articulated human motion. In: Techniacl Report CS-06-08, Brown University (2006)

# A Novel Histogram-Based Feature Representation and Its Application in Sport Players Classification

Paolo Spagnolo, Pier Luigi Mazzeo, Marco Leo, and Tiziana D’Orazio

Istituto di Studi sui Sistemi Intelligenti per l’Automazione, C.N.R.

Via G. Amendola 122/D 70126 Bari, Italy

{spagnolo,mazzeo,leo,dorazio}@ba.issia.cnr.it

<http://www.issia.cnr.it/>

**Abstract.** Automatic sport team discrimination, that is the correct assignment of each player to the relative team, is a fundamental step in high level sport video sequences analysis applications. In this work we propose a novel set of features based on a variation of classic color histograms called Positional Histograms: these features try to overcome the main drawbacks of classic histograms, first of all the weakness of any kind of relation between spectral and spatial contents of the image. The basic idea is to extract histograms as a function of the position of points in the image, with the goal of maintaining a relationship between the color distribution and the position: this is necessary because often the actors in a play field dress in a similar way, with just a different distribution of the same colors across the silhouettes. Further, different unsupervised classifiers and different feature sets are jointly evaluated with the goal of investigate toward the feasibility of unsupervised techniques in sport video analysis.

## 1 Introduction

In recent years sport applications of computer vision have been increasing in many contexts, such as tennis, football, golf, and so on. In particular, many works focus on football applications, since it is one among the most popular team sports around the world, and it has a large audience on television. The research activities in sports video have focused mainly on semantic annotation [1], event detection [2], generic content structure analysis [3] and summarization [4]. The high level applications mentioned above are based on structural low level procedures: the player segmentation [5], tracking [6],[7] and their classification [8],[9],[10].

In this work we focus our attention mostly on the last aspect of sport image analysis: the automatic classification of players according to their team membership. In particular our goal is to detect the most effective feature set that allows us to correctly distinguish the different classes of actors involved in a team sport match, reducing the interaction of human beings and making the whole system less dependent from particular match conditions (for example the a-priori knowledge about the team uniforms).

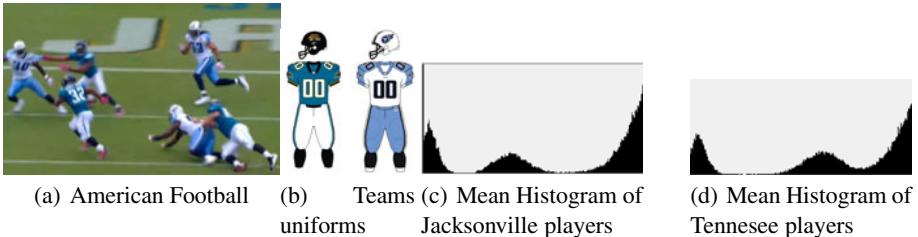
In the recent years several works have been published on this task. In [11] each player is characterized by a series of points; they are compared with points relative to each other player with the goal to agglomerate players with same color information. The color space they use is HSI. It appears to work well even if many static thresholds

have been used for the discrimination. In [6] a supervised approach has been used for the player classification. Each class of them is manually initialized and then, for each player provided by the segmentation step, its distance from the mean value of each class is evaluated: a minimum distance decision rule is used to assign each player to the relative class. A similar supervised approach is presented in [7]: color histograms are used for players and referees characterization. In [12] SVM is used to assign each segmented object to one of 5 relevant classes manually initialized. In [10] information about the local position of each player in the field is integrated to make the classification more reliable. A recent interesting work has been proposed in [13]: the authors have realized a system able to analyze broadcast images, even in the presence of camera motion, in order to detect objects that differ from a previously learned model of the field. The a priori knowledge is the basis also of the player classification procedure.

All the above works try to solve the problem of player team discrimination in a supervised way, by means of human-machine interactions for the creation of the reference classes. On the contrary, in this work we investigate the usability of unsupervised algorithms for the automatic generation of the class models (players and referee). To do it, classic feature representations based on histograms usually fail because the global information provided by an histograms can be unable to correctly distinguish actors in particular in presence of players dressed in similar colors, but physically positioned in a different way. This aspect seems to be rare or irrelevant, but our experience demonstrate that it happens in a great number of matches: typical examples are the players uniforms in American football players, where colors of pants and shirts are often exchanged for opposite teams, or the goalkeepers in football matches that often dress in a similar way with players of the opposite team or officials. In fig. 1 an example of a critical situation is proposed. It could be noted that uniforms of opposite teams are different for a human being, but their color distribution (colors of pants and shirts are exactly reversed!) makes very difficult an automatic discrimination between them based only on spectral information. In similar conditions more refined techniques based on correlograms [4] or spatiograms [15] are not able to provide good results.

For this purpose we have driven our efforts in the direction of modifying classic histograms in order to maintain a level of geometric information about the physical position of colors. So we introduce Positional Histograms: histograms are evaluated for geometric portions of the original image, with the constraint that all points of the image have to be analyzed once. The global content of the final feature set will be the same than classic histograms, but data now are organized in a different way: the physical distribution of color values in the image is now implicitly embedded in the data representation.

We started from the players segmentation algorithm proposed in [16]. Starting from the detected players, in a first phase we have deeply compared standard descriptors (RGB histograms) with Positional Histograms, with the goal of emphasize the pros and cons of each one. A manually labeled test set have been used for this purpose. After this analysis, the different feature sets have been tested together with different unsupervised clustering algorithms on a critical dataset, in order to demonstrate both the effectiveness of the proposed data representation and the feasibility of unsupervised approaches in sport players classification.



**Fig. 1.** Example of hard-distinguishable players. In the first row it can be seen an image from a match of the National Football League between Jacksonville Jaguars and Tennessee Titans, with the draft images of the uniforms; in the second row the mean histograms of players are plotted

In the rest of the paper, firstly the Positional Histograms are presented (section 2); then, their application in automatic classification of sport teams is analyzed and experimental results obtained on real image sequences acquired during football matches of the Italian Serie A are proposed.

## 2 Positional Histograms

The main goal of this work is to investigate on the feasibility of unsupervised classification techniques in sport recognition applications. This issue requires a good data representation, in order to obtain the best vectors configuration in the feature space. So, the feature selection and extraction procedure is the focus of this section.

As demonstrated in the last section, conventional color descriptors fail in presence of team uniforms with the same main colors, but distributed in different way on the whole uniform. So we need to detect a features set able to maintain a level of relationship between global distribution and the displacement of colors on the silhouette. In presence of well differentiated uniforms, conventional histograms perform well, as well as other more refined features, like correlograms, even if these last ones are onerous in terms of computational load. Our goal is to detect a feature set able to: perform in an acceptable way (compared with histograms) in the presence of easily distinguishable uniforms; outperform histograms in the presence of difficult to distinguish uniforms; maintain a low level of computational load, that allows the integration of this module in a higher level real time sport events detection system.

For these reasons we have chosen to work with a modified version of classic histograms, called Positional Histograms. These feature descriptors maintain basic characteristics of histograms (fast evaluation, scale invariance, rotation invariance, and so on); in addition, they introduce a dependance from the position of each point in the image: the global image is partitioned according to a geometrical relationship; the histograms are then evaluated for each region, and concatenated to obtain the final region descriptor. Formally, the image  $I$  is partitioned in  $n$  subregions  $R_i$ , that satisfy the rules:

$$\bigcup_{i=1}^n R_i = I \quad (1)$$

$$R_i \cap R_j = \emptyset \quad \forall i \neq j \quad (2)$$

The first equation guarantees that each point of the image contributes to the final feature set construction, while the second one guarantees that each point gives its contribute just to one partition of histogram. In this way the final feature set contains exactly the same main information as conventional histograms, but arranged in a different way, maintaining a level of information about the spatial distribution of points in the image.

The geometric rule for the partition should be fixed according to the nature of the problem to be solved. Our experience, and also experimental results we obtained, suggests using two main geometrical partitions: the angular sectors and the circular rings, and their fusion version (circular sectors). Polar coordinates allows an easy definition of the partitions. Each region  $R_i$  is composed by points  $(x, y)$  that satisfy:

$$R_i = \{(x, y) \mid x = r \cos \theta, y = r \sin \theta \\ r_{MIN}^i < r < r_{MAX}^i, \theta_{MIN}^i < \theta < \theta_{MAX}^i\} \quad (3)$$

With this notations, we can now explore the details of each partition used in this paper. The starting point of each partition is the center of the image, where reasonably is concentrated the main informative content (a good object detector/tracker is able to maintain the subject in the center of the image).

### Angular Sectors

In this case each partition is obtained by varying the angle in a given range, according to the desired details level, while the radius ranges in all available values. So, considering  $D$  as the main diagonal of the image, and  $n$  the number of desired sectors, we have:

$$r_{MIN}^i = r_{MIN} = 0; \quad r_{MAX}^i = r_{MAX} = D/2 \quad (4a)$$

$$\theta_{MIN}^i = \theta_0 + \frac{2\pi}{n}(i-1); \quad \theta_{MAX}^i = \theta_0 + \frac{2\pi}{n} * i \quad (4b)$$

$$i = 1..n \quad (4c)$$

In figure 2 we have plotted some examples of masks for the regions creation in the presence of Angular Sectors partitions; the first row refers to masks for  $n = 4$  and  $\theta_0 = 0$ , while the second one refers to masks for  $n = 8$  and  $\theta_0 = 0$ .

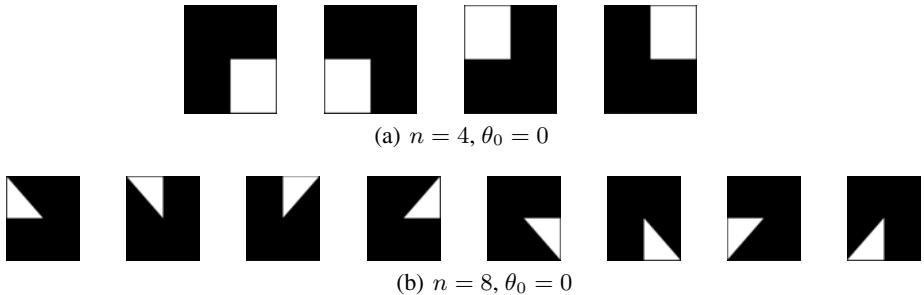
### Circular Rings

Each partition is obtained by varying the radius in a given range, according to the desired details level, while the angle varies in order to cover all possible values between 0 and  $2\pi$ . So, considering  $D$  as the main diagonal of the image, and  $n$  the number of desired sectors, we have:

$$r_{MIN}^i = \frac{D * (i-1)}{2n}; \quad r_{MAX}^i = \frac{D * i}{2n} \quad (5a)$$

$$\theta_{MIN}^i = \theta_{MAX} = 0; \quad \theta_{MAX}^i = \theta_{MAX} = 2\pi \quad (5b)$$

$$i = 1..n \quad (5c)$$

**Fig. 2.** Plot of some Angular Sectors**Fig. 3.** Plot of some Circular Rings

In figure 3 the masks in the presence of Circular Rings partitions with  $n = 2$  are plotted.

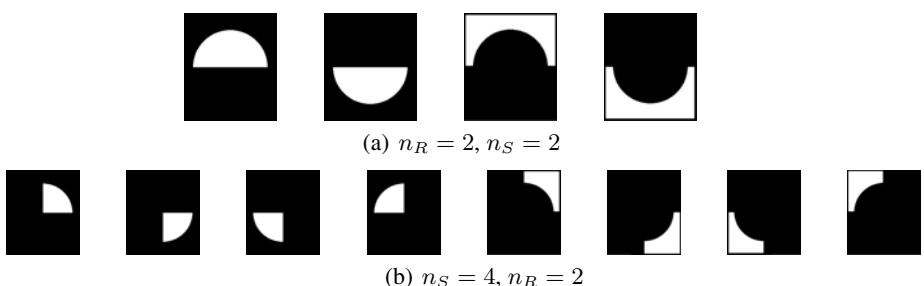
### Circular Sectors

The previously exposed partition rules can be combined (overlapped) in order to obtain another set of features that satisfies the conditions of equations 1 and 2. Now radius and angle vary simultaneously tracing circular sectors across the image. So it is necessary to define two levels of partitions: the number  $n_S$  of desired angular sectors (that influences the range of the angle  $\theta$ ) and the number  $n_R$  of desired circular rings (that influences the range of the radius).

$$r_{MIN}^i = \frac{D * (i - 1)}{2n}; \quad r_{MAX}^i = \frac{D * i}{2n} \quad (6a)$$

$$\theta_{MIN}^j = \theta_0 + \frac{2\pi}{n}(j - 1); \quad \theta_{MAX}^j = \theta_0 + \frac{2\pi}{n} * j \quad (6b)$$

$$i = 1..n_R; \quad j = 1..n_S \quad (6c)$$

**Fig. 4.** Plot of some Circular Sectors

In figure 4 some examples of masks in presence of Circular Sectors partitions are plotted.

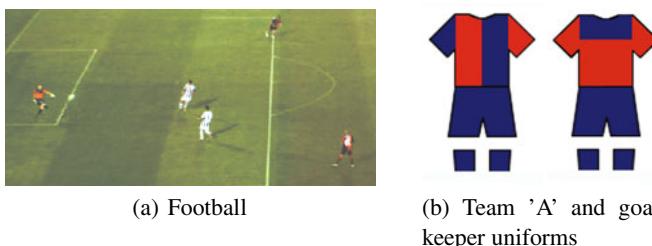
### 3 Experimental Results

Two main experiments have been carried out with the goal of:

1. demonstrate the goodness of the proposed feature set;
2. explore the feasibility of unsupervised approaches in sport video analysis.

The goal of these experiments is to demonstrate that the proposed feature set based on Positional Histograms outperforms classic histograms in critical conditions and performs in a comparable way in presence of easily distinguishable features (uniforms). To do it, we have tested the proposed features both on several grouped matches and on an entire critical football match, where some players dressed with uniforms of similar colors.

We have selected some real football matches of the Italian Serie A championship, in the presence of difficult to distinguish uniforms. Example of actors of ambiguous involved classes are proposed in figure 5: the goalkeeper of a team is dressed similarly to his teammates, even if colors are differently distributed. Note that in football applications we have to distinguish five classes: two for players, two for goalkeepers and one for referees. The particular image properties, especially the small size of players (patches are about 100\*60 pixels) suggest to use rough geometrical partitions for positional histograms. So, we have used sector based partitions (both Angular and Circular) up to 4 (it means a maximum angle of 90° for the mask creation). The selection of these features is strictly related with the nature of the problem: we have mainly tested partitions that separate the more discriminant regions (shirts and pants, respectively in the top and bottom of the image): for this reason we have worked with partitions that emphasize these topics (F2 and F4, with a 180° partition mask), excluding more fragmented ones that surely introduce a higher level of uncertainty in characterization. The F3 and F5 feature sets have been introduced and evaluated according to their capability of highlighting asymmetric structures in team uniforms.



**Fig. 5.** Example of similar uniforms from a match of the Italian Serie A

In detail, we have tested the discriminant capability of five different feature sets:

1. Classic rgb normalized histograms (in the following we refer to this set as F1);
2. Angular Sector with  $n = 2, \theta_0 = 0$  (F2);
3. Angular Sector with  $n = 4, \theta_0 = 0$  (F3);
4. Circular Sectors with  $n_S = 2, n_R = 2$  (F4);
5. Circular Sectors with  $n_S = 4, n_R = 2$  (F5);

The discriminant capability of the different feature sets has been evaluated jointly with different unsupervised clustering algorithms, in order to evaluate the feasibility of automatic techniques in sport contexts.

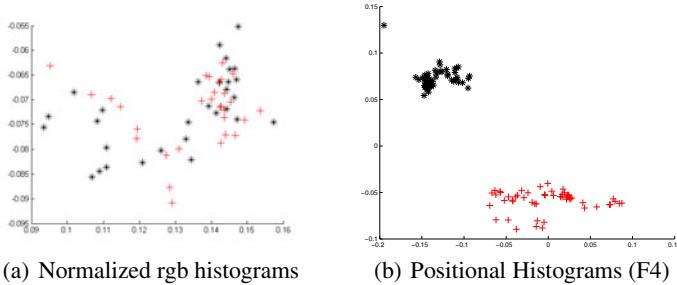
An observation needs to be made: in this kind of applications, each game is a different case, and overall results could be misleading. For example, in a match we can have well-contrasted uniforms, with well separated classes, while in another one the classes could overlap in the feature space. For this reason in the following we present results obtained both on several grouped matches (in particular for the evaluation of cluster creation phase), and on a single, random selected, match (for the evaluation of classification capability along an entire match). To make the analysis deeper, several unsupervised algorithms have been tested. In particular in our experiments we have implemented three different unsupervised clustering algorithms, belonging to different categories: MBSAS (Modified Basic Sequential Algorithm Scheme) algorithm, that is a sequential clustering algorithm, BCCLS (Basic Competitive Learning Algorithm) algorithm, that is a competitive one, and an hard-clustering algorithm, the Isodata (also known as k-means). We remand the reader to [17], Sec. 11-15 for a detailed explanation of them. Some of them need the definition of a proximity measure  $d(x, C)$ , a threshold of similarity  $th$  and the maximum number of clusters  $q$ . Euclidean distance has been used for similarity evaluations, while the maximum number of clusters has been fixed to five according to our domain constraint.

Before analyzing the results, in table 1 a summary of processing parameters for each algorithm is reported. They have been selected after several tests. For the features, we have always used 64-bin histograms. In the first experiment we have compared the capability of the training procedure to correctly detect the output clusters according to the different feature sets. For this purpose we carried out 10 experiments on 10 different matches; for each of them, about 1800 actors (players, goalkeepers and referees) images have been randomly collected in the training set, and provided to the algorithms. Details on the dataset can be found in [18].

In table 2 the results obtained for different feature sets are proposed. They have been tested jointly with different classifiers, and the resulting clusters have been manually

**Table 1.** List of parameters for each algorithm

MBSAS	BCCLS	K-Means
th=0.5	$\mu = 0.2$	k=5
	epochs=10000	exit th=0.01
		exit th=0.01



**Fig. 6.** Plot of clusters distribution of American Football players (figure I(a)) when respectively normalized rgb histograms and proposed Positional Histograms (F4) have been used for feature representation

evaluated by a human operator. As it can be noted, the best overall results have been reported by using the F4 feature set. However, even for this feature set, the perfect separation of clusters has not been obtained for all sequences: by accurately observing images, in some football matches we noted that some clusters are really difficult to be distinguished even for humans. For example, sometimes one of the goalkeepers was dressed similarly to the referee, while in another match a goalkeeper was dressed like players of opposite team. In this case a correct classification based only on spectral information (without considering the player position in the play field) is really difficult also for humans.

Starting from the results of these experiments, that demonstrates the better performance carried out by using the F4 feature set, we concentrate our efforts in order to detect the best unsupervised classifier (using just F4 as features set) by analyzing an entire match. In this experiment we compared the three unsupervised classifiers during the test phase, i.e. we evaluated their capability to properly classify each actor according to the previously detected classes. In table 3 the overall performances obtained in the test phase are presented. We can note that competitive algorithms perform better than sequential one, with a classification percentage always over 80%. In particular, BCCLS based approach seems to outperform the other ones, with a classification rate over then 90%. It should be noted that performances on an entire match are obviously worse if compared with results of the previous experiments (table 2), that refer to player images acquired just in the first minutes of the game: this is due to the variations in light conditions, that alter the color perception, and reduce the classification rate.

**Table 2.** Evaluation of different feature sets and classifiers on manually labeled sequences

	F1	F2	F3	F4	F5
<b>MBSAS</b>	71.24%	86.22%	89.31%	93.12%	83.32%
<b>BCCLS</b>	77.77%	87.33%	91.37%	95.78%	88.21%
<b>K-Means</b>	81.43%	88.04%	89.38%	94.31%	87.74%
<b>Overall</b>	78.99%	87.31%	89.11%	94.96%	89.66%

**Table 3.** Overall performance of the classifiers with F4 features

MBSAS	BCLS	K-Means
83.33%	86.65%	91.23%

Finally, with reference to the American Football match proposed in figure I(a), we have processed some minutes of match: as previously emphasized, classic histograms fail in classes separation. By analyzing the images, it can be noted that the main problem is the similarity between shirts of one team and pants of the opposite one. So it is reasonable that Positional Histograms based on Angular (F2) or Circular (F4) Sectors are sufficient to highlight the main characteristics of uniforms, allowing a good separation between clusters. In figure 6 we have plotted the clusters configuration in the features space obtained in presence of classic histograms and the F4 Positional Histograms. As evident, now feature vectors are well separated, and an unsupervised classification algorithm probably would easily distinguish them.

## 4 Discussion and Conclusions

In this paper, different color descriptors and unsupervised classifiers are studied in the context of automatic player discrimination in football matches. We introduce a novel color descriptor based on Positional Histograms, with the goal of improving classic histograms by introducing a level of relationship between spectral information and spatial position. In this way it has been possible to distinguish between players dressed with similar colors but distributed in a different way on the uniforms. The goodness of the proposed approach has been evaluated on a labeled training set; then, different features have been tested jointly with different unsupervised classifiers. After the experiments on real sequences, we can conclude that the better performances were carried out by using the F4 feature set and an unsupervised classifier based on BCLS algorithm.

As a future work, several statistics parameters, as mean intra-class and inter-class distances, intra-class standard deviation, and the mutual interaction between these parameters have to be evaluated in order to assert in an unambiguous way about the discriminant capabilities of these features. Moreover, we are evaluating the relationship between classification rates and partitions size.

## References

1. Assfalg, J., Bestini, M., Colombo, C., Del Bimbo, A., Nunziati, W.: Semantic annotation of soccer videos: automatic highlights identification. Computer Vision and Image Understanding 92, 285–305 (2003)
2. Zhong, D., Shih-Fu, C.: Real-time view recognition and event detection for sports video. Journal of Visual Communication and Image Representation 15, 330–347 (2004)
3. Xie, L., Xu, P., Chang, S.F., Divakaran, A.: Structure analysis of soccer video with domain knowledge and hidden markov models. Pattern Recognition Letters 25, 767–775 (2004)
4. Ekin, A., Tekalp, A., Mehrotra, R.: Automatic soccer video analysis and summarization. IEEE Transactions on Image Processing 12, 796–807 (2003)

5. Hayet, J., Mathes, T., Czyz, J., Piater, J., Verly, J., Macq, B.: A modular multicamera framework for team sports tracking. In: IEEE Conf. on Advanced Video and Signal based Surveillance, pp. 493–498 (2005)
6. Vandenbroucke, N., Macaire, L., Postaire, J.: Color image segmentation by pixel classification in an adapted hybrid color space. application to soccer image analysis. Computer Vision and Image Understanding 90, 190–216 (2003)
7. Ekin, A., Tekalp, A.: Robust dominant color region detection and color-based applications for sports video. In: International Conference on Image Processing, pp. 21–24 (2003)
8. Naemura, N., Fukuda, A., Mizutani, Y., Izumi, Y., Tanaka, Y., Enami, K.: Morphological segmentation of sport scenes using color information. IEEE Transactions on Broadcasting 46, 181–188 (2003)
9. Misu, T., Gohshi, S., Izumi, Y., Fujita, Y., Naemura, N.: Robust tracking of athletes using multiple features of multiple views. In: 12th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, vol. 12, pp. 285–292 (2004)
10. Xu, M., Orwell, J., Lowery, L., Thirde, D.: Architecture and algorithms for tracking football players with multiple cameras. IEEE Proc. Vision, Image and Signal Processing 152, 232–241 (2005)
11. Xu, Z., Shi, P.: Segmentation of players and team discrimination in soccer videos. In: IEEE Int. Work. VLSI Design Video Tech., pp. 121–212 (2005)
12. Yu, X., Sen Hay, T., Yan, X., Chng, E.: A player-possession acquisition system for broadcast soccer video. In: International Conference on Multimedia and Expo., pp. 522–525 (2005)
13. Beetz, M., Bandouch, J., Gedikli, S.: Camera-based observation of football games for analyzing multi-agent activities. In: Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems, pp. 42–49 (2006)
14. Huang, J., Ravi Kumar, S., Mitra, M., Zhu, W.J., Zabih, R.: Spatial color indexing and applications. International Journal on Computer Vision 35, 245–268 (1999)
15. Birchfield, S., Rangarajan, S.: Spatiograms versus histograms for region-based tracking. In: IEEE Conf. on Computer Vision and Pattern Recognition, vol. 2, pp. 1158–1163 (2005)
16. Spagnolo, P., D’Orazio, T., Leo, M., Distante, A.: Moving object segmentation by background subtraction and temporal analysis. Image and Vision Computing 24, 411–423 (2006)
17. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Academic Press, London, ISBN 0-12-686140-4
18. D’Orazio, T., Leo, M., Mosca, N., Spagnolo, P., Mazzeo, P.: A semi-automatic system for ground truth generation of soccer video sequences. In: 6th IEEE Int. Conf. on AVSS, Genova, Italy, September 2-4 (2009)

# Facial Expression Recognition Using Facial Features and Manifold Learning

Raymond Ptucha and Andreas Savakis

Computing and Information Sciences and Computer Engineering,  
Rochester Institute of Technology, Rochester, NY

**Abstract.** This paper explores robust facial expression recognition techniques based on the underlying low dimensional manifolds embedded in facial images of varying expression. Faces are automatically detected and facial features are extracted, normalized and mapped onto a low dimensional projection surface using Locality Preserving Projections. Alternatively, processed image pixels are used for manifold construction. Classification models robustly estimate expression from the low dimensional projections in manifold space. This method performs robustly in natural settings, enabling more engaging human computer interfaces.

## 1 Introduction

Face detection, pose estimation, and expression recognition are powerful enablers in intelligent vision systems for surveillance, security, entertainment, smart spaces, and human computer interfaces. Advances in face detection [1], most notably techniques developed by Viola-Jones [2] have made near frontal face detection ubiquitous. Upon detection of the size and location of each face, there are many ways to perform facial pose estimation [3] and facial expression recognition [4],[5]. Ghodsi [6] and Cayton [7] describe various dimensionality reduction techniques.

When accurately placed, facial feature locations such as corners of eyebrows, outline of mouth, etc., can produce accurate expression estimations. Active Shape Models (ASMs) and Active Appearance Models (AAMs), initially introduced by [8] were used for expression estimation in [9],[10],[11] because they offer good feature localization and are robust over appearance variations and partial occlusions. Yeongjae [9] uses multi-frame differential AAM and Kotsia [11] requires manual placement of grid points on an initial frame for tracking. Both Stasm [12] ASM and Bolin [13] ASM are considered in this work as each ASM implementation has its own cost-benefit trade-offs with regards to speed vs. accuracy.

Erroneous facial landmark assignments decrease the effectiveness of any expression classifier. The usage of facial landmarks alone ignores skin wrinkles associated with certain expressions and the process of locating these facial landmarks can be CPU intensive. As such, there are reasons to investigate bypassing the ASM procedure and performing expression recognition directly on image pixels. He [14] uses such raw pixels along with PCA and a multi-class Minmax Probability Machine (MPM) to perform classification.

By vectorizing the facial feature points or the raw image pixels, manifold learning [6],[7] may be used to reduce the dimension of input data by identifying a low dimensional embedded space for final classification. After mapping to the low dimensional space, methods such as k-nearest neighbors (k-NN), support vector machines (SVM), and MPM are used for final classification. Shan [15] used Locality Preserving Projections (LPP) [16] dimensionality reduction along with k-NN to do expression recognition. Depending upon the subject conditions, some classifiers perform better than others. Ying [17] has proposed a system to fuse the output of multiple classifiers into a single optimal classification.

Without referring back to the neutral expression of an individual, it is often difficult to precisely measure the amplitude of the expression in question. Furthermore, the temporal signature of the expression can be used to improve performance, but expression classification without considering multiple frames is more challenging. This paper describes methods for facial expression recognition in single frames without neutral expression references.

This work contrasts using facial feature landmarks against several pixel-based feature variants in performing facial expression recognition. Within a face detection bounding box, image pixels are cropped, resized, and processed through varying normalization methods and edge detection methods. Manifold Learning (ML) techniques were utilized to reduce the high dimensional data of both facial feature points and image pixels into as few as 3 dimensions for expression classification. A single low dimensional manifold surface is trained across all facial expressions. Final facial recognition is performed by a nearest class centroid, k-NN, and weighted k-NN.

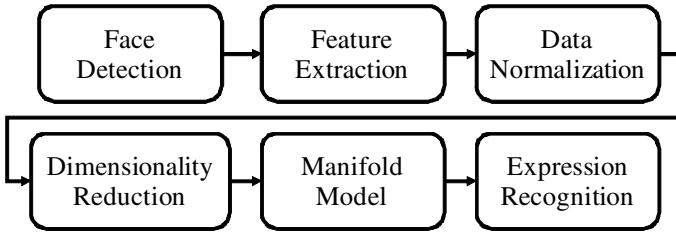
The rest of this paper is organized as follows. After the introduction, Section 2 describes the formulation of expression models, Section 3 presents facial expression recognition results, and Section 4 includes concluding remarks.

## 2 Formulation of Expression Models

Facial expression analysis can be classified as judgment-based or sign-based [5] methods. The former directly maps test subjects or attributes of test subjects to a predefined number of expression classes. The latter first deciphers facial motion into action classes such as Facial Action Coding System (FACS) [18] whereby groupings of muscles in the face form Action Units (AUs), the motions and combinations of which enable final classification. The methods in this paper use the judgment-based approach.

Figure 1 shows a flowchart illustrating the expression recognition approach used in this paper. Faces are detected, pixels are cropped, facial features are extracted, the data is normalized and projected onto a low dimensional manifold space, and a manifold model is used for expression recognition.

Face detection over a range of poses is accomplished using the Viola-Jones method, which detects most faces over  $\pm 20^\circ$  yaw and reasonable pitch ranges. The ASM algorithm is initialized using the locations of eye centroids and generates a vector of  $n$  feature positions for the eyes, eyebrows, nose, mouth, and face boundary. Anthropometry measures based upon standard faces [19] locate the seed eye positions for the ASM algorithm. Alternatively, SVM eye classifiers may be used. The resulting



**Fig. 1.** Flowchart of the facial expression recognition framework

$n$  ASM points represent the input face with variable size, location, and rotation. Scale, translation, and roll (in-plane rotation) are removed via a generalized Procrustes analysis. For this work, out of plane rotation is limited to  $\pm 20^\circ$  yaw and pitch.

In addition to ASM points, image pixel features can be used for expression recognition. Using eye centroid estimates, the facial area is cropped to a 1:1.3 aspect ratio centered on the eyes horizontally. In the vertical direction, the top of the crop box is above the eye centerline by an amount equal to two-thirds the interocular distance. This crop box is then resampled to a normalized size based on interocular distance. Pixel processing included: no processing; normalized to zero mean and unit variance; normalized, then over-sharpened with a 3x3 circular symmetric FIR filter; edge detection with 5x5 Laplacian of Gaussian; Canny edge detection; and Gabor processed. To avoid overfitting the training set, each training sample's eye coordinates are adjusted by modifying eye X centers by +/- 3 pixels and Y centers by +/- 1.5 pixels in a random fashion.

The Stasm ASM algorithm [12] produces 68 feature points in 2-D space to yield a 136 dimension input space. The Bolin ASM algorithm [13] produces 82 feature points in 2-D space to yield a 164 dimension input space. The Stasm ASM runs about 5x the speed of the Bolin ASM, but has smaller facial landmark templates and smaller search ranges. The processed image pixels produce anywhere from 20x26 to 50x65 images, yielding 520 to 3250 dimension input space. Regardless of whether ASM points or image pixels are used, the high dimensionality feature space is parameterized by a lower dimensional embedded manifold discovered using manifold learning. The resulting lower dimensional manifold representation is more compact, more receptive to subsequent classification analysis, and easier to visualize.

The input feature space contains  $n$  samples,  $x_1, x_2, \dots, x_n$ , each sample of dimension  $D$ ,  $x_i \in \mathbf{R}^D$ . These  $n$  samples are projected onto a lower dimensional representation, yielding  $y_1, y_2, \dots, y_n$ , each output sample of dimension  $d$ ,  $y_i \in \mathbf{R}^d$ . As  $d \ll D$ , we are interested in the case where  $d << D$ . In matrix notation, the input feature space is described by  $n \times D$  matrix  $X$ , where the  $i^{\text{th}}$  row of  $X$  corresponds to  $x_i$ . In the linear projection from  $D$  to  $d$  dimensions, we have  $Y^T = UX^T$ , where  $U$  is the  $d \times D$  projection matrix.

Non-linear manifold techniques are designed to learn the structure of the embedded manifold. Locality Preserving Projections (LPP) [16] is one such technique based on the geometric structure of the input space. Unlike manifold learning techniques such as Isomap [20] and LLE [21], LPP is defined everywhere in the input space, not just on the training data. LPP is found by solving a linear approximation to the nonlinear Laplacian Eigenmap and creates an adjacency map of the top  $k$  neighbors

for each feature point  $x_i$ , weighting each neighbor by distance to form weighted adjacency matrix  $W$  and then computes eigenvectors of the generalized eigenvector problem:

$$XLX^T U = \lambda XDX^T U \quad (1)$$

where  $D$  is a diagonal matrix of the column sums of  $W$ ,  $L$  is the Laplacian matrix =  $D - W$ , and  $U$  is the resulting projection matrix. LPP can be used in unsupervised or supervised manner. In the supervised manner, the adjacency matrix is adjusted such that all similar class samples have zero distance and all dissimilar class samples have infinite distance. Empirical testing has shown supervised LPP (SLPP) to perform favorably to other linear and non-linear dimensionality reduction techniques, and is therefore used to report results in this paper.

Training faces are used in conjunction with the SLPP manifold learning technique to reveal a low dimensional manifold surface. The training points on the surface of this manifold have known facial expressions of angry, happy, neutral, sad, and surprised. Test faces are projected to this manifold surface, where a classifier estimates expression. To resolve expression from this low dimensional space, nearest class center, k-nearest neighbors, and Gaussian weighted k-nearest neighbors are evaluated. The class centers are solved during training. During runtime, the distance between the test point and each class center is calculated. The class with the smallest Euclidean distance is assigned to each point. Using Gaussian weighted k-nearest neighbors, different Gaussian widths give different results. For this work, optimum results were obtained for  $\sigma=0.003$ .

### 3 Results

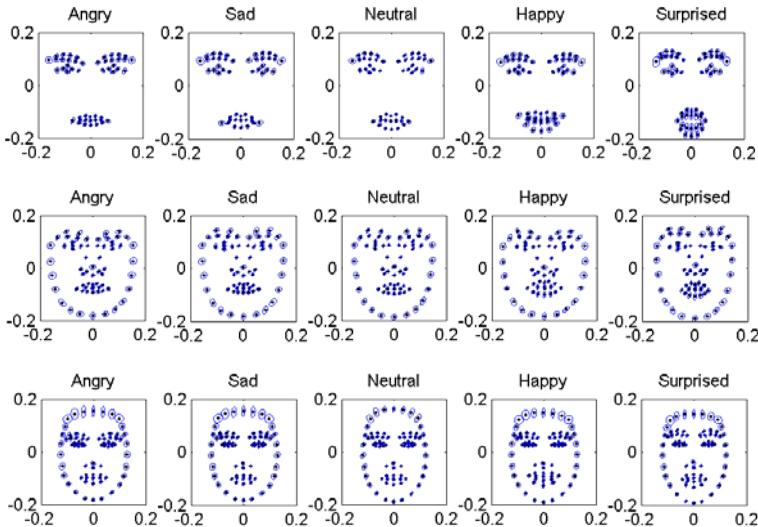
The Cohn-Kanade Facial Expression Database [22] was used as the primary ground truth data source. This database contains short video segments of 99 subjects, 29 of which exhibit angry, happy, neutral, sad, and surprised expressions. These 29 subjects were used for this study. The training set contains 1072 image frames (232 for angry, happy, sad, surprised; 144 for neutral). Each image frame has 56 manually annotated facial feature points.

Stasm ASM and Bolin ASM were automatically run on all 1072 images. This enables the calculation of three sets of ASM points:

- 1) Cohn-Kanade 56 manually annotated facial feature landmark points. These points have no position error, aside from human placement error.
- 2) Automatically generated Stasm ASM. 68 facial feature landmark points with position error.
- 3) Automatically generated Bolin ASM. 82 facial feature landmark points with position error.

A generalized Procrustes analysis on ASM point removes scaling (head size), translation, and rotation (head roll) variation amongst all subjects. Figure 2 shows sample facial landmark points for each of the three ASM point sets after Procrustes alignment. The 56 perfectly placed Cohn-Kanade points are unfairly advantaged in that both the training and testing sets had perfectly placed landmark points. The two ASM sets of points are more representative of performance expected in a natural setting.

When using the 56 perfectly placed Cohn-Kanade ground truth points, a 3-dimensional, k=3 k-NN gives an overall accuracy of 87.34% with a leave 1-subject out cross validation. Each training sequence starts from neutral, and then gradually develops to full expression, which makes it difficult to distinguish between neutral and sad on many of the subjects. When automatically generated ASM points are used to localize both training and test facial landmarks, the mean classification performance drops to 50.93% for Stasm and 65.35% for Bolin.



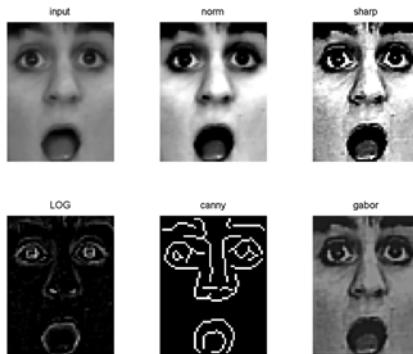
**Fig. 2.** Mean ground truth landmark points and two standard deviation limits for facial expressions. Top row is 56 point Cohn-Kanade. Middle row is 68 point Stasm. Bottom row is 82 point Bolin.

The leave 1-subject out cross validation method was repeated for several types of classification schemes. Table 1 shows the results of nearest class centroid matching, weighted Gaussian distance, and several k-NN methods for the Bolin ASM method using 3 and 4 SLPP dimensions. SLPP was not used in higher than 4 dimensions as the eigenvalues of the 5<sup>th</sup> and higher dimensions were not significant.

With regards to image pixels, the ground truth eye centroids were used to formulate the pixel crop areas which were resampled to give interocular distances from 10 to 25 pixels. This resulted in pixel crop boxes ranging in size from 20x26 to 50x65 pixels. Figure 3 shows the family of cropped and processed raw pixels for a sample surprised expression. The pixel inputs were: no processing (top left); normalized such that each had a mean of 128 and std dev of 100 (top center); normalized, then over-sharpened with a 3x3 circular symmetric FIR filter of boost 3.0 (top right); edge detection with 5x5 Laplacian of Gaussian (bottom left); edge detection with Canny Edge detector (bottom center); and four Gabor filtered images summed into a single plane using frequency= 0.3 cyc./sample, weighted phase  $((0 + \pi/2) + 0.5(\pi/4 + 3\pi/4))/3$  (bottom right). Each was processed using the leave 1-subject out methodology.

**Table 1.** Mean of diagonal of confusion matrix for various leave 1-subject out cross-validation experiments using the Bolin ASM data for training and testing

	3 SLPP Dims	4 SLPP Dims
Class Centroid	68.55	66.07
Weighted Distance	65.84	65.90
1 K-NN	63.53	64.47
3 K-NN	65.35	64.92
5 K-NN	64.31	67.17
7 K-NN	64.05	64.79
9 K-NN	64.51	66.17



**Fig. 3.** Six different variants of image pixel features studied in this paper

Figure 4 shows the normalized and sharpened images when reduced to 3 dimensions by SLPP. Tables 2 and 3 show the results with SLPP constrained to 4 dimensions using nearest class centers and 3 k-NN classification respectively. SVM classifiers with both linear and radial basis functions were found to yield similar results, but the fast and robust nearest class centers are preferred with the class separation as shown in Figure 4. Table 4 shows the resulting confusion matrix for the normalized and sharpened 50x65 images using 3 k-NN in 4 dimensional SLPP space.

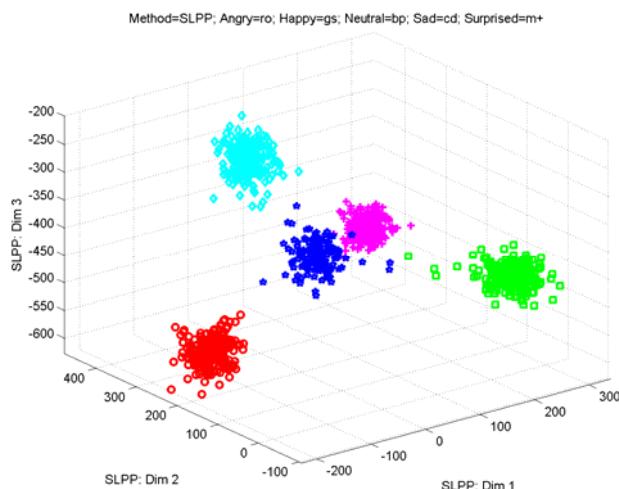
Figure 4 shows a clear intra-class separation in 3 dimensional SLPP space. With such clean separation, one might expect confusion matrices to be nearly perfect. Furthermore, previous publications, such as [15] report classification accuracies around 90%. This difference can be rectified via cross validation methodology. The commonly used k-fold cross validation selects the test and training sets randomly. Table 5 shows a comparison of 16-fold cross validation vs. leave 1-subject out cross validation. The results demonstrate that the method performs very well under 16-fold cross validation testing, while the leave 1-subject validation out is more challenging and does not generalize as well when the number of subjects is relatively small.

**Table 2.** Mean of diagonal of confusion matrix when raw pixels reduced to 4 dimensions and using nearest class center classifications

	20x26	30x39	40x52	50x65
Raw	71.72	67.66	71.41	72.03
Norm	70.63	71.56	72.81	72.97
NormSharp	69.06	66.88	71.56	74.84
LOG	66.25	63.13	62.66	64.53
Canny	50.63	51.88	50.47	55.94
Gabor	66.41	70.78	71.56	74.22

**Table 3.** Mean of diagonal of confusion matrix when raw pixels reduced to 4 dimensions and using 3 k-NN classifications

	20x26	30x39	40x52	50x65
Raw	66.72	65.78	69.84	70.63
Norm	68.28	72.81	72.50	73.59
NormSharp	68.13	70.31	73.28	75.47
LOG	64.84	60.47	62.03	65.47
Canny	50.16	51.25	51.56	55.78
Gabor	66.09	68.13	71.09	74.06

**Fig. 4.** Normalized and oversharpened raw pixels reduced to 3 dimensions using SLPP

**Table 4.** Confusion matrix using 50x65 normalized and sharpened pixels on all training and test subjects. Leave 1-subject out cross-validation. 3 k-NN classification. Mean of diagonal is 75.47%.

	Angry	Sad	Neutral	Happy	Surprised
Angry	78.13	7.03	11.72	3.13	0.00
Sad	7.03	75.00	15.63	0.00	2.34
Neutral	7.81	23.44	64.84	1.56	2.34
Happy	0.78	3.13	7.03	89.06	0.00
Surprised	0.00	12.50	6.25	10.94	70.31

**Table 5.** Classification accuracy using leave 1-subject out vs. 16-fold cross validation methodologies. Mean of diagonal of confusion matrix when normshap raw pixels (20x26 and 50x65 pixel faces shown) reduced to 4 dimensions via SLPP and classified using nearest class center, 3 k-NN, and SVM with linear kernel.

	Nearest Class Center		3 k-NN		SVM	
	20x26	50x65	20x26	50x65	20x26	50x65
Leave 1-subject out	69.06	74.84	68.13	75.47	64.69	75.31
16-fold cross validation	92.64	94.71	92.56	94.95	96.16	97.94

## 4 Conclusions

This paper presented several facial expression techniques, each of which relied on SLPP dimensionality reduction techniques. Normalized and sharpened pixel data delivered good results using 3 k-NN and SVM, however, the much faster nearest class centroid classification is the method of choice for real time processing, as it provides comparable performance while execution speed is much faster. Despite the fact that neither ASM method was trained with faces of varying expressions, both performed better than expected for expression classification. The most confusion lies between neutral and sad expressions. When used in a real-time interactive system, temporal filtering may be used to improve the classification accuracy. Further improvements may be made using multiple kernel learning approaches at the expense of slower classification compared to the simpler and faster nearest class centroid method.

## References

1. Yang, M.-H., Kriegman, D.J., Ahuja, N.: Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24(1), 34–58 (2002)
2. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features, 1, I-511-I-518, vol. 1 (2001)

3. Murphy-Chutorian, E., Trivedi, M.M.: Head pose estimation in computer vision: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 31(4), 607–626 (2009)
4. Shuai-Shi, L., Yan-Tao, T., Dong, L.: New research advances of facial expression recognition. In: 2009 Eighth International Conference on Machine Learning and Cybernetics (ICMLC), vol. 2, pp. 1150–1155 (2009)
5. Fasel, B., Luettin, J.: Automatic facial expression analysis: A survey. *Pattern Recognition*, 259–275 (2003)
6. Ghodsi, A.: Dimensionality Reduction A Short Tutorial, Department of Statistics and Actuarial Science, Univ. of Waterloo, Ontario, Canada (2006)
7. Cayton, L.: Algorithms for manifold learning, University of California, San Diego, Tech. Rep. CS2008-0923 (2005)
8. Cootes, T.F., Taylor, C.J., Cooper, D.H., et al.: Active shape models - their training and application. *Computer Vision and Image Understanding* 61(1), 38–59 (1995)
9. Yeongjae, C., Daijin, K.: Natural facial expression recognition using differential-AAM and manifold learning. *Pattern Recognition*, 1340–1350 (2009)
10. Zuo, K.-L., Liu, W.-Y.: Facial expression recognition using active appearance models. *Guangdianzi Jiguang/Journal of Optoelectronics Laser*, 853–857 (2004)
11. Kotsia, I., Pitas, I.: Facial expression recognition in image sequences using geometric deformation features and support vector machines. *IEEE Transactions on Image Processing*, 172–187 (2007)
12. Milborrow, S., Nicolls, F.: Locating facial features with an extended active shape model. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008*, Part IV. LNCS, vol. 5305, pp. 504–513. Springer, Heidelberg (2008)
13. Bolin, M.R., Chen, S.: An Automatic Facial Feature Finding System for Portrait Images. In: Society for Imaging Science and Technology: Image Processing, Image Quality, Image Capture, Systems Conference, pp. 226–231 (2002)
14. He, P., Pan, G., Zhou, Y., et al.: Facial expression recognition using multi-class minimax probability machine. In: 2008 7th World Congress on Intelligent Control and Automation, pp. 5933–5936 (2008)
15. Shan, C., Gong, S., McOwan, P.W.: Appearance manifold of facial expression. In: Sebe, N., Lew, M., Huang, T.S. (eds.) *HCI/ICCV 2005*. LNCS, vol. 3766, pp. 221–230. Springer, Heidelberg (2005)
16. He, X., Niyogi, P.: Locality Preserving Projections. *Advances in Neural Information Processing Systems* 16 (2003)
17. Ying, Z., Li, J., Zhang, Y.: Facial expression recognition based on classifier combinations. In: International Conference on Signal Processing Proceedings, ICSP. 3, Chinese Institute of Electronics; IEE; URSI; IEEE Beijing Section; National Natural Science Foundation of China; et al (2007)
18. Ekman, P., Friesen, W.V.: *The Facial Action Coding System*. Consulting Psychologists Press, Inc., San Francisco (1978)
19. Alattar, A.M., Rajala, S.A.: Facial features localization in front view head and shoulders images. In: Proceedings of the 1999 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 1999, vol. 6, pp. 3557–3560 (1999)
20. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science*, 2319–2323 (2000)
21. Roweis, S.T., Saul, L.K.: Nonlinear dimensionality reduction by locally linear embedding. *Science*, 2323–2326 (2000)
22. Kanade, T., Cohn, J.F., Yingli, T.: Comprehensive database for facial expression analysis. In: Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition, pp. 46–53 (2000)

# Blurring Mean-Shift with a Restricted Data-Set Modification for Applications in Image Processing

Eduard Sojka, Jan Gaura, Štepán Šrubář,  
Tomáš Fabián, and Michal Krumníkl

VŠB - Technical University of Ostrava, Faculty of Electrical Engineering and  
Informatics, 17. listopadu 15, 708 33 Ostrava-Poruba, Czech Republic

{eduard.sojka,jan.gaura,stepan.srubar}@vsb.cz,

{tomas.fabian,michal.krumnikl}@vsb.cz

**Abstract.** A new mean-shift technique, blurring mean-shift with a restricted dataset modification, is presented. It is mainly intended for applications in image processing since, in this case, the coordinates of the points entering into the mean-shift procedure may be obviously split into two parts that are treated in different ways: The spatial part (geometrical position in image) and the range part (colour/brightness). The basic principle is similar as in the blurring mean-shift algorithm. In contrast to it, the changes of the dataset are restricted only to the range values (colour/brightness); the spatial parts do not change. The points that are processed during computation may be viewed as points of a certain image that evolves during the iterations. We show that the process converges. As a result, an image is obtained with the areas of constant colour/brightness, which can be exploited for image filtering and segmentation. The geodesic as well as Euclidean distance can be used. The results of testing are presented showing that the algorithm is useful.

## 1 Introduction

Mean-shift (MS) is a density based clustering algorithm that was first proposed by Fukunaga and Hostetler in [1], and further studied, for example, by Cheng in [2], and Comaniciu and Meer in [3]. MS algorithms have recently attracted great attention and are now widely proposed for various imaging applications, such as image segmentation [3], object tracking [4], and image denoising [5].

Nowadays, the mean-shift algorithm as described in [3] may be regarded as a reference approach. In the original work by Fukunaga and Hostetler [1], however, the algorithm was proposed in a slightly different way that was later analysed by Cheng [2] and named as blurring mean-shift. The idea of blurring mean-shift was recently revisited by Carreira-Perpiñán in [6], where it was also shown how blurring mean-shift can be used for image segmentation. Both mean-shift and blurring mean-shift are iterative algorithms. During the iterations, the mean-shift algorithm always works with the original data set, whereas the blurring mean-shift modifies the data set in each iterative step before using it in the next

step (the basic principles of mean-shift and blurring mean-shift are summarised in Sect. 2). In [7] the algorithm named medoid-shift was introduced. In this algorithm, certain advantages are achieved at the expense that the positions that can be reached by the points during the iterations are restricted only to the original positions of input points. It seems that the following ranking of the mean-shift methods may be introduced: (i) blurring mean-shift (the dataset evolves, no restrictions on the positions reached during the iterations exist); (ii) mean-shift (fixed dataset, no restrictions on the positions reached during iterations), (iii) medoid-shift (fixed dataset, restricted positions that can be reached during the iterations; only the positions of input points can be used).

In this paper, we present a new mean-shift technique, blurring mean-shift with a restricted dataset modification. It is mainly intended for applications in image processing since, in this case, the coordinates of the points entering into the mean-shift procedure may be obviously split into two parts that are treated in different ways: The spatial part (geometrical position in image) and the range part (colour/brightness). From the point of view that the dataset evolves during the computation, the algorithm resembles to blurred mean-shift. Contrary to it, however, the changes are restricted only to the values of colour/brightness; the geometrical positions of points do not change. The points that are processed during computation may be viewed as points of a certain image that evolves during the iterations. We show experimentally that the process converges. As a result, an image is obtained with the areas of constant colour/brightness, which can be exploited for image filtering and segmentation.

By making the point positions stationary, the grid structure (usually regular) of image is preserved during the whole computation, which makes it possible to carry out faster implementations. An important motivation for creating the new algorithm was the possibility to use the geodetic distance too. Computing the geodesic distance in a fixed grid is easier than computing it under completely general conditions, which would be necessary in the case of blurring mean-shift. Moreover, an important operation in the mean-shift algorithms is to determine the points lying within a certain distance from a given point. If the grid structure of image is retained, the problem can be solved more easily.

The paper is organised as follows. The overview of the original mean-shift, blurring mean-shift, and medoid-shift algorithm is given in Section 2. Section 3 is devoted to the description of the method we propose. The issues of convergence and the clustering properties of the new method are discussed in Section 4. In Section 5, the experimental results are presented.

## 2 Mean-Shift Methods: A Review

In this section, a brief overview of the original mean-shift [3], blurring mean-shift [6,2], and the medoid-shift [7] algorithms is presented since the method we propose was inspired by them. Generally, a set  $\{x_i\}_{i=1}^n$ ,  $x_i \in \mathbb{R}^d$  of points (feature vectors) is to be processed. In images, each pixel gives one such feature vector containing the pixel position and the grey level or colour. The mean-shift

procedure is based on finding the local maxima of kernel density estimate (KDE). Usually, a special class of radially symmetric kernels is used that are of the form  $K(x) = c_{k,d} k(\|x\|^2)$ , where  $k(x)$  is a kernel profile and  $c_{k,d}$  is a normalisation constant that makes  $K(x)$  integrate to one. For  $\{x_i\}$ , the kernel density estimate at  $x$  is given by

$$\hat{f}(x) = \frac{c_{k,d}}{nh^d} \sum_{j=1}^n k(d^2(x_j, x)) , \quad (1)$$

where  $d^2(x_j, x)$  stands for the squared Euclidean length of the vector  $(x_j - x)$  that is rescaled by a chosen factor  $h$  determining so called bandwidth, i.e.,

$$d^2(x, y) = \left\| \frac{x - y}{h} \right\|^2 . \quad (2)$$

The two popular kernel profiles are the normal kernel profile  $k_N(t) = e^{-t/2}$  and the Epanechnikov profile  $k_E(t) = 1 - t$  if  $0 \leq t < 1$  and 0 otherwise.

From the stationary-point equation  $\nabla \hat{f}(x) = 0$ , the following formula can be deduced for iteratively moving a point  $x$  towards the position of the local density maximum

$$x^{(k+1)} = \frac{\sum_{j=1}^n x_j g(d^2(x_j, x^{(k)}))}{\sum_{j=1}^n g(d^2(x_j, x^{(k)}))} , \quad (3)$$

where the superscripts  $(k+1)$ ,  $(k)$  stand for the sequential number of iteration and  $g(\cdot)$  is the profile used for shifting that is connected with  $k(t)$  by  $g(t) = -k'(t)$ . For clustering on  $\{x_i\}$  in the original mean-shift algorithm, the iterations starting at each data point are run that map each  $x_i \equiv x_i^{(0)}$  to  $x_i^{(\infty)}$ . For  $x_i^{(\infty)}$ , we use the term *attractor* of  $x_i$  (clearly, only a finite number of iterations can be done in practice). Since many points can be attracted to one KDE mode, the algorithm may be used for clustering, e.g., for image segmentation.

The algorithm originally proposed in [1] was different. Following [2], we call it blurring mean-shift. In blurring mean-shift, one iteration described by Eq. [3] is carried out for each point  $x_i$ . In this way, a new dataset is obtained that is then used in the next iteration. Convergence of blurring mean-shift was studied in [2]. It has been proven that for the broad kernels (e.g., infinite-support kernels, such as the Gaussian), all the input points converge to a single attracting point. For the finite-support kernels and small enough values of  $h$ , convergence is to several attracting points; the clusters depend on the value of  $h$  and on the broadness of the kernel. The convergence to one attracting point would make the blurring mean-shift unsuitable for practical use. In [6], therefore, a stopping criterion based on analysing the evolution of clusters during the iterations is introduced, i.e., the algorithm is stopped before the convergence to a single attracting point is achieved (the Gaussian kernel is used).

In the medoid-shift algorithm, the positions that can be reached during the iterations are restricted only to the positions of input points. The algorithm does not require the definition of a mean and can operate directly on the distance matrix. In [8] the medoid-shift algorithm is further analysed. The authors note that the algorithm is not able to identify consistently all the modes of the density.

### 3 Proposed Method

The method we propose is inspired by the original mean-shift, blurring mean-shift, and by medoid-shift algorithms. The basic idea of computation is similar as in the original mean-shift. The similarly with blurring mean-shift lies in the fact the data set evolves during iterations. Similarly as in the medoid-shift algorithm, certain restriction is imposed on the moves of points. Contrary to the medoid-shift, no restriction is applied on the brightness/colour changes. The restriction we introduce lies in that the points (pixels) cannot change their geometrical position in image during computation. In Sect. 4 and 5, it will be shown that the algorithm works and that it also has some good properties. The motivation for introducing the restriction in this way is to preserve the grid structure that is typical for digital images and that often makes computation easier.

We consider an input image having  $n$  image points (pixels). The image can be regarded as a set  $\{x_i\}_{i=1}^n$  of points. Since the method runs iteratively, we add the superscript indicating the sequential number of iteration again; the input image can be written as  $\{x_i^{(0)}\}$ . We divide the vectors  $x_i^{(k)}$  into their spatial (spatial image coordinates) and range parts (grey level or colour) as follows

$$x_i^{(k)} = (s_i^{(k)}, r_i^{(k)}) . \quad (4)$$

For computing the values of the range parts of  $x_i^{(k+1)}$ , we use the following mean-shift formula that is only formally adapted from the formula in Eq. 3 used in the original mean-shift algorithm

$$r_i^{(k+1)} = \frac{\sum_{j=1}^n r_j^{(k)} g\left(d^2(x_j^{(k)}, x_i^{(k)})\right)}{\sum_{j=1}^n g\left(d^2(x_j^{(k)}, x_i^{(k)})\right)}, \quad (5)$$

where  $g$  stands for the kernel profile used for shifting, and  $d^2$  is a squared distance. In contrast to the range parts, the spatial parts of  $\{x_i^{(k)}\}$  remain stationary all the time, i.e., we have

$$s_i^{(k+1)} = s_i^{(k)} = s_i^{(0)} = s_i . \quad (6)$$

The iterative process just described runs in such a way that it starts from a discrete image  $I(s) \equiv I^{(0)}(s)$ ;  $I(\cdot)$  stands for the function of colour or brightness. A sequence of images  $I^{(1)}(s), I^{(2)}(s), \dots, I^{(\infty)}(s)$  is then computed by making use of Eq. (5) until the convergence is achieved. All the images from this sequence are defined over the same pixel grid. Therefore, all the time, we work with data that can be viewed as results of certain gradual filtering the input image. The image  $I^{(\infty)}(s)$  may be regarded as a final result (Sect. 4).

The squared distance  $d^2$  may be either Euclidean or geodesic. For the Euclidean distance, we have

$$d^2(x_i^{(k)}, x_j^{(k)}) = \left\| \frac{s_j^{(k)} - s_i^{(k)}}{h_s} \right\|^2 + \left\| \frac{r_j^{(k)} - r_i^{(k)}}{h_r} \right\|^2 , \quad (7)$$

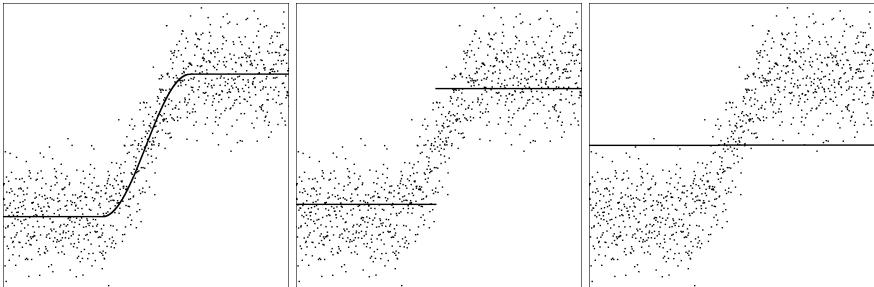
where  $h_s$ ,  $h_r$  are the quantities determining the bandwidth for the spatial and range component, respectively (generally, two different values can be used; it also holds for original mean-shift). Without loss of generality, the unit width of kernel may now be supposed. The uniform kernel or Epanechnikov kernel may be considered, for example.

Although, similarly as in the original mean-shift, the Euclidean distance may be used too, we prefer the use of geodesic distance since it gives better results (see Sect. 5). The possibility of easily using the geodesic distance was, in fact, one of the main motivations for developing this method. In the original mean-shift, the use of geodesic distance does not give a good sense since, during the computation, the points abandon the surface defined by the input image and do not create any meaningful surface later. In the blurring mean-shift, the use of geodesic distance would be possible since the evolving dataset may be regarded as repeatedly updated image function, i.e., updated surface. Unfortunately, since both the spatial and range parts of point positions change, the computation would be difficult. In the proposed method, the use of geodesic distance has a sense for the same reason as in the case of blurring mean-shift. Since the spatial components of points remain fixed, the computation can be done more easily as a length of the shortest path in a grid graph in which only the lengths of edges change (due to the changes of colour/brightness) during the computation, but the grid structure itself remains fixed. For computing the lengths of particular edges, the formula from Eq. (7) is used again. In Eq. (5), the squared geodesic distance is used. The new distances must be computed in each iteration.

Another remark deserves to be made regarding the use of geodesic distance. Classical algorithms (like, e.g., the algorithm by Dijkstra) compute the distances from one to the remaining points; the computation proceeds in the order as the distances increase. This not only allows to stop the computation of the distances from a given point if a certain distance is achieved or exceeded, but, on the other hand, it makes it also possible to easily consider a certain required number of nearest points, regardless the distance that is achieved. We may prescribe, for example, that at least a certain number (deduced from the minimal sizes of the areas we expect in image) of points must always contribute to the sum from Eq. (5). This may be understood as a dynamic change of the bandwidth, which partially compensates for the disadvantages of the finite kernel width that is required in the method.

## 4 Convergence and Clustering Properties

It seems natural that, due to its similarity with blurring mean-shift, we could expect similar convergence properties also in the new method. In this case, however, since the image points retain their spatial positions, we do not expect the result of convergence in the form of a relatively small number of attracting points. Instead, as an attractor, we should obtain a whole image function that is defined over the same number of image points (pixels) as the input image. An explanatory example for a simple one-dimensional (for clarity) image is presented in Fig. II. By setting the values of  $h_r$  and  $h_s$  appropriately, the algorithm



**Fig. 1.** Segmenting a one-dimensional image containing a theoretical edge (*solid line in the left image*) with the Gaussian noise superimposed (*points in the left image*); the horizontal axis corresponds to the spatial image coordinate; the vertical axis corresponds to the range coordinate (brightness). By making use of the proposed method, a filtered and sharpened image can be obtained if the values of  $h_r$ ,  $h_s$  are chosen appropriately, i.e., are not too big (*middle image*); the obtained attracting image function  $I^{(\infty)}(s)$  is depicted with *solid line*. Big values of  $h_r$ ,  $h_s$  inform the algorithm that only the very apparent changes in brightness are significant, and that a big neighbourhood is important. Therefore, the edge is filtered out in this case (*right image*). The results were obtained by a computer simulation.

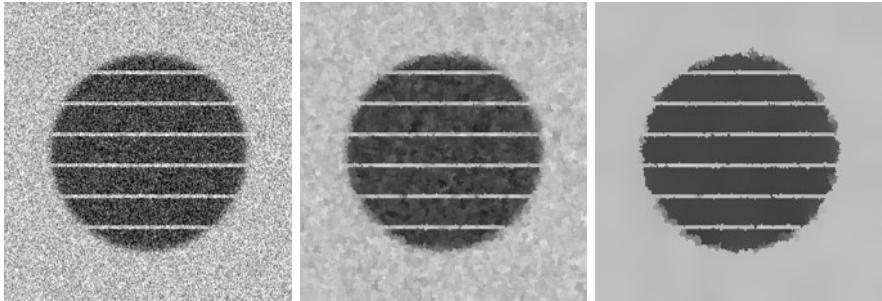
is informed how big brightness/colour fluctuations should be expected and to what spatial distance the neighbouring points should be examined. In this way, we may obtain the attracting image function that is either constant or piecewise constant (Fig. 1). For our approach, the latter case is important.

A two-dimensional example in a similar spirit can be seen in Fig. 2. In this case, an image containing a dark spot with bright stripes and added Gaussian noise was used. In the figure, the image  $I^{(1)}(s)$  obtained after the first iteration, and the image  $I^{(3)}(s)$  after the third iteration are presented (the final result  $I^{(\infty)}(s)$  is shown in Fig. 3). Relatively fast convergence of new method can be judged from Fig. 3.

The ability of the method to carry out clustering is now clear too. After achieving convergence (i.e., in the final attracting image), the clusters are created by connected areas of pixels that have the same brightness or colour (Fig. 2). Explicitly, the clusters can be easily determined by making use of usual region growing/filling algorithms.

## 5 Experimental Results

The algorithm was tested and compared with some other mean-shift algorithms (original mean-shift and medoid-shift). For computing the results presented here, the squared geodesic distance and the uniform kernel  $g(\cdot)$  were used in Eq. (5). The minimal required number of points that should contribute to the equation was also prescribed as was explained at the end of Sect. 3. Firstly, the  $256 \times 256$  grey-scale synthetic test image from Fig. 2 (left image) was used. In the new algorithm, the value of  $h_s$  was set to 10% of the theoretical spot diameter, the value



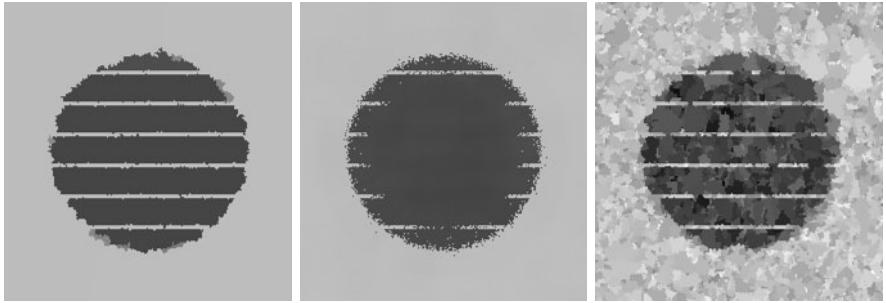
**Fig. 2.** Filtering/segmenting a two-dimensional synthetic input image containing a spot with stripes; Gaussian noise is superimposed (*left image*). The image  $I^{(1)}$  obtained after the first iteration (*middle image*) and the image  $I^{(3)}$  after the third iteration (*right image*); the final result is presented in Fig. 3.

of  $h_r$  to 10% of image brightness range. The minimal number of points that must contribute to Eq. (5) was set to 0.1% of theoretical spot size. In the remaining algorithms (original mean-shift and medoid-shift), various values of parameters were tried in order to obtain the best possible results (see further text). For visualisation of the results of all algorithms, we use the corresponding filtered images, which eliminates the influence of various possible post-processing algorithms that are needed for connecting the areas of particular attracting points in the original mean-shift and medoid-shift algorithms (see further text again).

As can be seen from Fig. 3, the result obtained by the new algorithm may be regarded as the best of all the three mentioned results (if the stripes inside the spot are to be seen). The important and determining factor for achieving this result seems to be that the geodesic distance can be used in the new algorithm.

The original mean shift-algorithm can only use a “direct distance” (e.g., the Euclidean distance; the use of geodesic distance does not give a sense). As a result, it is not possible to set the bandwidth appropriately. If we want to see the stripes, the values of  $h_s$ ,  $h_r$  should be low. Low values, on the other hand, prevent the algorithm from filtering the noise properly. If the values of  $h_s$ ,  $h_r$  are high, the algorithm filters out the noise, but also partially the stripes as is depicted in the figure.

In the case of medoid-shift, we also used the geodesic distance. The roots of rather poor result of this algorithm (Fig. 3) can be easily explained. As the attracting points, only points from the input dataset can be chosen. The points with needed brightness probably exist in the input image since the noise is Gaussian with a zero mean. It seems, however, that from many image points, no path following the image surface and leading to a correct attracting point exists such that the density always increases along that path as is required by the algorithm. Let it be pointed out that a remark in a similar spirit was also done in [8].

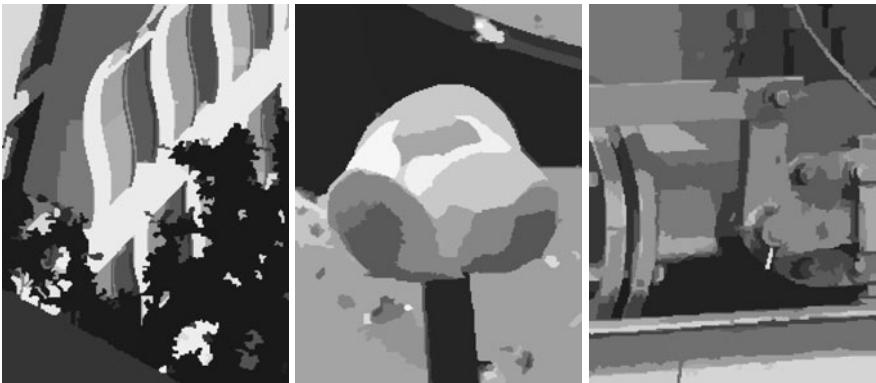


**Fig. 3.** Testing various image filtering/segmentation algorithms. The input image is the same as in Fig. 2; the results of filtering are presented for the proposed method (*left image*), for original mean-shift (*middle image*), and for medoid-shift (*right image*); see the text for further details.

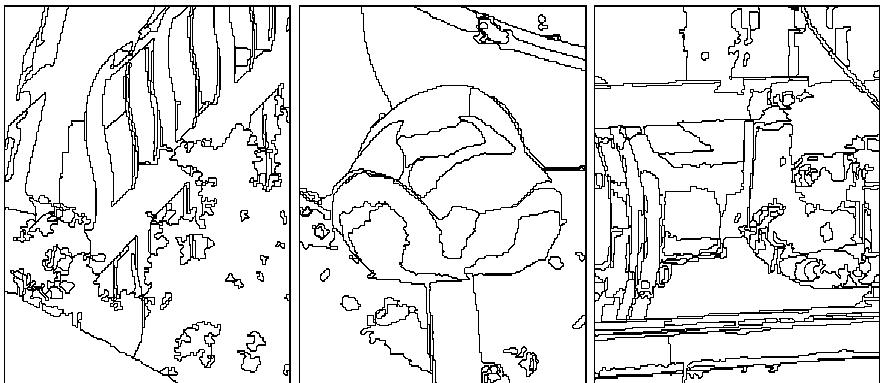
For further testing, real-life images were used too. The source images and the corresponding results obtained by the new method are presented in Fig. 4 and Fig. 5, respectively. Fig. 5, in fact, shows the corresponding attracting images  $I^{(\infty)}(s)$ , i.e., the results of filtering. The partitioning of images into the areas with a constant brightness is segmentation. In Fig. 5, such areas can be seen well; they can also be easily found algorithmically (Fig. 6). This unambiguous specification of the result may again be regarded as an advantage. In original mean-shift, one area may correspond to more than one attracting points. Therefore, a certain post-processing is needed for connecting more attracting points into their corresponding area. Although this step is mentioned by several authors [3,9] and some heuristics are proposed, no convincing rules have been presented how it should be carried out exactly. In the new method, this problematic step is avoided.



**Fig. 4.** Source real-life images for testing



**Fig. 5.** The results of filtering obtained by the new algorithm (image details are shown for better visibility). The big areas with constant brightness are present in filtered images. The degree of filtering may be adjusted by setting the values of  $h_s$ ,  $h_r$ .



**Fig. 6.** The boundaries of areas that were found by filtering

## 6 Conclusions

We have proposed a new type of mean-shift algorithm, a blurring mean-shift with a restricted data set modification. The algorithm was inspired by the original mean-shift, blurring mean-shift, and by medoid-shift algorithms. The basic idea of computation is similar as in the original mean-shift. From blurring mean-shift, the idea comes that the data set can evolve during computation. The idea of imposing certain restrictions on the moves of points is from medoid-shift. The new algorithm is an algorithm with the evolving dataset. The restriction used in the algorithm is that the points (image pixels) cannot change their geometrical position in image; only colour/brightness can change. The same, therefore, also holds for the changes of the whole dataset. Although this restriction may

seem quite severe, the algorithm is useful. By computer simulations, we have shown that the process converges; as a result an image is obtained with the areas of constant colour/brightness, which can be exploited for image filtering and segmentation. In the tests we have presented, the algorithm was better than both the original mean-shift and the medoid-shift algorithms. It seems that the good properties are mainly caused by the possibility to use the geodesic distance (although the Euclidean distance can be used too). The stationary geometrical position of pixels and changing only their colour/brightness makes the use of geodesic distance both meaningful and feasible. Certain drawback of the algorithm is its computational speed. In each iteration step and for each point, the algorithm computes the geodesic distances between that point and the points in its neighbourhood. In our experiments, it was always slower than the original mean-shift algorithm, but usually faster than the medoid-shift algorithm. Further research is being carried out on how to speed up the computation. We also recall that the algorithm requires that the data for each input point can be split into a space component and a range component. Image processing is an area where it can be easily done.

**Acknowledgements.** This work was partially supported by the grant FR-TI1/262 of the Ministry of Industry and Trade of the Czech Republic.

## References

1. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory* 21, 32–40 (1975)
2. Cheng, Y.: Mean shift, mode seeking, and clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 17, 790–799 (1995)
3. Comaniciu, D., Meer, P.: Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 603–619 (2002)
4. Comaniciu, D., Ramesh, V., Meer, P.: Kernel-based object tracking. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, 564–577 (2003)
5. Barash, D., Comaniciu, D.: A common framework for nonlinear diffusion, adaptive smoothing, bilateral filtering and mean shift. *Image and Video Computing* 22, 73–81 (2004)
6. Carreira-Perpiñán, M.A.: Fast nonparametric clustering with Gaussian blurring mean-shift. In: Airoldi, E.M., Blei, D.M., Fienberg, S.E., Goldenberg, A., Xing, E.P., Zheng, A.X. (eds.) ICML 2006. LNCS, vol. 4503, pp. 153–160. Springer, Heidelberg (2006)
7. Sheikh, Y.A., Khan, E.A., Kanade, T.: Mode-seeking by medoidshifts. In: IEEE International Conference on Computer Vision, pp. 1–8 (2007)
8. Vedaldi, A., Soatto, S.: Quick shift and kernel methods for mode seeking. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part IV. LNCS, vol. 5305, pp. 705–718. Springer, Heidelberg (2008)
9. Christoudias, C.M., Georgescu, B., Meer, P.: Synergism in low level vision. In: International Conference on Pattern Recognition, pp. 150–155 (2002)

# Detecting Straight Line Segments Using a Triangular Neighborhood\*

Shengzhi Du<sup>1</sup>, Chunling Tu<sup>2</sup>, and Barend Jacobus van Wyk<sup>2</sup>

<sup>1</sup> Department of Electrical and Mining Engineering, School of Engineering, College of Science Engineering and Technology, University of South Africa, Pretoria 0003, South Africa  
[dushengzhi@gmail.com](mailto:dushengzhi@gmail.com)

<sup>2</sup> French South Africa Technical Institute(FSATI), Tshwane University of Technology, Pretoria 0001, South Africa

**Abstract.** A novel straight line segment detection method is proposed in this paper, based on the theory of mapping straight line segment neighborhoods between the image and the HT spaces and the geometrical analysis of the HT butterfly wings. This paper makes full use of the information in the butterfly wings to detect the segments, i.e. detecting segments by matching its butterfly wings. Due to the fact that the butterfly changes its shape and orientation according to the segment parameters, this paper deduces an approximation of the butterfly wings with triangles by moving and/or flipping the segments to the position that minimizes the approximating error. This movement alleviates the computation and precision loss introduced by the butterfly distortions, because straight side triangular regions can be used to obtain the parameters of segments. Compared to existing methods that detect segments using HT data, the proposed method utilizes more information around the butterfly center, and hence is more effective, especially when it is used to detect collinear segments. The experiments verify the performance of the proposed method.

## 1 Introduction

The Hough Transform (HT) [1] is one of the most widely used techniques for locating straight lines, circles and ellipses in images. A large number of HT-based object recognition methods have been proposed in the literature [2-9].

The main idea of the HT is to map feature points  $(x, y)$  in the image space to the  $\rho$ - $\theta$  parameter space using

$$\rho = x \cos \theta + y \sin \theta, \quad (1)$$

---

\* This material is based upon work supported financially by the National Research Fundation (NRF) South Africa (Ref. IFR2010041400003). Any opinion, findings and conclusions or recommendations expressed in this material are those of authors and therefore the NRF does not accept any liability in regard thereto.

where  $\rho$  is the perpendicular distance of the line to the origin, and  $\theta$  is the angle between a normal to the line and the positive  $x$  axis. The HT maps each edge point in the discrete  $(x, y)$  space to a sine curve in the  $\rho$ - $\theta$  parameter space corresponding to all possible lines through the point as shown in eq.(II). The location  $(\theta, \rho)$  of peaks in the HT space represent the parameters of the straight lines appearing in the image, i.e. the slope and distance to the origin. Using the peak information in HT space is common to all HT related detecting methods. Except for the HT peak, the butterfly in the HT data was used to discover straight line segment parameters [13][10][11][12].

Atiquzzaman et al. [10][11] reported the microanalysis of the distribution of the votes around the peak in the accumulator array in order to determine the endpoints of a segment and hence the length was calculated as the distance between the two endpoints. Columns in the accumulator array around the HT peak were analyzed in order to find the first and the last non-zero cells. The  $\rho$  and  $\theta$  values were then used to calculate the end points. These methods did not verify the feature points in the image space which makes these methods computationally efficient. However, the existence of collinear disturbances and collinear segments which are very popular in real image processing applications were not considered. In fact, in collinear segments and multiple line segments scenarios, the collinear staff will severely affect the reliability of detecting the first and the last non-zero cells in [10][11], because there will be several isolated non-zero cells and intervals in the columns. Kamat et al. [13] discussed the multiple line segments problem, where different interesting butterflies were demonstrated due to different line segments. Du et al. [12] proposed a segments detection method making use of the quadrangle HT neighborhood, where the position of segments is represented by the position of the center points and the center point position is obtained by detecting the direction of the quadrangle neighborhood.

These butterfly based methods only used part of the butterfly wings. For example, [10][11] only used several columns of the wings and [12] only used the quadrangle part in the wings.

By considering the fact that all the cells in the wings represent a segment in a group, it is reasonable to recover the segment via synthesizing the information of the cells. This paper proposes a novel segment detection method using more information in the wings instead of the quadrangle neighborhood. By analyzing the butterfly of a segment with various of parameter settings, it is shown that the butterfly shapes evolve from straight-side triangles to curved-side triangles. Curved side shapes bring difficulties when calculating the votes in their coverage. Tense sampling could be used to approximate the curved shapes unfortunately at the expense of higher computational cost. To solve this problem, the sub-image that has the segment is moved/flipped to make the segment cross the origin to approximate the wings by straight side triangles. This motion results in lower computational cost and higher detection precision.

## 2 Observations Supporting the Proposed Method

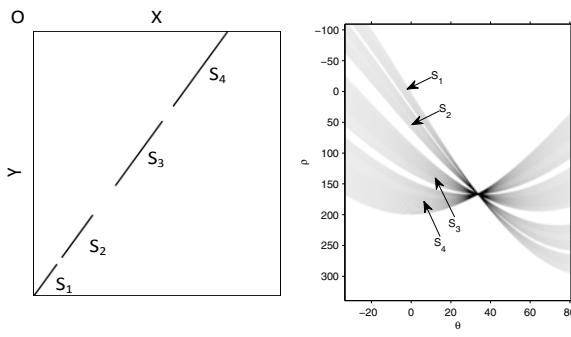
### 2.1 Butterfly Wings Represent the Unique Segment

In HT, apart from generating a peak, the process of peak formation results in a butterfly shaped spread of votes within the parameter space which varies its shape, location, width and orientation according to the slope, distance to the origin, length, and position of the corresponding segment. The relation between a butterfly and the parameters of the corresponding segment was discussed in [12] and [13]. In fact, it is easy to prove that the butterfly wings (or their lobes) are uniquely determined by the corresponding segment in the image space as shown in Fig. 1. This 1-1 mapping nature between butterfly wings (or their lobes) and corresponding segment means that the butterfly includes complete information of the corresponding segment appearing in the image space, i.e. it is possible to uncover full segment parameters from its butterfly. However, the information about how many segments the straight line have, where the segments are, and how long the segments are, is not addressed by the standard HT (SHT). This paper addresses these questions by analyzing the butterfly.

### 2.2 Information Used for Segment Detection

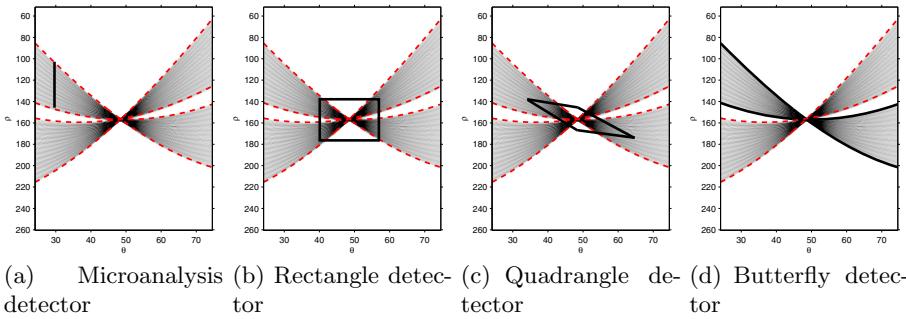
Most of the existing HT variations use the information included in the area of the neighborhood of straight lines in the parameter space. The verification process mainly focuses on the peaks appearing in HT space which are then interpreted as straight lines instead of straight line segments in image domain. In these methods, only the center of the butterfly is utilized. This leads to several difficulties as described in [2].

Several methods were proposed that utilize the information in different areas of the butterfly. The microanalysis methods [10][11] use one or two columns of the butterfly (Fig. 2(a)). Kamat et al. [13] discussed the multiple line segments problem and showed different interesting butterfly variations resulting from different line segments, and a rectangle area around the butterfly center point is



(a) Segments in the image (b) Butterfly of segments

**Fig. 1.** Straight Line Segments in Image Space and their Butterflies in HT Space



**Fig. 2.** The information used for segment detection

used to detect segments (Fig. 2(b)). Du et.al [12] proposed an approach to represent the neighborhood of straight line segments by an unique quasi-quadrangular region in the parameters space which can distinguish between collinear segments (Fig. 2(c)).

These methods rely on the fact that a butterfly shaped voting area is generated in parameter space because of the existence of edge points lying on a segment. It is therefore reasonable to investigate detecting segments using information contained in the butterfly. This paper considers the butterfly wings as the voting region of segments and proposes a segment detection method.

### 2.3 Difficulties Due to the Distortion of Wings and the Solution

In fact the butterfly wing areas around the center are usually curved side triangles bounded by the mapping sine curves of the segment end points. This deteriorates the reliability of employing the approximating quadrangle neighborhood to identify the center point of segments as proposed in [12]. This method may fail to distinct “close” collinear segments when the butterfly wings are heavily curved, because a straight side quadrangle might include information from adjacent lobes resulting in contamination. Because of this condition, using the butterfly shape area to detect the corresponding segment as mentioned in section 2.2, also has a problem, i.e. it is difficult to detect the curved bounds of the detection area. Piecewise linearizing can be considered, but dense sampling leads to a high computational load.

From eq. (1), one can obtain that the curved sides of the detection area are the function curves between  $\rho$  and  $\theta$ , and hence the camber of the curve can be represented by  $\frac{d^2\rho}{d\theta^2}$  as follows

$$\frac{d^2\rho}{d\theta^2} = -x \cos \theta - y \sin \theta = -\rho. \quad (2)$$

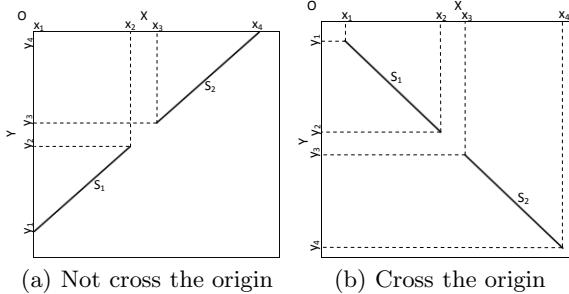
If we let

$$\rho = 0, \quad (3)$$

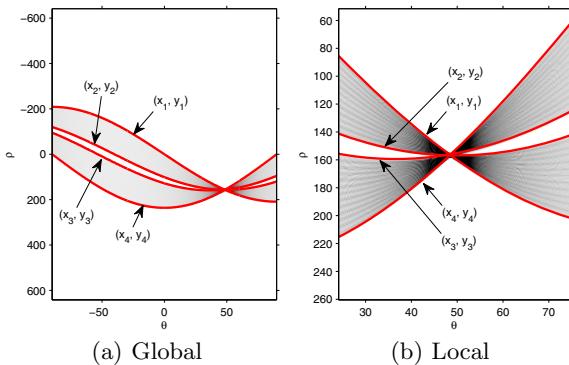
we obtain

$$\frac{d^2\rho}{d\theta^2} = 0. \quad (4)$$

This means that if the segment lies on a straight line crossing the origin then the sides of the detection region around the butterfly center can be considered as straight.

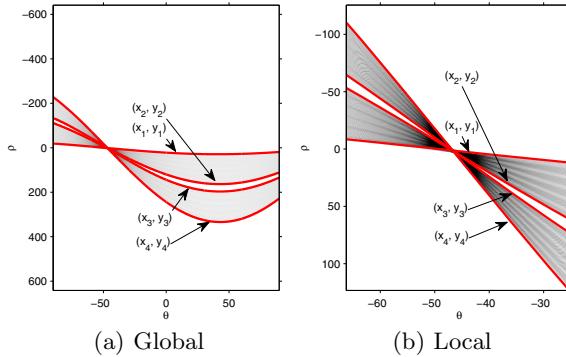


**Fig. 3.** Scenarios of segment lying on straight lines crossing and not crossing the origin



**Fig. 4.** Distortion of butterfly when segment does not cross the origin

Figs. 3, 4, and 5 demonstrate this conclusion. Fig. 3 shows two scenarios of a straight line crossing and not crossing the origin. Fig. 4 shows the curved sides of the detection region when the straight line does not cross the origin, while Fig. 5 shows that the sides of detection region are nearly straight when the straight line crosses the origin. This means one can minimize the distortion by moving or rotating a segment to a straight line crossing the origin. Using these straight sides triangles could reduce the computation and promote detection precision.



**Fig. 5.** Minimum distortion of butterfly if segment cross the origin

### 3 Using Butterflies to Detect Segments

By using the fact demonstrated in Fig. 1(b), i.e. the butterfly lobes rotate around the center when the segment moves along the straight line that it belongs to [2], Du et al. [2] detected the position of segments by counting the votes covered by the quadrangle neighborhood. This paper uses two triangular regions as shown in Fig. 6(a) instead of the quadrangle neighborhood to detect segments. The centers of these triangular regions coincide with the butterfly center and the orientation is determined by the neighborhood proposed in [2]. The height, i.e. the distance from the center to the vertical side, is selected to make the error of approximating the curved side butterfly wings by straight side triangles acceptable. For example when the height is  $10^\circ$  the approximation error is

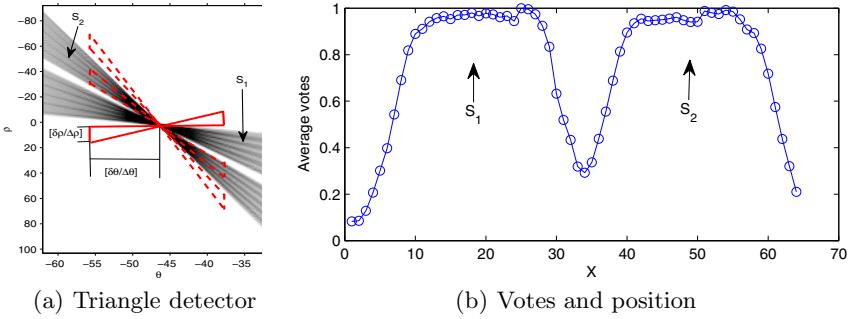
$$e = \frac{|\sin(\frac{10\pi}{180}) - \frac{10\pi}{180}|}{\sin(\frac{10\pi}{180})} \times 100\% = 0.51\%. \quad (5)$$

The length of the vertical side is selected small enough to get good distinguishing ability.

According to [2], the orientations of these triangles, i.e. the orientation of segment neighborhoods, can be obtained by enumerating the center position along the straight line corresponding to the center of butterfly shape. By counting the average votes in these triangular regions, one can obtain the relational curve between votes and the position, as shown in Fig. 6(b).

From Fig. 6(b), one can obtain:

- 1. The number of collinear segments appearing in the image space can be simply obtained by counting the number of butterfly lobe pairs around the peak in HT data;
- 2. The length of the segment can be obtained by seeking the rising- and fallsdaaing-edges of the bumps, and then the distance between these two edges is considered as the length of the project of the segment to  $X$  axis, so the length is obtained by dividing this distance by  $\cos\theta$ ;

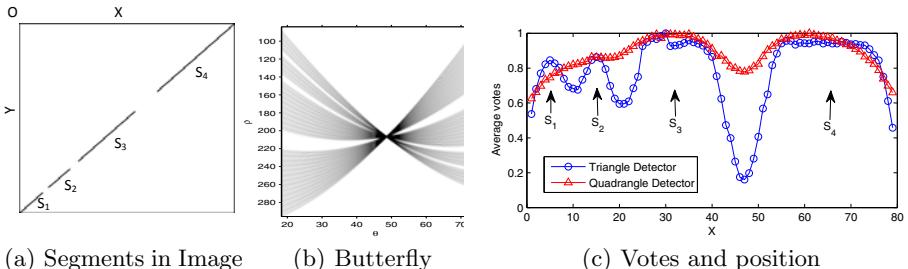
**Fig. 6.** Triangle region to detect segments

- 3. The center position of segments can be obtained by seeking the center of the bumps.

## 4 Experiments and Applications

### 4.1 The Improvement of Distinguishing Ability for Collinear Segments by Using Triangle Neighborhood

This experiment focuses on improved collinear segment distinguishing ability when the proposed method is employed. Fig. 7(a) shows the collinear segments used for this experiment. Fig. 7(b) is the butterfly corresponding to the segments in Fig. 7(a). It is clear that the wings are distorted because the distance to the origin of these segments are large (206 pixels). Fig. 7(c) shows the average votes corresponding to various segment position. It is clear that it is much easier to detect the bumps, i.e. segments, by using the votes-X curve obtained by the proposed triangular detection regions. The performance of the quadrangle neighborhood is deteriorated by the butterfly shape distortion. The latter method mixed up segments  $S_1$ ,  $S_2$  and  $S_3$  because the distances between them are not big enough. This verifies the difficulty mentioned in section 2.3.

**Fig. 7.** Improvement of collinear segments distinguishing ability

## 4.2 Image Applications

An arrow painted on a public road is used to demonstrate the performance of the proposed method. The arrow has several collinear edges with small distances. Although detection errors for some endpoints are observed, all edges are correctly detected and collinear segments are separated. This is because we only employ simple methods to detect the center of gravity, rising- and dropping-edges of bumps in vote-X curves. Advanced methods should be employed to obtain higher precision.



**Fig. 8.** Arrow mark on public road

## 5 Conclusion

Based on the theory of mapping a straight line segment neighborhood between the image space and the HT space, a new segment detection method based on the butterfly shaped regions in the parameter space, was proposed in this paper. To alleviate butterfly wings distortion, the proposed method moved/flipped the segment to a straight line crossing the origin. A pair of triangular regions are used to approximate the butterfly wings to reduce computational load. We focussed on the detection of collinear segments. The experiments show the proposed method has a high collinear segment distinguishing ability. The applications in images show that the proposed method is effective. Advanced peak searching methods should be considered to obtain higher precision in the future.

## References

1. Hough, P.V.C.: A method and means for recognizing complex patterns. US Patent 3,069,654 (1962)
2. Duda, R.O., Hart, P.E.: Use of Hough transform to detect lines and curves in picture. Communications of the ACM 15(1), 11–15 (1972)
3. Song, J., Lyu, M.R.: A Hough transform based line recognition method utilizing both parameter space and image space. Pattern Recognition 38, 539–552 (2005)

4. Duan, H., Liu, X., Liu, H.: A nonuniform quantization of Hough space for the detection of straight line segments. In: Proceedings of International Conference on Pervasive Computing and Applications ICPCA 2007, pp. 216–220 (2007)
5. Shapiro, V.: Accuracy of the straight line Hough Transform: The non-voting approach. Computer Vision and Image Understanding 103, 1–21 (2006)
6. Walsh, D., Raftery, A.E.: Accurate and efficient curve detection in images: the importance sampling Hough transform. Pattern Recognition 35, 1421–1431 (2002)
7. Ching, Y.T.: Detecting line segments in an image - a new implementation for Hough Transform. Pattern Recognition Letters 22, 421–429 (2001)
8. Cha, J., Cofer, R.H., Kozaitis, S.P.: Extended Hough transform for linear feature detection. Pattern Recognition 39, 1034–1043 (2006)
9. Fernandes, L.A.F., Oliveira, M.M.: Real-time line detection through an improved Hough transform voting scheme. Pattern Recognition 41, 299–314 (2008)
10. Atiquzzaman, M., Akhtar, M.W.: Complete line segment description using the Hough transform. Image Vision Comp. 12(5), 267–273 (1994)
11. Atiquzzaman, M., Akhtar, M.W.: A robust Hough transform technique for complete line segment description. Real-Time Imaging 1(6), 419–426 (1995)
12. Du, S., van Wyk, B.J., Tu, C., Zhang, X.: An Improved Hough Transform Neighborhood Map for Straight Line Segments. IEEE Trans. on Image Processing 19(3) (2010)
13. Kamat, V., Ganesan, S.: A Robust Hough Transform Technique for Description of Multiple Line Segments in an Image. In: Proceedings of 1998 International Conference on Image Processing (ICIP 1998), vol. 1, pp. 216–220 (1998)

# Size Distribution Estimation of Stone Fragments via Digital Image Processing

Mohammad Salehizadeh<sup>1</sup> and Mohammad T. Sadeghi<sup>2</sup>

<sup>1</sup> Department of Electrical and Computer Engineering, Concordia University,  
1455 de Maisonneuve Blvd. West, Montreal, Canada H3G 1M8

<sup>2</sup> Signal Processing Research Lab., Department of Electronics, Yazd University, Yazd, Iran  
[moha\\_sa@encs.concordia.ca](mailto:moha_sa@encs.concordia.ca)

**Abstract.** Precise statistics play a key role in the management of systems and processes. For instance, having knowledge about size distribution of stone fragments in a mining factory can allow suitable choosing of the diameter of a sieve or designing of a better crusher, hence optimizing the production line. This paper describes and compares three image-based techniques that statistically estimate stone size distribution. The techniques are watershed, granulometry and area boundary. Results show that in many mining stone factories due to identical stone texture, granulometry is a good replacement for edge detection based methods. An important point about granulometry is that its results are very qualitative; it cannot determine the exact number of stone fragments, but it can superlatively distinguish size distribution of objects in real images including objects with different textures, disparity and overlapping.

**Keywords:** Size distribution, Watershed, Granulometry, Area boundary, Segmentation.

## 1 Introduction

Enhancement in video systems and image processing algorithms is increasingly becoming an important issue in quality control and automation. Compared to traditional methods such as mechanical sieving, centrifugation and sedimentation, image-based techniques are not invasive and also have reduced sample time.

I.Levner et al., in [1], present a new method to create topographical object markers applied in watershed segmentation. In contrast to the conventional method which considers marker location and the gradient of original input image as regional minima and topological surface, respectively, this approach uses two machine learned pixel classifiers to produce markers and object boundaries applicable to multichannel data. Also in [2], the drying process of water borne paint is studied through a method based on mathematical morphology applied to the THSP (temporal history of the speckle pattern) image processing; it is based on obtaining the granulometry of these images and their characteristic granulometric spectrum. There are many applications where the texture of the images is characterized through the size distribution of objects present in them, as in the cases of granulometric classification of speckle signals related to biological fruit samples [3], morphological characterization of cultured cells

[4], etc. Similar approaches have been used by authors [5] based on scale-space decomposition to determine size distribution of oil sand materials and in [6] is discussed the case of estimation of the particles' size distribution in an image without the use of segmentation techniques by using neural networks. In this paper we investigate the performance of three mentioned algorithms on two sets of simulated images and then on photos taken from the conveyor belt of an iron ore mine. The framework described in this paper has been implemented using Matlab. Before showing the experimental results, three different methods are briefly examined.

## 2 Materials and Methods

The different methods discussed in this paper are based on two fundamental morphological operations: dilation and erosion. With  $I$  (binary image) and  $E$  (4 or 8 connected binary structuring element) assets in  $\mathbb{Z}^2$ , the dilation of  $S$  by  $E$  [1] is defined as,

$$I \oplus E = \{z / (\hat{E})_z \cap I \neq \emptyset\} \quad (1)$$

$\hat{E}$  being the reflection of  $E$  about its origin, and  $(\hat{E})_z$  the shifting of this reflection by  $z=(z_1, z_2)$ . Eq. (1) indicates that the dilation of  $I$  by  $E$  is the set of all displacements  $z$ , such that  $\hat{E} \cap I$  overlap by at least one element [1]. The erosion of  $S$  by  $E$  [1] is defined as,

$$I \ominus E = \{z | (E)_z \subseteq A\} \quad (2)$$

According to this equation, the erosion of  $I$  by  $E$  is the set of all points  $z$  such that  $E$  translated by  $z$  is contained in  $I$  [1]. The opening of a binary image  $I$  by structuring element  $E$  is a basic morphological operation for image processing which is defined as the locus of translations of the structuring element  $E$  inside the image  $I$ , given by

$$I \odot E = ((I \ominus E) \oplus E) \quad (3)$$

and is obtained by the erosion of  $I$  by  $E$ , followed by dilation of the resulting image by  $E$  [1]. The closing is another basic morphological operation which is obtained by the dilation of  $I$  by  $-E$ , followed by dilation of the resulting image by  $-E$ .

$$I \odot E = ((I \oplus (-E)) \ominus (-E)) \quad (4)$$

Opening has a different geometric interpretation from closing in the sense that opening smoothes boundaries from inside while closing does it from outside. Therefore opening can remove these connections, but closing thickens(see fig. 1, [7]).

### 2.1 Watershed

Watershed is one of most exceedingly useful image processing algorithms in the area of segmentation. After obtaining gray-level image, we perform Top-Hat transform to delete the effects of shadow. The Top-Hat transform of  $I$  by  $E$  is defined as, [7]

$$I - (I \odot E) \quad (5)$$



**Fig. 1.** (a) closing. (b) opening.

## Morphological Reconstruction

In this procedure we use morphological techniques called *opening-by-reconstruction* and *closing-by-reconstruction* to clean up the image. These operations will create flat maxima inside each object; the smoothening of the object is performed using the morphological closing by partial reconstruction operator,  $\Phi$  on the pre-processed, dilated image,  $\partial(I)$  with a reference image,  $\varphi_k(I)$  which is obtained by closing the pre-processed image  $k$  times. This is given by, [8]

$$MF(I) = \Phi^{(rec)}(\partial(I), \varphi_k(I)) \quad 0 \leq k \leq n \quad (6)$$

Where  $n$  is the size of the structure element,  $E$ . Some of the mostly-occluded and shadowed objects are not marked. The Object's edge is another problem which causes confusion of one object with several other ones. To solve this problem, we should clean the edges of the marker blobs and then shrink them a bit; we can do this by a closing followed by an erosion.

## Gradient Image Generation

In the next part, we obtain the gradient magnitude of the last image; in both directions, horizontally and vertically. A gradient helps detect ramp edges and avoids thickening and merging of edges. The gradient image,  $G(I)$  is morphologically obtained by subtracting the eroded image,  $\varepsilon(I)$  from its dilated version,  $\partial(I)$ . A multiscale gradient,  $MG(I)$  is the average of morphological gradients taken for different scales of the structure element,  $E_i$ . [8]

$$MG(I) = \frac{1}{n} \sum_{i=1}^n [\varepsilon(\partial(MF(I), E_i) - \varepsilon(MF(I), E_i), E_{i-1})] \quad (7)$$

where  $E_i$  is a SE of size  $(2i+1) \times (2i+1)$ .

## Marker Extraction

The Watershed segmentation algorithm applied directly to the gradient image can cause oversegmentation due to serious noise or image irregularities. The concept of Markers can be used to solve this oversegmentation problem whose goal is to detect the presence of homogeneous regions from the image by a set of morphological simplifications. An Internal marker or foreground is inside each of the objects of interest and external marker or background is contained within the background. [8]

Marker Extraction involves production of markers by identifying the interior of the objects to be segmented. The resulting marker image  $M(I)$  is a binary image such that

a pixel is a marker (made black) if it belongs to a homogeneous region, and a pixel will be white if it does not belong to homogeneous region. If an object is not marked properly, then the final segmentation will miss that object, too many markers will lead to extreme oversegmentation and too fewer markers will merge different objects. To make sure that the interior of an object is kept as a whole, the extracted markers are imposed on the gradient image,  $MG(I)$  as minima and all other gradient minima are suppressed. [8]

## Image Segmentation

The Gradient image which is marker extracted is subjected to watershed segmentation. Watershed segmentation [9,10] produces a more stable segmentation of objects including continuous segmentation boundaries by a concept of producing catchment basin (watershed) and watershed line (divide lines or dam boundaries). After performing distance transform on the binary image, we find SKIZ<sup>1</sup> that are virtually the watershed lines of the image. If we treat the values of an image as a relief map describing height in a geographical terrain, then an image can be segmented by partitioning it into areas that correspond to catchment basins in the geographical watershed. The watershed transformation of an image is a mapping from the original image to a labeled image such that all points in a given catchment basin have the same unique label. In the given segmentation technique, watershed segmentation algorithm is enriched with additional information about the regions of interest through the preprocessed gradient image and the marker extracted image. Gradient image provided the variations in gray level to segmentation algorithm and the marker extracted image without irrelevant minimal gray-level details cause watershed segmentation to produce more perfect results. Thus the watershed algorithm provides the image  $C(I)$  which was obtained from the Marker image  $M(I)$  and the Gradient image,  $MG(I)$ . For any pixel  $p$  at position  $(i,j)$ ,  $C$  can be obtained by, [8]

$$C(I)_p = \begin{cases} MG(I)_p & p \text{ is black in } M(I) \\ \frac{1}{2}(MG(I)_p + M(I)_p); & p \text{ is white in } M(I) \end{cases} \quad (8)$$

Since the Marker image,  $M(I)$  provides a rough partition of the objects, the gradient image,  $MG(I)$  avoids overmerging and the average of the Marker and the Gradient image preserves the contour of objects,  $C(I)$  ensures that both interior and contour of an object will be detected as it should be. [8] In fact the gradient magnitude image is modified so that its only regional maxima occur at foreground and background marker pixels, then different counted objects are visualized by true-color labeling of the last image. An obstacle existing in size distribution estimation is the weakness of the algorithm in counting big-sized stone fragments, since they are similar in texture with small-sized stone fragments. The solution applied to this problem is the use of a threshold to locate image regional maxima. The segmented output (Fig. 2) is more pleasing without over-segmentation.

---

<sup>1</sup> Skeleton Influence Zone.



**Fig. 2.** (a) Input simulated image, (b) Watershed output image

## 2.2 Granulometry

In the preprocessing part, one method used to remove the background effect is gray-level slicing. The most important part of granulometry preprocessing is image contrast enhancement performed by CLAHE<sup>2</sup> transform and then normalizing image histogram. Granulometry deals mainly with determining the size distribution of particles in an image without explicitly segmenting each object first. To achieve this, successive opening operations with scaled structuring elements are performed on the original image. As the scale increases, the particles with size smaller than the size of the structuring element are removed. The pattern spectrum is obtained computing the image difference between the original image and its opening by the structuring elements of increasing size. For an image  $I$  and a structuring element  $E$ , then for a variable parameter  $t \geq 0$  the family of opening images  $I \odot tE$  is called granulometry. In other words, granulometry is a finite union of openings, each by a parametrized convex, compact structural element that must contain the origin. [2] Consider a continuous grayscale image as a 3-D, then the total gray intensity of the image represents the volume of the image object [2]. If  $\Omega(t)$  is the volume of the opened image by the structuring element  $tE$ , then  $\Omega(t)$  is a decreasing function of  $t$ . The volumes  $\Omega(t)$  are normalized through the volume  $\Omega(0)$  of the original image  $I$ . The probability granulometric distribution function is then defined as

$$\phi(t) = 1 - \frac{\Omega(t)}{\Omega(0)} \quad (9)$$

This function increases from 0 to 1 as  $t$  increases [2]. The probability density function or pattern spectrum of the image is the derivative of the granulometric probability distribution with respect to the continuous parameter  $t$ : [2]

$$\phi'(t) = \frac{d\phi}{dt} \quad (10)$$

The statistical moments of the pattern spectrum of the original image can be calculated and used to characterize the image texture. In case of a digital image  $I$ , consider a sequence of structuring elements  $E_1 \dots E_K \dots$  of increasing size with  $k$  discrete, where  $E_K = KE_1$  and  $E_1$  consisting of a single pixel. Therefore, the opening

---

<sup>2</sup> Contrast-limited adaptive histogram equalization.

$I \odot E_{K+\Delta K}$  is a subimage of the opening  $I \odot E_K$ . In consequence, the opening by the scaled structuring elements yields a decreasing sequence of images: [2]

$$I \odot E_1 \geq \dots \geq I \odot E_K \geq I \odot E_{K+\Delta K} \dots \quad (11)$$

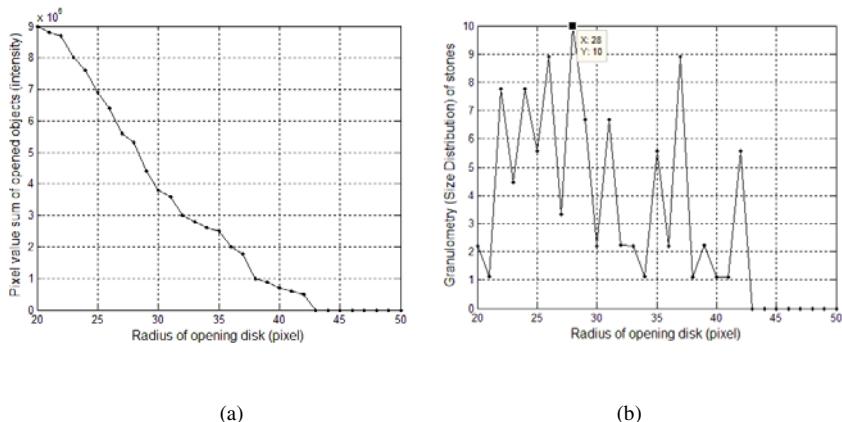
If  $\Omega(t)$  is the volume of the opening  $I \odot E_K$  for each  $k$ , then  $\Omega(k)$  is a decreasing function of  $k$ . Under the assumption that  $E_1$  consists of a single pixel and  $I$  is finite, then  $\Omega(1)$  gives the volume of the original image  $I$ , and  $\Omega(k)=0$  for sufficient large  $k$ . Normalizing the discrete volumes, substituting  $k$  in place of  $t$ , and 1 in place of 0 in (9) the discrete probability distribution function  $\phi(k)$  can be obtained:[2]

$$\phi(k) = 1 - \frac{\Omega(k)}{\Omega(1)} \quad (12)$$

Then the discrete probability density distribution is given by, [2]

$$\phi'(k) = \frac{\Delta\phi}{\Delta k} = \phi_{k+\Delta k} - \phi_k \text{ with } \Delta k = 1 \quad (13)$$

This density is called discrete granulometric size distribution. Both distributions  $\phi_k$  and  $\phi'_k$ , and consequently the granulometric moments depend on the structuring elements used in the openings. [2] Should user have information about the minimum size of image objects, opening begins with this threshold size, preventing from stray isolated pixels with lower size. A significant drop in intensity surface area between two consecutive openings indicates that the image contains objects of comparable size to the smaller opening. This is equivalent to the first derivative of the intensity surface area which signifies the size distribution of the stone fragments in the image. Notice where the minima and the radii occur in the Fig.3 (b) (distribution values are normalized and multiplied by -1).The more negative the minimum point, the higher stone fragments cumulative intensity at that radius. For example, the most negative

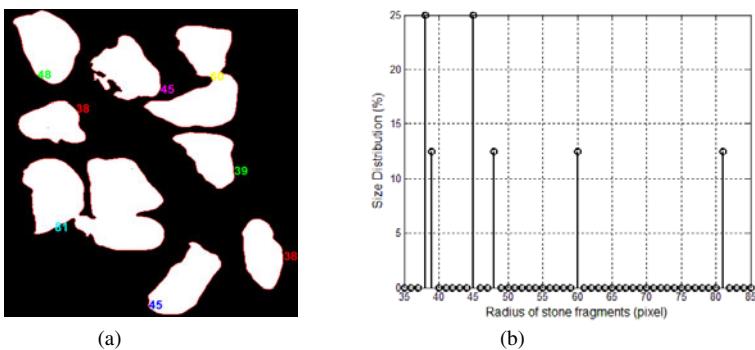


**Fig. 3.** (a) Pixel value sum of opened objects (intensity), (b) Granulometry of stones

minimum point occurs at radius by 28 pixel size in Fig.3 (b). Note that this algorithm has its best performance for images with predominant characters such as images of snowflakes in night sky or images of cultured cells [4].

### 2.3 Area Boundary

This algorithm has three sections: 1) feature extraction, 2) labeling, and 3) size-based sorting. In the first section, we found each object's boundary and geometric properties like its area, centroid and bounding box. Then, we attributed these features to a structure as its properties. In the next section, we illustrated each counted object's edge with its radius. Finally, we sorted these objects according to their radius (Fig. 4).



**Fig. 4.** (a) Labeling, (b) Size-based distribution

## 3 Results and Discussions

Average performance of each algorithm computed for 20 real images is shown in Table 1. By doing a study of Table 2, concerning simulated images (e.g. F1) containing simple texture, limited number and visible edges, watershed has the best performance. However, for real images (e.g. D2) which are more complicated, granulometry has fewer errors than other algorithms, but it is not able to count the precise number of stone fragments which is due to the aspect of granulometry discussed in the previous parts. Moreover, as illustrated in Table 2, granulometry is the only algorithm which can superlatively determine the dominant size of objects in an image. Area boundary algorithm also has noticeable error in spite of its high speed. In the following tables, for each property the best algorithm is highlighted.

One reason that gives granulometry better performance is the fact that the algorithm is based on gray-level input images instead of binary ones. As mentioned before, granulometry obtains gray-level size distribution from pixel values. A possible scenario of poor performance due to binary images is that in which the algorithm yields smaller than expected object sizes, for objects with dark inner portions. In this case, the inner dark portion is considered to be the object's background rather than part of the object itself. The image of the object is therefore filled with smaller discs and hence the reported size of the non-solid object is smaller than of a solid object.

This idea is shown below on the image of rolls. As it is observable in fig.5 (b), due to the limited range of values (0-1), granulometry has many more errors when implemented on binary images than on gray-scale images with the range of values (0-255), fig.5 (c). Note that two other algorithms are implemented on binary images.

**Table 1.** Average error

Algorithm	Error Average %	Error Deviation %	NMSE
Area boundary	21.12	9.77	.6016
Watershed	15.12	6.88	.3163
<b>Granulometry</b>	<b>11.03</b>	<b>5.57</b>	<b>.1790</b>

**Table 2.** Algorithm performance for two sample images (simulated and real)

Distribution Percent %								
Photo No.	Result	Scale(pixel)			Number	Error Average $\pm$ Deviation %	NMSE	Dominant Size
		small	Medium	Large				
		0-28	28-69	69-200				
F1 Simulated	Area boundary	0	87.5	12.5	8	8.33 $\pm$ 5.89	.0937	N/A
	<b>Watershed</b>	<b>0</b>	<b>100</b>	<b>0</b>	<b>11</b>	<b>0<math>\pm</math>0</b>	<b>0</b>	<b>N/A</b>
	Ganulometry	41.94	58.05	0	10	27.96 $\pm$ 19.7	1.055	27
	Ground truth	0	100	0	11	-	-	27
D2 Real	Area boundary	85	9.82	5.78	173	25.00 $\pm$ 13.8	.7314	N/A
	Watreshed	75.45	21.23	3.31	452	18.84 $\pm$ 8.24	.3804	N/A
	<b>Granulometry</b>	<b>47.16</b>	<b>38.00</b>	<b>14.83</b>	<b>133</b>	<b>2.22<math>\pm</math>1.54</b>	<b>.0066</b>	<b>13</b>
	Ground truth	47.20	41.3	11.50	339	-	-	13



(a)



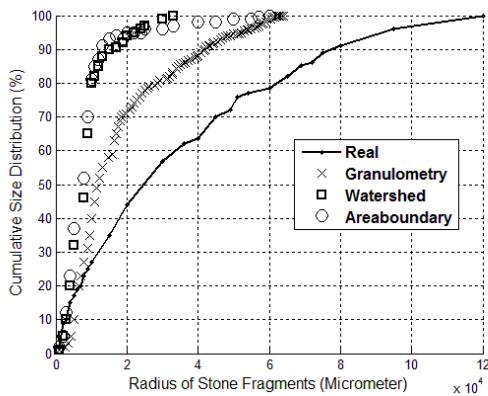
(b)



(c)

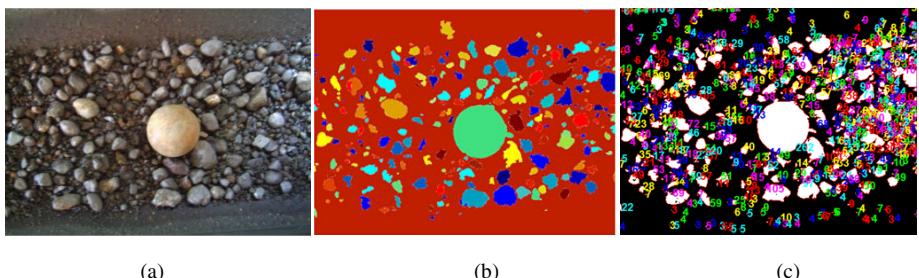
**Fig. 5.** (a) Input image of rolls, (b) Granulometry implemented on binary image, (c) Granulometry implemented on gray-scale image

Experiments have established the general flexibility of granulometry approach versus real peripheral conditions; Fig.6 illustrates experimental results corresponding to the images of group C (Ground truth data was obtained by physically sieving and weighing the stones in the images after consecutive radius). A common problem of used algorithms is that they are unable to count very small-sized stone fragments. The main reason is the existence of high frequency noises such that by using low frequency filters we will lose information about small-sized stone fragments; hence, it could be a good idea to isolate the camera area as much as possible.



**Fig. 6.** Experimental results corresponding to the images of group C

In the following figure, each algorithm is implemented on a real image from group C.



objects of different textures, disparity and overlapping. For future work, using complementary granulometry based on wavelet transform is proposed.

## References

1. Levner, I., Zhang, H.: Classification driven Watershed segmentation. *IEEE Transaction on Image Processing* 16(5) (May 2007)
2. Mavilio, A., Fernández, M., Trivi, M., Rabal, H., Arizaga, R.: Characterization of a paint drying process through granulometric analysis of speckle dynamic patterns, 2009 Elsevier B.V. *Signal Processing* 90, 1623–1630 (2010)
3. Blotta, E., Pastore, J., Ballarin, V., Rabal, H.: Classification of dynamic speckle signals through granulometric size distribution. *Latin American Applied Research Journal* 39, 179–183 (2009)
4. Prodanov, D., Heeroma, J., Marani, E.: Automatic morphometry of synaptic boutons of cultured cells using granulometric analysis of digital images. *Journal of Neuroscience Methods*, Elsevier (2005)
5. Zadoro Zny, A., Zhang, H.: Contrast enhancement using morphological scale space. In: *Proceedings of the IEEE International Conference on Automation and Logistics*, Shenyang, China, pp. 804–807 (August 2009)
6. Ferrari, S., Piuri, V., Scotti, F.: Image Processing for Granulometry Analysis via Neural Networks. In: *IEEE International Conference on Computational Intelligence for Measurement Systems and Applications*, Instabul, Turkey, July 14–16 (2008)
7. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn. Prentice-Hall, Upper Saddle River (2002)
8. Nallaperumal, K., Krishnaveni, K., Saudia, S.: A novel multi-scale morphological Watershed segmentation algorithm. *International Journal of Image Science and Engineering* 1(2), 60–64 (2007)
9. Lotufo, R., Silva, W.: Minimal set of markers for the watershed transform. In: *Proceedings of ISMM*, pp. 359–368 (2002)
10. Mukhopadhyay, S., Chanda, B.: Multiscale Morphological Segmentation of Gray Scale Image. *IEEE Transactions on Image Processing* 12(5), 533–549 (2003)

# Image Enhancement by Median Filters in Algebraic Reconstruction Methods: An Experimental Study

Norbert Hantos and Péter Balázs\*

Department of Image Processing and Computer Graphics  
University of Szeged  
Árpád tér 2. H-6720, Szeged, Hungary  
[{nhantos,pbalazs}@inf.u-szeged.hu](mailto:{nhantos,pbalazs}@inf.u-szeged.hu)

**Abstract.** Algebraic methods for image reconstruction provide good solutions even if only few projections are available. However, they can create noisy images if the number of iterations or the computational time is limited. In this paper, we show how to decrease the effect of noise by using median filters during the iterations. We present an extensive study by applying filters of different sizes and in various times of the reconstruction process. Also, our test images are of different structural complexity. Our study concentrates on the ART and its discrete variant DART reconstruction methods.

**Keywords:** algebraic reconstruction methods; discrete tomography; median filter; image enhancement.

## 1 Introduction

The main task of tomography is to reconstruct images representing two-dimensional cross-sections of three-dimensional objects from their projections. Undoubtedly, the main applications of tomography arise from the field of medicine, but it is a very useful imaging procedure also in physics, chemistry, biology, industry, and so on. Nowadays, the most widely used reconstruction methods are the transformation-based ones – like the filtered backprojection, inverse-Radon transform, etc. – which need several hundreds of projections to ensure an acceptable image quality [1]. However, in some applications, e.g. in ultramicroscopy [2] or in neutron tomography [3] only a few projections are possible to acquire due to potential damage the electron beams can cause in the first case, and the expense of the acquisition procedure in the latter one. In such cases the above mentioned techniques usually do not give good results. Algebraic reconstruction methods can perform well even with a smaller number of projections but - unfortunately - the image they produce can be very noisy [8].

\* Corresponding author. This research was partially supported by the TÁMOP-4.2.2/08/1/2008-0008 and TÁMOP-4.2.1/B-09/1/KONV-2010-0005 programs of the Hungarian National Development Agency and by the János Bolyai Research Scholarship of the Hungarian Academy of Sciences.

One way to enhance noisy images is to use filters during the iterations. Filters are well-known techniques in digital image processing for decreasing noise. However, designing appropriate filters for a given image processing problem is not a simple task. The aim of this paper is to perform an extensive study on what filtering techniques can significantly improve the quality of the reconstructed image.

We are especially interested in the effect of median filters for decreasing the noise. We present a general framework by using filters of different sizes and with different frequency during the reconstruction. We investigate two variants of the algebraic reconstruction methods. In addition, test images of varying topology and complexity will be used to get a sufficiently deep insight into the behavior and performance of the algebraic reconstruction methods.

The structure of the paper is the following. In Section 2 we give some details on the reconstruction task. In Section 3 we describe the variants of the algebraic reconstruction methods we examined. The importance of applying filters in the reconstruction is pointed out in Section 4. In Section 5 we present our experimental results. Finally, Section 6 is for the conclusion.

## 2 The Reconstruction Problem

The central problem in tomography is to reconstruct a two-dimensional image, from its projections. In the continuous setting the task is to recover an unknown function  $f(x, y) : \mathbb{R}^2 \rightarrow \mathbb{R}$  (the image) from a bunch of its line integrals given by the Radon-transform

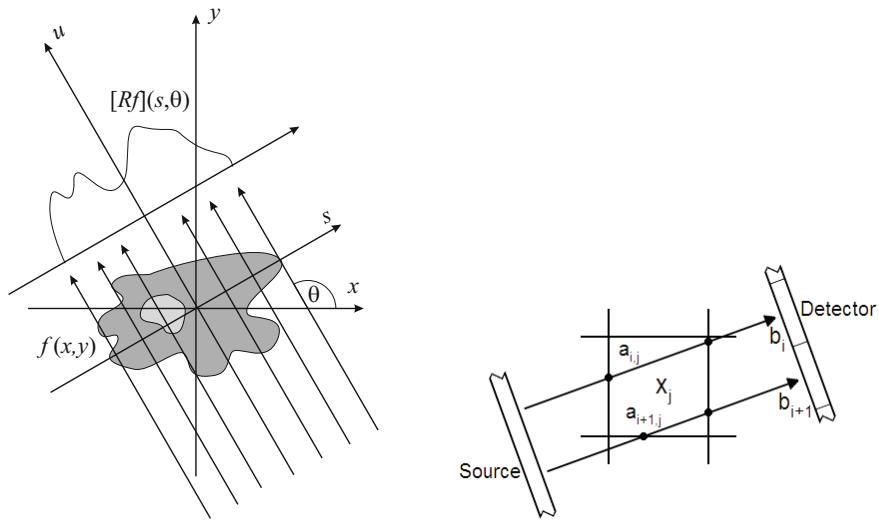
$$[Rf](s, \theta) = \int_{-\infty}^{\infty} f(s \cos \theta - u \sin \theta, s \sin \theta + u \cos \theta) du \quad (1)$$

for certain fixed  $\theta$  angles. Here  $s$  and  $u$  denote the variables of the coordinate system rotated by the angle  $\theta$  and  $[Rf](s, \theta)$  is called the  $\theta$ -angle parallel projection of  $f$  (see the left of Fig. 1).

Although there is a thorough mathematical theory and an exact formula for continuous reconstruction when all the possible projections are available, in a practical application we can only collect a finite number of projections. On the other hand, the function domain is also discretized, as the image is represented by a set of pixels (or more formally, by a matrix). We stress the fact, that the range of  $f$  not necessarily has to be discrete. Within this context the projection rays can be regarded as straight lines passing through the pixels, and the projection value along a ray is calculated as the weighted sum of the pixel values hit by that ray. This yields that the reconstruction problem can be formulated as the task of solving a matrix equation

$$\mathbf{Ax} = \mathbf{b}, \quad \mathbf{A} = (a_{i,j})_{n^2 \times m} \in \mathbb{R}^{n^2 \times m}, \quad \mathbf{x} \in \mathbb{R}^{n^2}, \quad \mathbf{b} \in \mathbb{R}^m, \quad (2)$$

where the image to be reconstructed is assumed to be of size  $n \times n$  (which can be assumed without loss of generality),  $m$  is the total number of projection rays



**Fig. 1.** A function  $f(x,y)$  and one of its projections (left), and the reconstruction geometry (right) showing a pixel  $x_j$  of the (unknown) image and two rays passing through it

used,  $a_{i,j}$  gives the weight of the  $j$ -th pixel values when calculating the projection on the  $i$ -th ray, and  $b_i$  stands for the projection value along the  $i$ -th projection ray (see the right of Fig. 1 for an illustration).

### 3 Algebraic Reconstruction Methods

Equation (2) cannot be solved directly in this form. First, the system can be underdetermined owing to the small number of projections. Moreover, due to measurement errors on projections, it can happen that the system is inconsistent, i.e. there is no solution satisfying all the equations. Algebraic reconstruction methods try to overcome these problems by iteratively approaching the solution of Equation (2). There is a broad class of continuous reconstruction methods of this kind and Algebraic Reconstruction Technique (ART) [7] serves as the basis of all of them.

The main idea of ART is to backproject the error of the actual image onto the pixels responsible to that error. More precisely, if  $\mathbf{x}$  is the current solution and  $b_i$  is the value on the  $i$ -th projection ray then the difference (the projection error) on that ray can be calculated by

$$\Delta = a_{i,1}x_1 + a_{i,2}x_2 + \dots + a_{i,n^2}x_{n^2} - b_i . \quad (3)$$

The total sum of the weights on the  $i$ -th ray is  $W = \sum_{j=1}^{n^2} a_{ij}$ . To correct the error  $\Delta$  is backprojected on the pixels hit by the ray according to the pixel weights. Thus, the new image  $\mathbf{x}'$  will have the pixel values

$$x'_1 = x_1 + a_{i,1} \frac{\Delta}{W}, \dots, x'_{n^2} = x_{n^2} + a_{i,n^2} \frac{\Delta}{W}. \quad (4)$$

In this way the error on the given ray will be zero but on some other rays it may increase. The method is repeated ray by ray, iteratively, until termination. Further details on the ART method can be found in [8].

In discrete image reconstruction (also called *discrete tomography*) we assume that the range of the function  $f$  consists of just a few, say up to 4, known discrete values. As an extremity, if the function can take only the values 0 or 1 then we are speaking about *binary tomography*. With this prior information the number of projections needed to obtain an accurate reconstruction can be extremely reduced (usually, no more than 10 projections are sufficient to achieve reconstructions of good quality). Discrete tomography has its own mathematical tools and a wide range of applications (see [9][10] for an overview).

Recently, a variant of ART has been developed for discrete image reconstruction which is called Discrete Algebraic Reconstruction Technique (DART) [2]. Thresholding an image obtained by a continuous reconstruction method provides a simple way to reconstruct discrete images. DART is based on the observation that the result of such a thresholding is often inaccurate exclusively on the boundaries of the objects. Thus, after an initial thresholding of a continuous result, DART modifies just the boundary pixels, iteratively. The main steps of the algorithm are the followings.

1. Compute an initial continuous reconstruction by using a variant of ART.
2. Threshold the actual image  $\mathbf{x}_{act}$  by using the known discrete values the image pixels can take to obtain  $\mathbf{x}'$ .
3. Compute the set  $S$  of non-boundary pixels of  $\mathbf{x}'$ .
4. Compose a new image by taking the pixel value from  $\mathbf{x}'$  if the pixel is in  $S$  and from  $\mathbf{x}_{act}$  otherwise.
5. Perform one ART iteration on the non-boundary pixels of the composed image.
6. Smooth the boundary pixels.
7. If the termination criteria is met then perform a final thresholding and quit, otherwise go to Step 2.

For a more detailed description of DART the reader is referred to [2].

## 4 Median Filters in Algebraic Reconstruction

Algebraic reconstruction methods, and in particular ART has the drawback of producing salt-and-pepper noisy images when the number of iterations is small [8]. Our aim was develop a method that can reduce this effect. For this purpose, we decided to choose the median filter, as it is especially appropriate to handle this problem (see e.g. [6]). The median filter replaces all the image pixels in the same time with the median of the image pixel values in a predefined neighborhood of the given pixel. More precisely, if  $f(x, y)$  denotes the pixel value in the position  $(x, y)$  then the new  $f'(x, y)$  value of that position will be reckoned by

$$f'(x, y) = \text{med}_{-k \leq u, v \leq k} \{f(x + u, y + v)\}, \quad (5)$$

where the median is calculated over a  $(2k + 1) \times (2k + 1)$  window, that is the size of the filter is  $(2k + 1) \times (2k + 1)$ .

Naturally, the noise in the resulted image will be reduced. Nevertheless, changing the pixel values will in most of the cases yield an image whose projection error is bigger than it was before the filtering. Since our final aim is to keep this error low, we apply the filter not in each iteration (or just at the end) of the reconstruction process but only in each  $l$ -th turn. Thus, we hopefully obtain a smooth result whose projection error is acceptable, too.

Another important observation is that in the case of DART we can apply the filter at the beginning of the process to obtain a starting reconstruction, as well as in the inner loop of the procedure. Filters of this latter type will be referred to as *inner filter*.

## 5 Experimental Results

### 5.1 Implementation Details

To investigate the effect of median filtering under different circumstances we developed a general reconstruction framework.<sup>1</sup> We assumed that the distance between two consecutive parallel projection rays is always unitary (equal to the length of the pixel edges), for any directions. Moreover, in order to ensure a fast implementation, the weights  $a_{ij}$  are set to 1 or 0, according to whether the  $i$ -th ray hits the  $j$ -th pixel or not, respectively. That is,  $\mathbf{A}$  is assumed to be a binary matrix. Although more sophisticated weight settings could be done, this simple approach is sufficient for our investigations, and it enables us to calculate the projections very fast by using Bresenham's algorithm for drawing lines [4]. The software can perform both ART and DART reconstructions. In both cases one has to give the number of projections as input. The projections are always placed equiangularly with a starting angle of  $0^\circ$ . In addition, one also has to specify the size and the frequency of the filter to be applied during the reconstruction. In the case of DART we can choose if inner filters are also to be applied, or they are omitted. To perform the median filtering we used the fast algorithm of [12]. The reconstruction terminates if a predefined number of iterations has been reached.

The implementation was done under Windows 7, and the experiments were conducted on an Intel Core 2 Duo T2520 of 1.5 GHz, and with 2MB of RAM.

### 5.2 The Test Dataset

Our binary test dataset consisted of 6 images with different complexities. Two of them, the **Simple** and **Cylinders** phantoms can be seen on the left of Fig. 2. They were taken from [2] to make it possible to compare our results to the performance of the standard DART algorithm. These are of size 512 by 512

---

<sup>1</sup> The program is available from the authors upon request.



**Fig. 2.** Binary and continuous test phantoms used for testing the effect of filtering. The **Simple** and the **Cylinders** binary phantoms and the continuous images **Phantom1** and **Phantom2** (from left to right, respectively).

pixels. The rest of the images were of size 256 by 256 pixels, and contained simple shapes (rings, circles, ellipses, rectangles, etc.) arranged in different ways. Images like these are often used to evaluate reconstruction algorithms developed for nondestructive testing (see, e.g. [3]).

For the continuous reconstruction tests we used the two images on the right of Fig. 2, a simpler one called **Phantom1** which is a multilevel image, but with homogeneous parts, and a more complicated one (**Phantom2**) with continuous inhomogeneous parts. Both of them have the same size of 256 by 256 pixels.

### 5.3 Experimental Results

Filters can be used for two different goals. We can try to decrease the number of iterations needed to achieve a given image quality, thus reducing the time cost of the reconstruction. On the other hand, we can also try to enhance the quality of a reconstruction while keeping the number of iterations fixed. It also can happen that in this way we can get images of good quality by using less projections which is of special importance from applicational point of view. For the numerical evaluation of the quality of the binary reconstructions we calculated the relative mean error (RME) given by

$$RME = \frac{\sum_{i=1}^{n^2} (p_i - p'_i)^2}{|p_o|}, \quad (6)$$

where  $|p_o|$  is the number of white (object) pixels in the original image.

For the continuous reconstructions we used the simple error measure

$$ERR = \frac{\sum_{i=1}^{n^2} (p_i - p'_i)^2}{n^2}, \quad (7)$$

where  $p_j$  and  $p'_j$  denote the value of the  $j$ -th pixel in the original, and the reconstructed image, respectively.

In the case of the **Simple** phantom image we first tried to perform the reconstruction without filters, with the same parameters as in [2]. Here, and for all binary phantoms we performed 10 iterations of ART as the first step of DART. We mainly found the same as in [2], 5 projections are sufficient to achieve a

**Table 1.** Image **Simple**: Reconstruction from less projections with filters

#projs	#iters	Filter freq./size	Inner?	Time (s)	RME
4	120	no	no	13.1	0.120692
5	110	no	no	13.8	0.008913
6	90	no	no	12.3	0.000084
4	120	3 / 7 × 7	no	13.3	0.095846
4	120	3 / 7 × 7	yes	16.7	0.020000
4	120	2 / 7 × 7	yes	18.7	0.013433
4	120	3 / 9 × 9	yes	15.5	0.009558
4	120	4 / 11 × 11	yes	15.6	0.007423
4	120	5 / 13 × 13	yes	15.5	0.009961
4	120	5 / 11 × 11	yes	15.1	0.009875

**Table 2.** Image **Simple**: Reconstruction from less iterations

#iters	Filter freq./size	Inner?	Time (s)	RME
30	no	no	4.2	0.0088174
20	no	no	3.1	0.0118270
10	no	no	1.9	0.0291250
8	no	no	1.6	0.0344616
5	no	no	1.3	0.0441635
3	no	no	1.1	0.0549905
30	4 / 11 × 11	yes	5.3	0.0078749
20	4 / 11 × 11	yes	3.4	0.0078557
10	4 / 11 × 11	yes	2.4	0.0223558
8	4 / 11 × 11	yes	2.1	0.0364710
5	4 / 11 × 11	yes	1.7	0.0444808
3	4 / 11 × 11	yes	1.4	0.0542116

good reconstruction quality with about 100 DART iterations. The results are presented in the first three rows of Table 1. From the further rows of this table we can deduce that the result cannot be improved in the case of 4 projections if no inner filtering is allowed. However, if inner filtering is present we can get good results (especially, for the filter parameters  $4/11 \times 11$ ), even from 4 projections.

In the second test we tried to decrease the number of iterations during the reconstruction with the aid of a  $4/11 \times 11$  filter. We used 5 projections for the reconstruction. Table 2 shows that filtering in this case does not have a significant effect on the reconstruction quality.

For the image **Cylinders**, which is of highly different topology than the phantom **Simple** we get totally different results. In [2] the authors found that for this image 10 projections is necessary to get an acceptable reconstruction result. This is in accordance with our observations, too (see the first three rows of Table 3). We also found, that filtering helps not much when we tried to reduce the number of necessary projections (see the rest of Table 3). However, when our aim was to reduce the the number of iterations, it turned out that, for example in the case of a filter  $2/9 \times 9$ , the number of iterations can be extremely reduced with the

**Table 3.** Image Cylinders: Reconstruction from less projections with filters

#projs	#iters	Filter freq./size	Inner?	Time (s)	RME
9	130	no	no	24.6	0.3360736
10	110	no	no	19.9	0.0379534
11	120	no	no	24.5	0.0389934
9	130	3 / 7 × 7	yes	27.6	0.3221821
10	110	3 / 7 × 7	yes	23.2	0.0399356
11	120	3 / 7 × 7	yes	28.1	0.0409918
9	130	8 / 11 × 11	yes	25.5	0.3064226
9	130	11 / 9 × 9	yes	25.4	0.2639848
9	130	9 / 9 × 9	yes	25.5	0.3192740

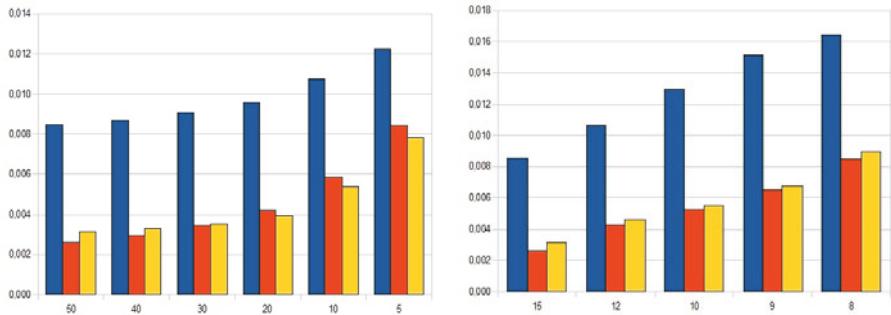
**Table 4.** Image Cylinders: Reconstruction from less iterations

#iters	Filter freq./size	Inner?	Time (s)	RME
80	no	no	15.1	0.0401468
70	no	no	12.9	0.0480104
60	no	no	11.5	0.0606834
50	no	no	9.7	0.0743960
30	no	no	6.4	0.1023899
20	no	no	4.6	0.1208957
80	2 / 9 × 9	yes	18.1	0.0342978
70	2 / 9 × 9	yes	16.2	0.0345904
60	2 / 9 × 9	yes	14.1	0.0341517
50	2 / 9 × 9	yes	11.9	0.0342654
30	2 / 9 × 9	yes	7.9	0.0342978
20	2 / 9 × 9	yes	5.9	0.0360849

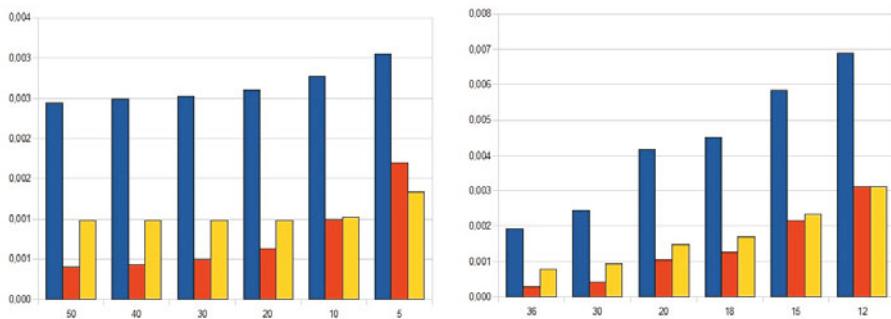
aid of the filtering, without significant degradation in the resulted image (see Table 4). The cost of filtering is just a very small increase in the reconstruction time, which becomes visible if compare the running times in the upper and lower parts of Table 4.

All the remaining 4 binary phantoms with the simple shapes showed similar behaviours (the detailed results are not presented here, due to space considerations). Either the number of projections and/or the number of iterations could be reduced by a suitably chosen (but from the image highly dependent) filter. Therefore, we turned our attention to the continuous reconstructions. Figure 3 shows the results of Phantom1 when reconstructing it from 15 projections and varying number of iterations, and when the number of iterations was fixed to 50, but the number of projections was changing.

Although Phantom2 is much more difficult (we needed here 30 projections for an accurate reconstruction) we can obtain similar results, in this case, too (see Fig. 4). Again, here the number of projections was fixed to 50 when we tested the effect of decreasing the number of iterations.



**Fig. 3.** Reconstruction errors (vertical axis) for **Phantom1** depending on the number of iterations (left) and the number of projections (right) presented on the horizontal axis, with no filter (blue), with a  $3/7 \times 7$  filter (red), and with a  $2/11 \times 11$  filter (yellow)



**Fig. 4.** Reconstruction errors (vertical axis) for **Phantom1** depending on the number of iterations (left) and the number of projections (right) presented on the horizontal axis, with no filter (blue), with a  $3/5 \times 5$  filter (red), and with a  $2/13 \times 13$  filter (yellow)

## 6 Conclusion and Further Work

Filters are effective tools to enhance the result of both binary and continuous reconstruction algorithms. As those images are typically degraded by a salt-pepper-like noise, we have chosen median filters to remove this unwanted effect. We found that a median filter of proper frequency and size can reduce the number of iterations and/or the number of projections needed to an acceptable reconstruction. Thus, median filtering during the reconstruction can play an important role, if only few projections are available or if the reconstruction should be speeded up, for some reasons. Unfortunately, finding the proper parameters of a filter is highly dependent on the image, and is a difficult task. Our further work will concentrate on this problem. We intend to apply learning methods to predict the proper filter parameters, solely from the projections, as it was carried out in [5] for some other geometrical features of the image to be reconstructed.

## References

1. Batenburg, K.J., Bals, S., Sijbers, J., Kuebel, C., Midgley, P.A., Hernandez, J.C., Kaiser, U., Encina, E.R., Coronado, E.A., Van Tendeloo, G.: 3D imaging of nanomaterials by discrete tomography. *Ultramicroscopy* 109(6), 730–740 (2009)
2. Batenburg, K.J., Sijbers, J.: DART: A fast heuristic algebraic reconstruction algorithm for discrete tomography. In: Proceedings of the IEEE International Conference on Image Processing (ICIP), San Antonio, Texas, USA, vol. 4, pp. 133–136 (2007)
3. Baumann, J., Kiss, Z., Krimmel, S., Kuba, A., Nagy, A., Rodek, L., Schillinger, B., Stephan, J.: Discrete tomography methods for nondestructive Testing. In: [10], ch. 14
4. Bresenham, J.E.: Algorithm for computer control of a digital plotter. *IBM Systems Journal* 4, 25–30 (1965)
5. Gara, M., Tasi, T.S., Balázs, P.: Learning connectedness and convexity of binary images from their projections. *Pure Mathematics and Applications* 20(1-2), 27–48 (2009)
6. Gonzalez, R.C., Woods, R.E.: Digital Image Processing, 2nd edn. Prentice Hall, New Jersey (2002)
7. Gordon, R., Bender, R., Herman, G.T.: Algebraic reconstruction techniques (ART) for three-dimensional electron microscopy and x-ray photography. *Journal of Theoretical Biology* 29, 471–481 (1970)
8. Herman, G.T.: Fundamentals of Computerized Tomography: Image Reconstruction from Projections, 2nd edn. Springer, Heidelberg (2009)
9. Herman, G.T., Kuba, A. (eds.): Discrete Tomography: Foundations, Algorithms and Applications. Birkhäuser, Boston (1999)
10. Herman, G.T., Kuba, A. (eds.): Advances in Discrete Tomography and Its Applications. Birkhäuser, Boston (2007)
11. Kak, A.C., Slaney, M.: Principles of Computerized Tomographic Imaging. IEEE Service Center, Piscataway (1988)
12. Perreault, S., Héber, P.: Median filtering in constant time. *IEEE Transactions on Image Processing* 16, 2389–2394 (2007)

# 3D Curvature-Based Shape Descriptors for Face Segmentation: An Anatomical-Based Analysis

Augusto Salazar<sup>1</sup>, Alexander Cerón<sup>2,3</sup>, and Flavio Prieto<sup>3</sup>

<sup>1</sup> GTA Percepción y Control Inteligente, Universidad Nacional de Colombia - Sede Manizales, Manizales, Km 7 vía al aeropuerto la Nubia Manizales, Colombia

<sup>2</sup> Universidad Militar Nueva Granada, Carrera 11 No. 101-80 Bogotá D.C., Colombia

<sup>3</sup> Universidad Nacional de Colombia - Sede Bogotá D.C., Carrera 30 No 45 - 03 Bogotá D.C., Colombia  
[{aesalazarj,aceronco,faprietoo}@unal.edu.co](mailto:{aesalazarj,aceronco,faprietoo}@unal.edu.co)

**Abstract.** The behavior of six curvature-based 3D shape descriptors which were computed on the surface of 3D face models, is studied. The set of descriptors includes  $k_1$ ,  $k_2$ , Mean and Gaussian curvatures, Shape Index, and Curvedness. Instead of defining clusters of vertices based on the value of a given primitive surface feature, a face template composed by 28 anatomical regions, is used to segment the models and to extract the location of different landmarks and fiducial points. Vertices are grouped by: vertices themselves, region, and region boundaries. The aim of this study is to analyze the discriminant capacity of each descriptor to characterize regions and to identify key points on the facial surface. The experiment includes testing with data from synthetic face models and 3D face range images. In the results: the values, distributions, and relevance indexes of each set of vertices, were analyzed.

## 1 Introduction

In statistical modelling of faces, it is interesting to know which shape descriptors are capable of defining a point or region in the face without having further transformations (e.g., curvature-based shape descriptors). Generally, this is due to feature detection process which are intended to minimize a function which implies an iterative process where the features in every stage must be calculated [1]. The complexity of the face surface implies that a single descriptor is unable to represent the entire surface. Therefore, it is interesting to analyze the behavior of a descriptor on the face surface.

The goal of this work is to determine which of the curvature-based shape descriptors offer more information of facial features over 3D representations. Although there may be a wide range of research where feature detection is carried out based on this type of descriptors, it is uncommon to make quantitative comparisons among them. We may say that the closest work is the one presented in [2] where the Mean ( $H$ ) and Gaussian ( $K$ ) curvatures sign are used to obtain

the interest regions and to describe them by using a set of geometric descriptors such as: regions areas, relations between areas, mean of areas, average and variance of the  $H$  and  $K$ , among others. A ranking of the best performance in discriminant capacity was done; the ranking was performed by using the Fisher coefficient [3] in order to obtain the best descriptors.

In this work, instead of defining clusters of vertices based on the value of a given primitive surface feature [24], a face template composed by 28 anatomical regions, was used to segment the models and to extract the location of different landmarks and fiducial points. The template is fitted to each of the 3D face models, in order to group the vertices into different groups: the vertices themselves, those within the region, and belonging to the boundaries (i.e., those which lie on the curves that define the regions). Shape descriptors considered in this work were: Minimum, Maximum,  $H$ , and  $K$  curvatures, Shape Index and Curvedness. The values and distributions of each descriptor were analyzed.

In our case, we developed a study of relevance to measure the representation capacity of the six mentioned descriptors to determine which descriptors are better according to the location in the different regions of the face. By doing so, when developing searching strategies that include iterative processes, these can be reconfigured according to the region in the face on which we want to have the feature detected.

The organization of the paper is as follows, first we will begin talking about the 3D shape descriptors derived from curvature. Next, we will briefly introduce the discriminant analysis used. After that, we will describe the experimental setup. Finally, we will show the results of our relevance analysis and our conclusions.

## 2 3D Shape Descriptors

This work is the first stage of a system to detect specific points on the facial surface which define different facial regions, therefore, we have the problem to establish which features are relevant to identify each point. The characterization process is based on the different ways used to represent the specific points using only the available information (avoiding sophisticated transformations). Since curvature is a property of the local surface, which has the advantage to be invariant against translations and rotations, shape descriptors derived from curvature and curvature itself are used.

The curvature in the plane is defined as  $\kappa = \frac{d\alpha}{ds}$ , where  $\alpha$  is the angle formed by the vector tangent to the curve (to the direction of displacement) with a fixed direction. Some curvatures can be associated to a  $S$  surface in  $\mathbb{R}^3$ : principal curvatures  $k_1$  and  $k_2$ , mean curvature  $H = \frac{k_1+k_2}{2}$  and Gaussian curvature  $K = k_1k_2$ .

The curvature in face recognition and facial feature detection has been widely used, specially when 3D face geometry information is available. Analyzing the different curvature values, it is possible to detect mouth and eyes region [5], to segment a face model as input of a recognition system [6], or to detect several landmarks on the facial surface [7]. Other approaches combine the curvature

information along with other features obtained from 2D information [8] and/or a priori knowledge of the face geometry [9].

In a similar way, based on the principal curvature values, other descriptors such as the Shape Index  $S_I$  have been proposed [10]. For a point  $p$  on a surface,  $S_I$  is defined as  $S_I(p) = \frac{1}{2} - \frac{1}{\pi} \tan^{-1} \frac{k_1(p) + k_2(p)}{k_1(p) - k_2(p)}$ .  $S_I$  has been used to locate facial features in [11][12][13][14]. One more descriptor is the Curvedness defined as  $R(p) = \sqrt{(k_1^2(p) + k_2^2(p))/2}$ . This represents the amount of curvature in a region making possible the perception of the variation in the shape scale of the objects.  $R$  is useful in defining the criteria for automatic segmentation of triangular meshes [15].

### 3 Discriminant Analysis

Despite algorithms like Principal Components Analysis, which find components of a set of features useful for data representation, they do not allow to find the features that have the most relevant information, which is an important step before performing a classification or recognition process. Fisher's discriminant analysis finds the features that carry the most relevant information by projecting the data in a space with less overlapping within classes.

Consider a data set with  $d$  dimensions and  $n$  measures  $\mathbf{x}_1, \dots, \mathbf{x}_n$  which is composed of  $l$  classes  $C_i$ . Fisher's linear discriminant is performed with two (sub sets or) classes with  $N_1$  and  $N_2$  elements obtaining a criterion of separation.

Each class has mean  $m_i = \frac{1}{N_i} \sum_{x \in C_i} \mathbf{x}$ . The data are projected in a new space  $y = \mathbf{w}^T \mathbf{x}$ . These two projected classes have means  $\mu_1$  and  $\mu_2$  respectively by using  $\mu_i = \frac{1}{N_i} \sum_{y \in C_i} \mathbf{y}$ .

The separation between classes is obtained by finding a  $\mathbf{w}$  that maximizes  $m_2 - m_1 = \mathbf{w}^T (\mu_2 - \mu_1)$  [16], where  $m_i = \mathbf{w}^T \mu_i$ . The within-class variance of the transformed data from the class  $C_i$  is obtained from  $\sigma_k = \sum_{n \in C_i} (y_i - m_k)^2$ .

The total within-class variance for the whole data set is defined as  $\sigma_i^2 + \sigma_j^2$ . Fisher criterion is obtained as the ratio of the between-class variance to the within-class variance by using the Equation 1

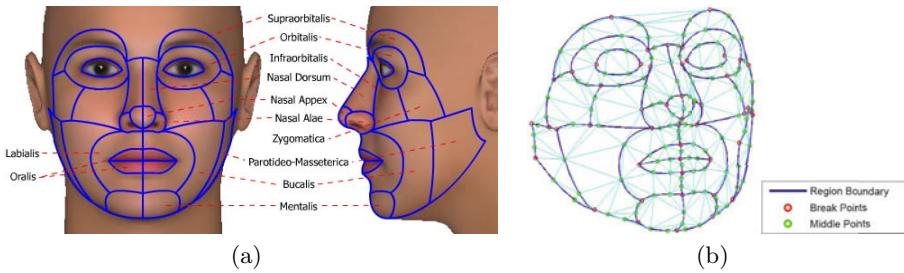
$$F_{ij} = \frac{(\mu_i - \mu_j)^2}{\sigma_i^2 + \sigma_j^2}. \quad (1)$$

Finally, Fisher coefficient is computed as the mean of all combinations of the Equation 1 evaluated for each feature.

### 4 Experimental Setup

#### 4.1 Face Template

Each 3D face model has to be segmented in 28 regions (see Figure 1(a)) which correspond to the anatomical regions in the soft tissue which are used to describe an injury in forensics and/or to plan a surgery in many other medical contexts.



**Fig. 1.** (a) Facial regions. (b) Face Template.

A region boundary is defined by using Bezier curves each one with two break points and two control points. The entire template is composed by 68 region boundaries, 46 break points, 136 middle points and 338 triangles, both break and middle points are the vertices of the 3D face template (see Figure 1(b)).

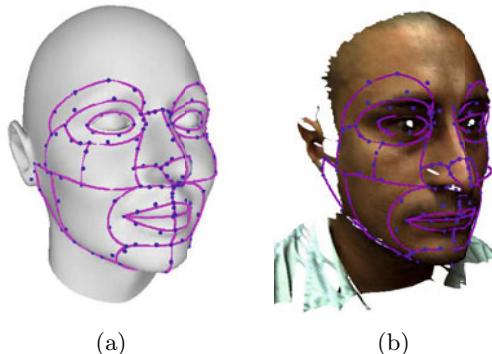
## 4.2 Databases

In this work, two databases were used: synthetic face models database ( $DB_s$ ) and range-scans face database ( $3DImDB$ ). Models in the database  $DB_s$  were generated using a commercial software. The database  $DB_s$  contains 10 models of 10 different characters, 5 female and 5 male. Different racial origins (African, Asian, European and Indian) were considered in the face generation process. Images in the database  $3DImDB$  were captured from 5 subjects using a Minolta Vivid 9i 3D digitizer. In a preliminary test for 24 points, the noise presented in the images generates shape descriptors with dispersed values. In order to improve the mesh quality a Laplacian smoothing filter stage was carried out. In order to assure the integrity of the results, the template is manually fitted on each of the face models used in this study (see Figures 2(a) and 2(b)).

## 4.3 Tests

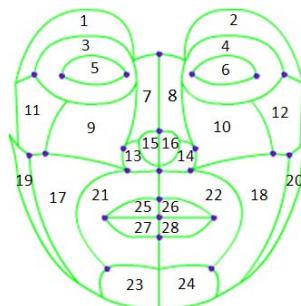
Two kinds of tests were designed. The first one to establish which descriptor is the most representative over all the set of points (global relevance). The second test is composed by several tests depending on the facial region where the points are located (local relevance). The experiments were the following:

1. Global analysis of descriptors in 182 points of face by using the face template for adjust the points on the surface of each of the models. Six Fisher's coefficients were obtained.
2. Local relevance of the descriptors computed on the contours and regions of representative areas of the face: eyes, nose, cheeks, mouth and chin. Figure 3 shows which regions are considered in each test. In this case, we carried out a discriminant analysis of the mean values of regions for each one of the regions by using the Fisher coefficient.



**Fig. 2.** (a) Fitted template ( $DB_s$ ). (b) Fitted template ( $3DImDB$ ).

<b>Test Side Regions</b>			
<i>Eyes</i>	Left	1, 3, 5	
	Right	2, 4, 6	
<i>Nose</i>	Front	7,8,13,14,15,16	
<i>Mouth</i>	Front	25,26,27,28	
<i>Chin</i>	Front	23,24	
<i>Cheek</i>	Left	9,11,17,19	
	Right	10,12,18,20	



**Fig. 3.** Regions considered for each local test

## 5 Experimental Results

### 5.1 Global Relevance

First of all, we compared the value of the descriptor at each vertex of the template against each one of the remainder points. In addition, we obtained the relevance between the mean of the descriptors of the 28 regions of the face (see Table 1).

Global test showed that the best descriptor was the  $S_I$  for the most of the cases considered and the worst was  $K$  in both datasets. For the  $DB_s$  database, some differences were obtained in the results by regions and contours, obtaining that  $R$  and  $k_2$  were the best for contours and regions, respectively. It suggest that the mean of the models has great variations for the synthetic models because they are not generated with regular distances between its vertices, which generates a concentration of descriptors values in dense areas. On the other hand, real images have meshes with more regularity and resolution; most of the descriptors does not have a remarkable difference between them in value and ranking, for this reason it is necessary to analyze the relevance by areas of the face in order to establish which descriptor or combination of them are better for each area.

**Table 1.** Fisher's coefficients for the Global test. **R** indicates the ranking.

D	DBs						3DImDB					
	Points			Contour Regions			Points			Contour Regions		
	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R
$k_1$	2,5764	<b>5</b>	11,7057	<b>4</b>	16,1132	<b>5</b>	0,9355	<b>3</b>	1,2956	<b>4</b>	4,0512	<b>2</b>
$k_2$	2,5786	<b>4</b>	19,9167	<b>2</b>	63,1980	<b>1</b>	0,7804	<b>4</b>	1,5369	<b>3</b>	2,1186	<b>5</b>
$H$	4,7612	<b>2</b>	9,5491	<b>5</b>	22,7574	<b>3</b>	0,9860	<b>2</b>	1,2336	<b>5</b>	2,1788	<b>4</b>
$K$	0,9465	<b>6</b>	4,0545	<b>6</b>	13,6120	<b>6</b>	0,3556	<b>6</b>	0,6841	<b>6</b>	1,0063	<b>6</b>
$S_I$	5,2886	<b>1</b>	14,0069	<b>3</b>	21,6360	<b>4</b>	1,7799	<b>1</b>	2,6034	<b>1</b>	4,6142	<b>1</b>
$R$	2,7021	<b>3</b>	23,3434	<b>1</b>	46,0875	<b>2</b>	0,7675	<b>5</b>	1,5952	<b>2</b>	3,9350	<b>3</b>

**Table 2.** Fisher's coefficients for the *Eyes* test

D	DBs						3DImDB <sub>L</sub>						3DImDB <sub>R</sub>					
	Contour			Region			Contour			Region			Contour			Region		
	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R
$k_1$	7,026	<b>3</b>	12,464	<b>6</b>	1,549	<b>1</b>	1,731	<b>2</b>	0,230	<b>5</b>	1,446	<b>2</b>						
$k_2$	13,635	<b>1</b>	104,646	<b>1</b>	0,424	<b>6</b>	1,292	<b>4</b>	0,767	<b>2</b>	1,073	<b>4</b>						
$H$	3,561	<b>5</b>	31,700	<b>4</b>	0,458	<b>4</b>	0,157	<b>6</b>	0,479	<b>4</b>	0,404	<b>6</b>						
$K$	1,499	<b>6</b>	34,759	<b>3</b>	0,435	<b>5</b>	0,448	<b>5</b>	0,092	<b>6</b>	0,668	<b>5</b>						
$S_I$	3,735	<b>4</b>	17,541	<b>5</b>	0,979	<b>2</b>	1,718	<b>3</b>	0,574	<b>3</b>	1,162	<b>3</b>						
$R$	11,750	<b>2</b>	53,642	<b>2</b>	0,889	<b>3</b>	2,194	<b>1</b>	0,808	<b>1</b>	1,630	<b>1</b>						

## 5.2 Eye Area Test

In this test the face side variable was included. The results are shown in Table 2 (subscripts  $L$  and  $R$  correspond to the left and right eyes respectively). For the 3DImDB database, a remarkable difference between the rankings of the contours of both sides was observed. This is due to the contours of the eye area present great shape variation, can be asymmetric, and the points are not enough or could be located in different places on both sides of face. Also, the ranking of contours and regions were very different. It implies that contours and regions should be analyzed separately. Regarding to the regions, we obtained that both side regions share the same ranking, this shows that in regions with high detail, in order to describe the shape properly, a big amount of points is required.

## 5.3 Nose Area Test

The values obtained in this test were higher than the ones of the other regions (see Table 3). Results were similar for both datasets. Despite the shape of the

**Table 3.** Fisher's coefficients for the *Nose* test

D	DBs				3DImDB			
	Contour		Region		Contour		Region	
	Value	R	Value	R	Value	R	Value	R
$k_1$	20,439	<b>2</b>	28,721	<b>2</b>	2,033	<b>2</b>	3,935	<b>2</b>
$k_2$	3,853	<b>6</b>	19,389	<b>4</b>	1,063	<b>4</b>	1,712	<b>5</b>
$H$	11,629	<b>3</b>	27,992	<b>3</b>	1,882	<b>3</b>	2,310	<b>4</b>
$K$	7,905	<b>4</b>	12,168	<b>5</b>	0,618	<b>5</b>	0,602	<b>6</b>
$S_I$	26,144	<b>1</b>	41,916	<b>1</b>	7,287	<b>1</b>	8,002	<b>1</b>
$R$	6,459	<b>5</b>	11,621	<b>6</b>	0,446	<b>6</b>	3,082	<b>3</b>

nose varies considerably from a subject to other, shape variation occurs in a low scale, then, the differences between the resolution of the datasets prevent that subtle variations could be compared between them.

#### 5.4 Cheeks Area Test

Despite the surfaces in the cheeks are soft, values of the Fisher's indexes were high. Contrary to the eyes test, contours of both sides shared almost the same ranking, and the ranking of the regions are different. Difference in the values shows that the asymmetry affects the capacity of representation of the descriptors. In this case, variations of shape occurs in a great scale. Therefore, for real data, the descriptors were able to characterize the morphology changes in both sides of the face, which was not possible in the nose region.

**Table 4.** Fisher's coefficients for the *Cheeks* test

D	DBs				3DImDB <sub>L</sub>				3DImDB <sub>R</sub>			
	Contour		Region		Contour		Region		Contour		Region	
	Value	R	Value	R	Value	R	Value	R	Value	R	Value	R
$k_1$	31,574	<b>2</b>	23,886	<b>2</b>	1,670	<b>2</b>	3,197	<b>1</b>	3,399	<b>2</b>	12,594	<b>1</b>
$k_2$	0,227	<b>6</b>	1,005	<b>6</b>	0,213	<b>6</b>	0,214	<b>6</b>	0,010	<b>6</b>	0,016	<b>6</b>
$H$	26,886	<b>3</b>	23,700	<b>3</b>	1,373	<b>4</b>	1,166	<b>5</b>	3,037	<b>3</b>	2,230	<b>3</b>
$K$	1,838	<b>5</b>	2,288	<b>5</b>	0,957	<b>5</b>	1,224	<b>4</b>	1,486	<b>5</b>	0,784	<b>5</b>
$S_I$	39,157	<b>1</b>	34,222	<b>1</b>	3,422	<b>1</b>	3,175	<b>2</b>	4,664	<b>1</b>	6,284	<b>2</b>
$R$	17,174	<b>4</b>	14,474	<b>4</b>	1,391	<b>3</b>	1,234	<b>3</b>	1,500	<b>4</b>	1,176	<b>4</b>

#### 5.5 Mouth Area Test

The values obtained in this test were the lowest (see Table 5). As we can expect, a great difference between the ranking of the contours and regions was obtained,

**Table 5.** Fisher's coefficients for the *Mouth* test

D	DBs				3DImDB			
	Contour		Region		Contour		Region	
	Value	R	Value	R	Value	R	Value	R
$k_1$	0,182	<b>4</b>	0,061	<b>6</b>	0,014	<b>5</b>	0,071	<b>1</b>
$k_2$	0,248	<b>3</b>	0,141	<b>3</b>	0,027	<b>3</b>	0,004	<b>6</b>
$H$	0,154	<b>5</b>	0,130	<b>4</b>	0,010	<b>6</b>	0,017	<b>4</b>
$K$	0,146	<b>6</b>	0,076	<b>5</b>	0,056	<b>1</b>	0,053	<b>2</b>
$S_I$	0,294	<b>2</b>	1,198	<b>1</b>	0,018	<b>4</b>	0,016	<b>5</b>
$R$	0,300	<b>1</b>	0,153	<b>2</b>	0,043	<b>2</b>	0,043	<b>3</b>

this is because the contours are located in the place where the surface changes its orientation, then, the nature of the information from the vertices of the region and contours is very different. Another reason is that the changes in the surface of the mouth area are soft (except in the mouth corners) making the task of characterization more difficult than in other regions of the face.

## 5.6 Chin Area Test

The results of this test are shown in Table 6. As in the mouth test, the situation regarding to the values and rankings were similar, which shows that for the kind of surfaces present in the mouth and chin areas, the six curvature-based shape descriptors are not able to describe the morphology properly.

**Table 6.** Fisher's coefficients for the *Chin* test

D	DBs				3DImDB			
	Contour		Region		Contour		Region	
	Value	R	Value	R	Value	R	Value	R
$k_1$	0,026	<b>4</b>	0,005	<b>4</b>	0,001	<b>6</b>	0,375	<b>3</b>
$k_2$	0,066	<b>2</b>	0,047	<b>1</b>	0,318	<b>2</b>	0,557	<b>2</b>
$H$	0,077	<b>1</b>	0,027	<b>3</b>	0,394	<b>1</b>	0,123	<b>5</b>
$K$	0,010	<b>6</b>	0,005	<b>5</b>	0,032	<b>4</b>	0,982	<b>1</b>
$S_I$	0,021	<b>5</b>	0,000	<b>6</b>	0,005	<b>5</b>	0,102	<b>6</b>
$R$	0,031	<b>3</b>	0,039	<b>2</b>	0,082	<b>3</b>	0,151	<b>4</b>

## 6 Conclusions

An analysis of the relevance of six curvature-based 3D shape descriptors on points, contours, regions, areas and sides of the face was carried out. Based on

the Fishers analysis, we showed how the morphology of the face surface influences the capacity of representation of the descriptors. From this, it was determined which descriptors are more appropriate to characterize each of the major areas that define the face.

In tests where laterality was included as a variable, it was shown that the curvature-based descriptors are suitable to capture changes due to the natural asymmetry of the face. However, in order to characterize such variation, an analysis should be performed on a comprehensive database. Also, it should be noted that depending on the area to be assessed, an analysis at a higher or lower resolution (e.g., regions of the nose and mouth) is required. We emphasize these results because in most of the works that discuss about the asymmetry, analysis are qualitative, by contrast, this paper shows in a quantitative manner, how a descriptor increase or decrease its ability to represent a region or contour of the same area but different face side.

In the regions belonging to the mouth and chin, the descriptors studied did not show a difference in their levels of relevance, therefore, it is necessary to conduct a study which includes other kind of descriptors. In any case, it should be noted that in order to identify those regions, the search strategy should consider the low variability of the surfaces.

With the information collected from this study, it is possible to design a feature detection system by taking into account aspects such as asymmetry, distribution of points, morphology of the facial regions, among others. Similarly, this work can be extended to other parts of the body and its results applied in anthropometric studies.

## Acknowledgment

This work was supported by the programs “créditos condonables para estudiantes de Doctorado convocatoria 2007” from *Departamento Administrativo de Ciencia, Tecnología e Innovación - COLCIENCIAS* and “Convocatoria de apoyo a tesis de posgrado - DIMA 2010 - DOCTORADOS” from *Dirección de Investigaciones de Manizales*. The authors want to thank the MeshLab developers (<http://meshlab.sourceforge.net/>) who provided them with an excellent tool which helped them save valuable time.

## References

1. Salazar, A.E., Prieto, F.A.: 3d bsm for face segmentation and landmarks detection. In: Baskurt, A.M. (ed.) Three-Dimensional Image Processing (3DIP) and Applications, vol. 7526, p. 752608 (2010)
2. Díaz, A.B.M.: Reconocimiento Facial Automático mediante Técnicas de Visión Tridimensional. PhD thesis, Universidad Politécnica de Madrid, Facultad de Informática (2004)
3. Duda, R.O., Hart, P.E., Stork, D.G.: Pattern Classification and Scene Analysis. John Wiley & Sons, New York (1998)

4. Gatzke, T., Grimm, C.: Feature detection using curvature maps and the min-cut/max-flow algorithm. In: Kim, M.-S., Shimada, K. (eds.) GMP 2006. LNCS, vol. 4077, pp. 578–584. Springer, Heidelberg (2006)
5. Colombo, A., Cusano, C., Schettini, R.: 3d face detection using curvature analysis. *Pattern Recognition* 39, 444–455 (2006)
6. Hallinan, P.W., Gordon, G.G., Yuille, A.L., Giblin, P., Mumford, D.: Two-and Three-dimensional patterns of the face. A. K. Peters, Ltd., Wellesley (1999)
7. Deo, D., Sen, D.: Automatic recognition of facial features and land-marking of digital human head. In: 6th International Conference on Computer Aided Industrial Design and Conceptual Design, pp. 506–602 (2005)
8. Xue, F., Ding, X.: 3d+2d face localization using boosting in multi-modal feature space. In: 18th International Conference on Pattern Recognition, ICPR 2006 (2006)
9. Sun, Y., Yin, L.: Automatic pose estimation of 3d facial models. In: 19th International Conference on Pattern Recognition, ICPR 2008, pp. 1–4 (2008)
10. Koenderink, J.J., Van Doorn, A.J.: Surface shape and curvature scales. *Image and Vision Computing* 8, 557–564 (1992)
11. Lu, X., Colbry, D., Jain, A.K.: Three-dimensional model based face recognition. In: 17th International Conference on Pattern Recognition, vol. 1, pp. 362–366 (2004)
12. Colbry, D., Stockman, G., Jain, A.K.: Detection of anchor points for 3d face verification. In: IEEE Workshop on Advanced 3D Imaging for Safety and Security (2005)
13. Lu, X., Colbry, D., Jain, A.K.: Matching 2.5d scans to 3d models. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 28, 31–43 (2006)
14. Guangpeng, Z., Yunhong, W.: A 3d facial feature point localization method based on statistical shape model. In: IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2007, vol. 2, pp. II-249–II-252 (2007)
15. Jagannathan, A., Miller, E.L.: Three-dimensional surface mesh segmentation using curvedness-based region growing approach. *IEEE Transactions Pattern Analysis and Machine Intelligence* 29, 2195–2204 (2007)
16. Bishop, C.: Pattern Recognition and Machine Learning. Springer Science Business + Media, LLC (2006)

# Computational Hemodynamics in Intracranial Vessels Reconstructed from Biplane Angiograms

Fabien Scalzo<sup>1,2</sup>, Qing Hao<sup>1</sup>, Alan M. Walczak<sup>3</sup>, Xiao Hu<sup>2</sup>, Yiemeng Hoi<sup>4</sup>, Kenneth R. Hoffmann<sup>5</sup>, and David S. Liebeskind<sup>1</sup>

<sup>1</sup> Dept. of Neurology, University of California, Los Angeles (UCLA)

<sup>2</sup> Dept. of Neurosurgery, NSDL, University of California, Los Angeles (UCLA)

<sup>3</sup> Imagination Software Corporation, Buffalo, New-York

<sup>4</sup> Dept. of Mechanical and Industrial Engineering, University of Toronto, Toronto

<sup>5</sup> Dept. of Neurosurgery, University at Buffalo, New-York

**Abstract.** Recent works in neurology have explored ways to obtain a better understanding of blood flow circulation in the brain with the ultimate goal of improving the treatment of cerebrovascular diseases, such as strokes, stenosis, and aneurysms. In this paper, we propose a framework to reconstruct three-dimensional (3D) models of intracerebral vessels from biplane angiograms. The reconstructed vessel geometries are then used to perform simulations of computational fluid dynamic (CFD). A key component of our framework is to perform such a reconstruction by incorporating user interaction to identify the centerline of the vessels in each view. Then the vessel profile is estimated automatically at each point along the centerlines, and an optimization procedure refines the 3D model using epipolar constraints and back-projection in the original angiograms. Finally, the 3D model of the vessels is then used as the domain where the wall shear stress (WSS), and velocity vectors are estimated from a blood flow model that follows Navier-Stokes equations as an incompressible Newtonian fluid. Visualization of hemodynamic parameters are illustrated on two stroke patients.

## 1 Introduction

The quantitative assessment of blood flow in the brain is essential for an effective treatment of patients admitted for cerebrovascular diseases, such as stroke, stenosis, and aneurysm. A correct understanding of the underlying dynamics of blood flow would have fundamental implications in the management of these cerebrovascular diseases.

Because noninvasive measurement techniques of the flow dynamic around regions of interest in intracerebral vessels remains challenging in clinical practice, most research groups perform computational fluid dynamic (CFD) simulations to predict the local flow environment. This methodology is very promising and potentially leads to unique insights into the treatment of cerebrovascular pathophysiologies.

Recent works about CFD in intracranial vessels usually rely on three-dimensional (3D) images acquired using computed tomographic angiography (CTA) [1], magnetic resonance angiography (MRA) [2], or 3D rotational angiography (3DRA) [3]. Although these modalities provide the 3D geometry of intracranial arteries, they have several intrinsic limitations [3]. CTA, which is frequently used for stroke patients, usually consists

of an image of resolution  $512 \times 512$ ,  $0.23 - 0.45$  mm pixel size, and  $0.5 - 1.3$  mm slice thickness. Its main weaknesses are that it may not be able to detect small arteries, it is subject to interpolation errors, and it may be affected by other anatomical structures, such as bones. Unlike CTA, MRA has the advantage that the bone does not disturb the image, but it has a lower spatial resolution of a  $256 \times 256$ ,  $0.78 - 1.25$  mm and slice thickness  $0.7 - 1.6$  mm. MRA requires longer acquisition times and is therefore more sensitive to patient motion. It may also underestimate the size of the vessels due to signal loss in regions of low or complex flow activities. Recently, there has been increasing interest in utilizing the minimally invasive 3DRA which, compared to CTA and MRA, provides a higher spatial resolution  $1283 - 5123$ , voxel size of  $0.42 - 1$  mm, and lower sensitivity to patient motion.

An attractive alternative to these images is the digital subtraction angiography (DSA), or angiogram, that is usually considered as the best imaging technique to observe blood flow in arteries. This technique uses a contrast agent that is injected via a catheter inserted into the femoral artery and strategically navigated to the carotid artery. Because DSA subtracts out images of anatomical structures other than contrast-filled blood vessels, it makes the blood vessels more distinguishable. The main drawback of angiograms is that, unlike CTA and MRA, they are made of two-dimensional projected images that lack of the depth information required to reconstruct the 3D models intended for CFD studies. To overcome this difficulty, we propose, as it has been done for coronary arteries [4][5][6][7][8], to perform 3D reconstruction of intracerebral arteries from two-dimensional biplane angiograms. Despite the popularity of angiograms in the clinical world, the automatic 3D reconstruction of intracerebral vessels from two orthogonal views is a problem that is still largely unresolved [9][10][11][12][13][14]. The proposed framework reconstructs the cerebral arteries using the following steps; vessel centerlines are first estimated from user annotations and the diameter is estimated by automatic evaluation of the intensity profile on each point along the centerline, then the 3D geometry is estimated using an optimization algorithm that minimizes the error measured using back-projection of the vessels in the original images. A volumetric mesh is generated using the reconstructed geometry that, in turn, is used during CFD simulations. This leads to unique quantitative estimation of flow parameters in the brain from biplane angiograms.

## 2 Data Acquisition

The data used to evaluate our framework originate from two patients treated at the University of California Los Angeles (UCLA) Medical Center for cerebrovascular diseases; one had a stroke located in the proximal middle cerebral artery (MCA) territory, and the other patient was also treated for a stroke but the occlusion was more distal, and its MCA exhibited normal flow. To illustrate our framework, flow parameters will be compared for these two patients around the junction of the internal carotid artery (ICA) and the middle cerebral artery (MCA). The use of these data was approved by the local institutional review boards.

Source images were acquired using 2D DSA. Each run is made of a sequence of 20 frames, and each frame has a resolution of  $1024 \times 1024$  pixels. The orthogonal

angiograms were acquired in an interleaved fashion; one frontal, one sagittal, and so on. Note that this configuration is actually more challenging for the 3D correspondence task during reconstruction, in comparison with two oblique views.

### 3 Vessel Reconstruction from Biplane Angiograms

#### 3.1 Background

The 3D reconstruction of intracerebral vessels from two orthogonal views addresses an important and challenging clinical problem. The arterial tree in the brain contains tortuous vessels that often overlap when seen from a single view. There are therefore a number of ambiguities that make it difficult to establish point correspondence between the views.

Several groups [9][10][11][12][13][14] have attempted to develop frameworks for the automatic 3D angiographic reconstruction of the intracranial vasculature. Unfortunately, only a few of them have obtained satisfying results, and because the works were usually evaluated on artificial datasets, or on very few patients often with ideal conditions; the results are not satisfying in real clinical conditions. Automatic 3D reconstruction of complex intracerebral arterial trees from two biplane angiograms remains largely unresolved.

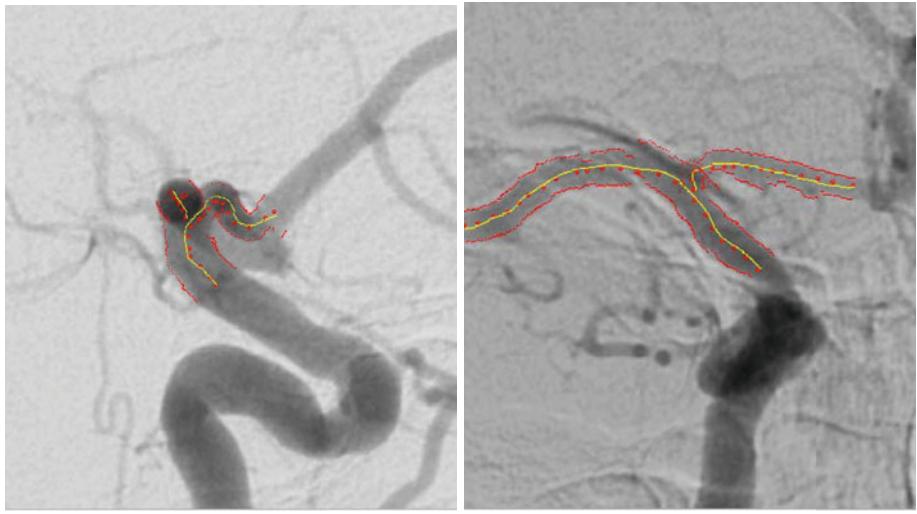
Interestingly, the problem becomes more feasible by using user interaction [15]. Basically, if the user is able to select a few points along the centerline of the vessels intended to be reconstructed in each view, the correspondence problem becomes easier. Such an approach potentially leads to a more precise reconstruction and can work in clinical conditions. Our framework, which is described in the next sections, is based on this key feature.

#### 3.2 Vessel Centerline Extraction

The first task for the user is to select the image, in each of the two angiogram sequences, that is the best to identify the vessels intended to be reconstructed. Given these two DSA images, an approximate vessel centerline is indicated by the user as a series of points. The indicated points are then fitted with a cubic spline. Using interpolation of the spline model, a new set of points is extracted and centered within the vessel cross-section based on measured vessel edges (Section 3.3) to generate a more refined centerline to be used for analysis (Figure 1).

#### 3.3 Vessel Sizing and Edge Extraction

Using the two-dimensional images and the vessel centerlines as input, the entire vessel region along the centerline is segmented. At each point along the vessel centerline, a line centered on that point is generated perpendicular to the tangent of the centerline (Figure 2). Along this perpendicular line, the pixel values are extracted from the image and the vessel profile is generated using bilinear interpolation. The vessel region of the vessel profile is identified and fitted with an elliptical model [16]. From this elliptical



(a) Sagittal View

(b) Frontal View

**Fig. 1.** Illustration of the two orthogonal angiograms used to reconstruct the normal case. Center-lines are depicted by yellow lines, the boundary of the vessels is shown as red lines, and the red points were specified by the user. The results of computational fluid dynamics (CFD) applied on the reconstructed model are shown in Figure 3.

model, the vessel edges, size, contrast, and center in the image are determined for each point of the vessel independently.

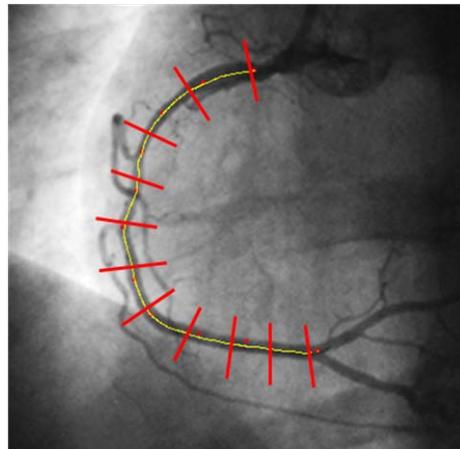
Once this analysis has been performed on each point along the vessel centerline, the parameters of the entire vessel segment are linearly fitted (to model linear tapering) to estimate the parameters of the vessel edges, sizes, and contrasts along the vessel segment.

### 3.4 Determination of the Geometric Transformation Relating the Two Imaging Systems

Each projection imaging system (*i.e.* gantry), consisting of a x-ray source and detector, has its own coordinate system, and this coordinate system moves with the respective gantry. To reconstruct a 3D model of the surface from two projection images, the geometric transformation relating the two gantry systems needs to be determined to establish a common coordinate system. The relationship between the two systems can be represented by the following equation,

$$x_0 = R(x - t) \quad (1)$$

where  $x_0 \in \mathbb{R}^3$  and  $x \in \mathbb{R}^3$  are corresponding coordinates in the two respective imaging systems, and  $R$  and  $t$  are the rotation matrix and translation vector of the transformation relating the two coordinate systems. The parameters  $R$  and  $t$  of the transformation are approximated using the gantry information (angles, magnification, and source-to-image



**Fig. 2.** The vessel is segmented by extracting a set of lines perpendicular to the centerline. The pixel intensity along each line are used to generating a vessel profile. The vessel region of the vessel profile is identified and fitted with an elliptical model.

distance) usually available in the image header. Once the transformation is estimated, the corresponding points along the two centerlines are determined using epipolar constraints as explained below.

A point  $\alpha$  along the vessel centerline in one image is selected. A line is generated which connects this point with the x-ray source for that projection (*i.e.* projection line). The transformation relating the two imaging systems (Eq. ⑪) is used to calculate the coordinates of this projection line in the other imaging system and to project it into the other image. The intersection of this projected line with the centerline in the second image is taken as the point  $\alpha_0$  which corresponds to the point  $\alpha$  in the first image. Lines from each pair of corresponding image points are projected back to the respective sources and the intersection of these lines is taken as the 3D point (corresponding to the two points  $\alpha_0, \alpha$  in the images). This procedure is repeated for each point along the centerline, thus producing an initial guess of the 3D vessel centerline.

Unfortunately, the estimated imaging geometry is not always accurate. As a result, the estimated correspondences may be incorrect and yield incorrect 3D centerlines. The quality of the reconstruction can be seen and quantified by comparing the indicated centerlines and the projections of the reconstructed 3D centerlines in the respective images. The accuracy of the estimated 3D centerlines is improved using an optimization procedure (based on the simplex algorithm) that iteratively changes  $R$  and  $t$  (then reconstructs the 3D centerline and reprojects the 3D centerline), so as to minimize the difference between the 2D manually indicated centerlines and the projection of the reconstructed 3D centerline. The transformation of the 3D centerlines yielding the best agreement, in terms of average error, is selected.

### 3.5 Reconstruction

The full reconstructed 3D centerline and vessel size data are then used to determine the dimensions of the volume in which the final full reconstruction of the vessel will be placed. A tapered cylindrical tube is generated and centered on the 3D centerline. The local diameter of the tube corresponds to the local size of the vessel lumen corrected for magnification. The 3D surface can then be rendered by visualization software (Figure 3) and analyzed.

## 4 Computational Fluid Dynamics (CFD)

### 4.1 Mesh Generation

The 3D surface reconstructed using the technique presented in the previous section is then meshed using the commercial software ICEM-CFD developed by ANSYS, INC. To obtain mesh-independent flow estimation, we increased the number of elements in the volume of the reconstructed vessels until the solution did not vary significantly with element size. After processing, both models were made of approximately 3 millions tetrahedral volume elements (Figure 3).

### 4.2 Blood Flow Model

Blood flow was modeled as an incompressible Newtonian fluid described by the Navier-Stokes equations [17] in 3D,

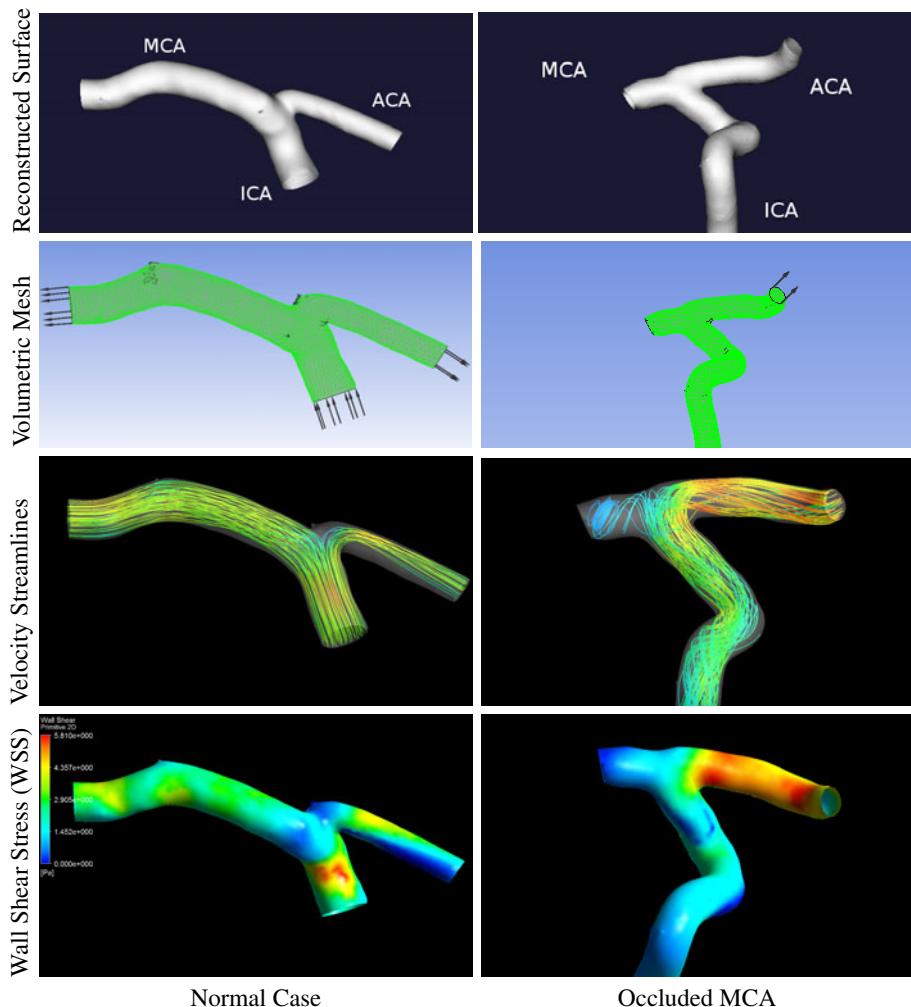
$$\nabla \cdot v = 0 \quad (2)$$

$$\rho \left( \frac{\partial v}{\partial t} + v \cdot \nabla v \right) = -\nabla p + \mu \nabla^2 v \quad (3)$$

where  $\rho$  is the density,  $\mu$  is the viscosity,  $p$  the pressure and  $v$  the velocity. For blood  $\rho = 1.105 \text{ g/cm}^3$ , and  $\mu = 0.04 \text{ Pa s}$ .

For both models, steady-state condition was assumed. The flow rates at the inlet of the models were both set to  $5 \text{ ml s}^{-1}$ . For simplicity, we assumed that blood behaves as a Newtonian-fluid, although such assumption requires further validation in the reconstructed arteries (ICA, MCA, ACA). We assumed rigid wall and no slip boundary conditions at the walls. A uniform velocity profile was imposed at the inlet, and the outlet had a traction free boundary condition.

Numeric solutions of the Navier-Stokes equations were obtained with the use of a fully implicit finite element formulation that allows arbitrary time-step sizes (convergence was obtained in less than 100 iterations). Simulations were performed using CFX (ANSYS, INC). Each simulation was ran in a distributed fashion on a CRAY CX1 cluster made of five bladed cadenced with two 2.26GH processors each. The results were obtained within a few minutes, depending on the number of cells in the mesh.



**Fig. 3.** Illustration of the 3D Reconstructed Surface, Volumetric Mesh, Velocity streamlines, and Wall Shear Stress (WSS) under normal (left) and occluded (right) MCA conditions

#### 4.3 CFD Results and Post-processing

CFD analysis was successfully performed on two patient-specific geometries. One of the cases presented a proximal MCA occlusion and the other shows normal MCA flow. We focused on the junction between the ICA, MCA, and ACA.

The simulation results of CFX were post-processed in the software ANSYS CFX-POST. Upstream velocity streamlines were generated for both cases using 75 sources from the inlet. Note that for the proximal MCA occlusion, only one outlet was available. Wall shear stress (WSS) was plotted on the wall.

As shown in Figure 3, presence of MCA occlusion resulted in a significant increase in flow velocity compared to the normal MCA flow model. In addition, dramatic changes in WSS occur for the occluded case right after the occluded site (for the MCA-occluded case). In contrast to uniform streamlines observed in the normal case, we observed complex flow patterns at the ICA, MCA, ACA junctions in the occluded model.

## 5 Conclusion

Computational hemodynamic in intracerebral arteries has important implications in the management of cerebrovascular diseases. Although biplane DSA is known to be the gold standard for flow assessment in the brain, exploiting this modality into CFD simulations is challenging as the 3D vascular geometry is not trivially obtained from biplane angiograms.

We have proposed a framework to do such a reconstruction using user-defined centerlines in each view. With the reconstructed geometries, we were able to perform CFD simulations around ICA, MCA, ACA junctions on two patient-specific cases.

Although the reconstructed geometry and CFD results require further validation, and need to be applied to a larger number of cases, the proposed framework opens the door to provide quantitative estimation of flow parameters from two orthogonal biplane angiograms, and potentially applicable to other cerebrovascular diseases. We anticipate that the current technology leads to new insights in the treatment of cerebrovascular diseases.

## Acknowledgments

This work was supported by the National Institutes of Health [K23 NS054084 and P50 NS044378 to D.S.L.], [R21-NS055998, R21-NS055045, R21-NS059797, R01 NS054881 to X.H.], and [R01-HL52567 to K.R.H.].

## References

1. Cebral, J.R., Hernandez, M., Frangi, A.F.: Computational analysis of blood flow dynamics in cerebral aneurysms from CTA and 3D rotational angiography image data. In: ICCB (2003)
2. Deschamps, T., Schwartz, P., Trebotich, D., Colella, P., Saloner, D., Malladi, R.: Vessel segmentation and blood flow simulation using Level-Sets and Embedded Boundary methods. In: CARS, vol. 1268, pp. 75–80 (2004)
3. Chang, H.H., Duckwiler, G.R., Valentino, D.J., Chu, W.C.: Computer-assisted extraction of intracranial aneurysms on 3d rotational angiograms for computational fluid dynamics modeling. Med. Phys. 36, 5612–5621 (2009)
4. Quatember, B., Mühlthaler, H.: Generation of CFD meshes from biplane angiograms: an example of image-based mesh generation and simulation. Appl. Numer. Math. 46, 379–397 (2003)
5. Corney, S., Johnston, P.R., Kilpatrick, D.: Construction of realistic branched, three-dimensional arteries suitable for computational modelling of flow. Med. Biol. Eng. Comput. 42, 660–668 (2004)

6. Wahle, A.: Quantification of coronary hemodynamics and plaque morphology using X-ray angiography and intravascular ultrasound. In: CARS, vol. 1268, pp. 1035–1039 (2004)
7. Platzer, E., Deinzer, F., Paulus, D., Denzler, J.: 3D Blood Flow Reconstruction from 2D Angiograms. *Bildverarbeitung für die Medizin*, 288–292 (2008)
8. De Santis, G., Mortier, P., De Beule, M., Segers, P., Verdonck, P., Verhegge, B.: Patient-specific computational fluid dynamics: structured mesh generation from coronary angiography. *Med. Biol. Eng. Comput.* 48, 371–380 (2010)
9. Smets, C., van de Werf, F., Suetens, P., Oosterlinck, A.: An expert system for the labeling and 3d reconstruction of the coronary arteries from two projections. *Int. J. Card Imaging* 5, 145–154 (1990)
10. Henri, C., Collins, D., Peters, T.: Multimodality image integration for stereotactic surgical planning. *Med. Phys.* 18, 167–177 (1991)
11. Delaere, D., Smets, C., Suetens, P., Marchal, G., Van de Werf, F.: Knowledge-based system for the three-dimensional reconstruction of blood vessels from two angiographic projections. *Med. Biol. Eng. Comput.* 27, 27–36 (1991)
12. Bullitt, E., Soltys, M., Chen, J., Rosenman, J., Pizer, S.: Three-dimensional reconstruction of intracranial vessels from biplane projection views. *J. Neurosci. Methods* 66, 13–22 (1996)
13. Zifan, A., Liatsis, P., Kantartzis, P., Gavaises, M., Karcanias, N., Katritsis, D.: Automatic 3-D reconstruction of coronary artery centerlines from monoplane X-ray angiogram images. *Int. J. Biol. Med. Sci.* 1, 44–49 (2008)
14. Zheng, S., Meiying, T., Jian, S.: Sequential reconstruction of vessel skeletons from x-ray coronary angiographic sequences. *Comput. Med. Imag. Grap.* 34, 333–345 (2010)
15. Hoffmann, K., Sen, A., Lan, L., Chua, K., Esthappan, J., Mazzucco, M.: A system for determination of 3D vessel tree centerlines from biplane images. *Int. J. Card Imaging* 16, 315–330 (2000)
16. Gopal, A., Hoffmann, K., Rudin, S., Bednarek, D.: Reconstruction of asymmetric vessel lumen from two views. In: Medical Imaging, vol. 4684, pp. 257–265 (2002)
17. Batchelor, G.K.: An introduction to fluid dynamics. Cambridge University Press, Cambridge (1967)

# Object Distance Estimation Based on Stereo Vision and Color Segmentation with Region Matching

Guangming Xiong<sup>1</sup>, Xin Li<sup>1</sup>, Junqiang Xi<sup>1</sup>, Spencer G. Fowers<sup>2</sup>, and Huiyan Chen<sup>1</sup>

<sup>1</sup> School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China

<sup>2</sup> Dept. of Electrical and Computer Engineering, Brigham Young University,  
Provo, Utah, USA

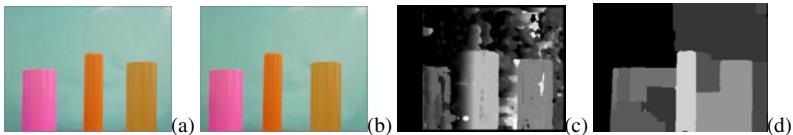
**Abstract.** Human vision system relies on stereovision to determine object distance in the 3-D world. Human vision system achieves this by first locating the objects, then matching the corresponding objects seen by the left and right eyes, and finally using triangulation to estimate the object distance. Inspired by the same concept, this paper presents a depth estimation method based on stereo vision and color segmentation with region matching in CIE Lab color space. Firstly, an automatic seeded region growing approach for color segmentation in perceptually uniform color space was proposed. Then color region matching method was implemented after color segmentation. Thereafter, 3D reprojection method was employed to calculate depth distances. Experimental results are included to validate the proposed concept for object distance estimation.

## 1 Introduction

Objects distance estimation using image processing mainly include single-camera-based and stereovision-based methods. The stereovision-based approaches have the advantage of directly estimating the 3-D coordinates of an image feature [1]. Many stereovision-based object distance estimation algorithms have been proposed to extract 3-D information and generate the disparity map [2,3,4,5].

Almost all stereovision-based approaches need to find correspondences between the left and right images. Local methods are the simplest stereo correspondence techniques that can be thought of as using a winner-take-all optimization approach, with the similarity measure as the only comparison criteria. In contrast, recent research in stereovision has centered on global methods, which usually employ a much more complex selection process, such as graph cuts [6] or Bayesian belief propagation [7].

Many algorithms use a disparity map to determine whether or not there are objects in front of the cameras and what their distances from the cameras are. For many applications such as 3-D shape matching, robot guidance, an accurate disparity map is critical to calculating the depth. However, in some cases, it is difficult to get an accurate disparity map directly using local methods or even global methods. Figs.1 (a) and (b) show the left and right images of three color columns in front of a fairly uniform background. In this case, which is very common for robot grasping applications, the disparity maps using a simple correspondence method called sum of



**Fig. 1.** A stereo pair of images (a) and (b) and two resulting disparity maps using (c) SAD and (d) graph cuts

absolute differences (SAD) and a global method based on graph cuts are shown in Figs. 1(c) and (d). Due to the so-called blank wall problem, the disparity maps cannot be used to estimate object distances correctly.

The key problem of object detection such as the example shown in Fig. 1 is how to segment the objects from the background and capture the object disparities. A simple object detection approach based on stereovision using object segmentation and region matching for grayscale images was proposed in [8]. This type of segmentation-based techniques for solving the stereo correspondence problem has gained attention in recent years [9,10,11]. In some cases, color information significantly improves segmentation result up to 20% to 25% [12].

In this paper, we propose an object distance estimation method based on stereovision and color segmentation with region matching in CIE Lab color space. First, an automatic seeded region growing approach for color segmentation in perceptually uniform color space is proposed to isolate objects from the background. Then color region matching method is implemented to determine matching objects in the left and right images. Finally, 3-D reprojection method is employed to estimate the object distance.

## 2 Automatic Seeded Region Growing in Perceptually Uniform Color Space

Deng et al. proposed an unsupervised image segmentation method called JSEG which separated the segmentation process into two stages: color quantization and spatial segmentation [13]. However, the result often suffered from over segmentation because of the drawbacks in the design of J measure for boundary detection. Fan et al. [14] presented an automatic color image segmentation algorithm by integrating color edge extraction and seeded region growing in the YUV color space. The weakness of this method was that it often generated redundant seeds. Shih et al. [15] proposed an automatic seeded region growing algorithm (ASRG) for color image segmentation. Their experiments showed that for various applications this method outperformed methods presented in [13] and [14].

Color segmentation in perceptually uniform color spaces is an ongoing research area in image processing. Paschos et al. proposed an evaluation methodology for analyzing the performance of various color spaces for color-texture analysis methods such as segmentation and classification [16]. The use of uniform color spaces was found to be suitable for the calculation of color difference using the Euclidean distance, employed in many segmentation algorithms. In this work, we propose to

adapt Shih's method in CIE Lab color space and to confirm that the color difference formula in CIE Lab color space is suitable for this segmentation algorithm.

## 2.1 Perceptually Uniform Color Space

A color space is considered a perceptually uniform color space if the Euclidean distance between two color points closely corresponds to the perceptual difference determined by the human vision system [17]. In 1976, the CIE recommended to use the CIE Lab as an approximately uniform color space and use Euclidean distance  $\Delta E_{76}$  to represent the color difference.

$$\Delta E_{76} = \|v_i - v_j\| \quad (1)$$

where,  $\|\cdot\|$  means L2-norm,  $v_i = [L_i, a_i, b_i]$ ,  $v_j = [L_j, a_j, b_j]$

## 2.2 Automatic Seeded Region Growing in CIE Lab Space

### 1) Automatic seed selection

Two conditions are used to conduct automatic seed selection in an original image [15]. In this work, we adapt Equation (1) for condition 2 to measure color difference. A seed pixel candidate  $j$  must have the maximum relative Euclidean distance  $d_{max}$  to its eight neighbors less than a threshold value, where

$$d_{max} = \max_{i=1}^8 (d_i) \quad (2)$$

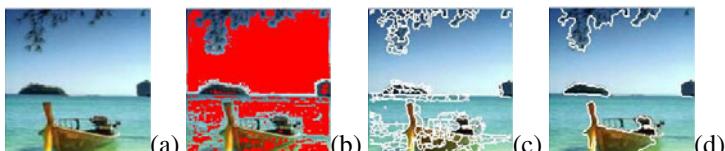
and

$$d_i = \frac{\|v_i - v_j\|}{\sqrt{L_j^2 + a_j^2 + b_j^2}}, i=1,2,\dots,8 \quad (3)$$

Figure 2(a) is an original image used in [15], the result using new automatic seed selection is shown in Figure 2(b) in which the red areas are the seed regions using the new color difference measure.

### 2) Seeds growing and region merging

Similarly, we modified the relative Euclidean distance  $d_j$  between the pixel  $j$  and its adjacent seed regions and the relative color difference  $d(R_i, R_j)$  between two adjacent region  $R_i$  and  $R_j$  in the steps of seeds growing and region merging in [15]. Figure 2(c) shows the result of seeds growing and Figure 2(d) shows the final result after region merging.



**Fig. 2.** (a) original image, (b) seed regions, (c) region growing result and (d) final segmentation result

This new approach of using CIE Lab color space was tested with a variety of images and was proven to be suitable for color implemented for many images.

### 3) Comparison between CIE Lab and ab

Gomez et al. performed experiments on color based image segmentation using CIE Lab color space and found that the *ab* color component combination outperforms all the other possible combinations such as Lab for their segmentation algorithm [18]. Therefore, we also made comparisons between CIE Lab and ab space for our color segmentation approach to confirm which one is more suitable for our specific application.

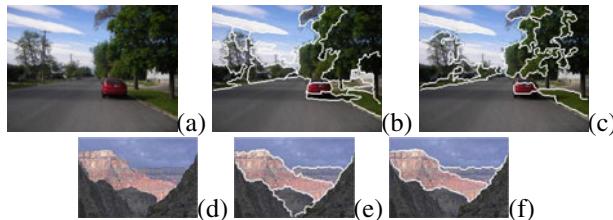
In the ab space, Euclidean distance CIE76 was modified as

$$\Delta E_{76ab} = \|v'_{i,j} - v'_{j,j}\| \quad (4)$$

where,  $v'_{i,j} = [a_i, b_i]$ ,  $v'_{j,j} = [a_j, b_j]$

Similarly, Equation (3) was modified to its corresponding formulations by deleting L component.

The proposed automatic seeded region growing approach for color image segmentation in CIE Lab and ab space were respectively employed to several images. The segmented results are shown in Fig.3. For the street scene, the segmented result in CIE ab is better than that in CIE Lab since the former classifies the shadow on the ground near the car as a single part while the latter merges it with the ground. For mountain scene, the approach in CIE ab outperforms that in CIE Lab since the segmented results in CIE Lab for them are closer to human perception. Thereby, the performance of the segmented results depends on specific application with regard to which one is more suitable between CIE Lab and ab color space.



**Fig. 3.** (a) Street scene, (b) segmented results in CIE Lab space and (c) CIE ab space; (d) mountain scene, (e) segmented results in CIE Lab space, and (f) CIE ab space

## 3 Color Region Matching in Perceptually Uniform Color Space

Characteristic points are generally used to solve the stereoscopic problem. A number of relevant approaches have been proposed in the literature. However, mismatches over pairs of segments appear frequently due to the lack of features available for distinguishing between segments. These shortcomings may be overcome by taking more developed characters such as regions. Since regions contain much richer information, the possibility of making a wrong decision upon a region could be

greatly reduced [19]. The region matching process is achieved usually in two steps: finding the candidate matches and selecting the correct matching regions.

After image segmentation was accomplished by the proposed automatic seeded region growing algorithm in CIE Lab, in addition to segmenting images into regions, the algorithm also extracts the geometrical properties of each of those segmented regions. These properties include: area  $A$ , centroid  $(x,y)$ , perimeter length  $P$ . Candidates are chosen based on the epipolar and the relative position constraint [19]. For each region in the segmented right image, only those regions in the left image that meet the constraints are chosen as the candidates for matching.

For comparing similarities between regions, we need to construct a cost function. El Ansari et al. proposed to compute color edge magnitude on the leftmost and rightmost boundary as a cost function to match regions [20]. They treated and processed color information in three separate channels instead of points in a color space that represents actual colors. However, human vision system processes real color data not in three separate dimensions. Hereby, in this study, we proposed to replace their color edge magnitude in RGB color space with color difference  $\Delta E_{76}$  in CIE Lab color space which can be used to directly represent human perception. At the same time, geometrical region descriptors were included since they are usually simple but important properties for a region. For a given region  $R_L$  in the left image, we consider only the regions  $R_R$  in the right image, which meet the relative position constraints [19]. The best match for  $R_L$  constitutes the solution to the following minimization problem.

$$\hat{R}_R = \min_{R_R} (\Delta E_{76RL} + \| R - L \|) \quad (5)$$

where,  $\Delta E_{76RL}$  means color difference between left and right region,  $R=[A_R, P_R, x_R, y_R]$ ,  $L=[A_L, P_L, x_L, y_L]$

## 4 Depth Map Computation

Many methods use the disparity map directly to detect whether there are objects or not in front of the cameras. For many applications such as 3D shape matching, robot grasping, an accurate depth map is very critical. Some depth map computation methods for objects have been developed [21, 22]. Labayrade et al. [23] proposed the “V-disparity” concept aiming at simplifying the process of separating obstacles from road surfaces. Although this method has been employed in many applications, it has some drawbacks. For example, it needs to use complicated thresholding methods to extract different obstacle features from V-disparity map [24].

Another method for depth map computation is to use 3D reprojection. Given a 2-D homogeneous point  $(x, y)$  and its associated disparity  $d$ , we can project the point into three dimensions using:

$$Q \begin{bmatrix} x \\ y \\ d \\ 1 \end{bmatrix} = \begin{bmatrix} X \\ Y \\ Z \\ W \end{bmatrix} \quad (6)$$

Here, the 3D coordinates are  $(X/W, Y/W, Z/W)$  and the reprojection matrix Q is [25]:

$$Q = \begin{bmatrix} 1 & 0 & 0 & -c_x \\ 0 & 1 & 0 & -c_y \\ 0 & 0 & 0 & f \\ 0 & 0 & \frac{-1}{T_x} & \frac{(c_x - c_x')}{T_x} \end{bmatrix} \quad (7)$$

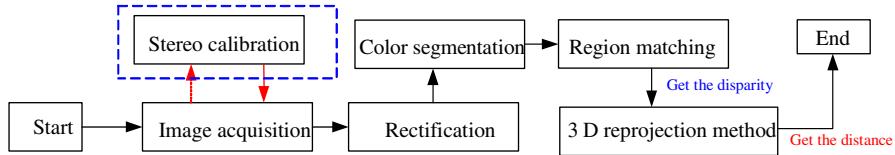
From (6) and (7), the distance z can be derived as

$$z = \frac{-T_x f}{d - (c_x - c_x')} \quad (8)$$

where,  $c_x$  and  $c_x'$  are the x coordinates of the principal points in left and right images, respectively.  $T_x$  (the baseline) is the calibrated distance between the two cameras,  $f$  is the calibrated focal length.

Given the rotation matrix and translation vector ( $R, T$ ) between the stereo images and the camera intrinsic matrix, Bouguet algorithm [26] for stereo rectification can be used to obtain the reprojection matrix Q.

We constructed a framework to calculate depth distances of objects in combination with color region matching and color segmentation in CIE Lab color space. The flowchart is illustrated in Fig.4.

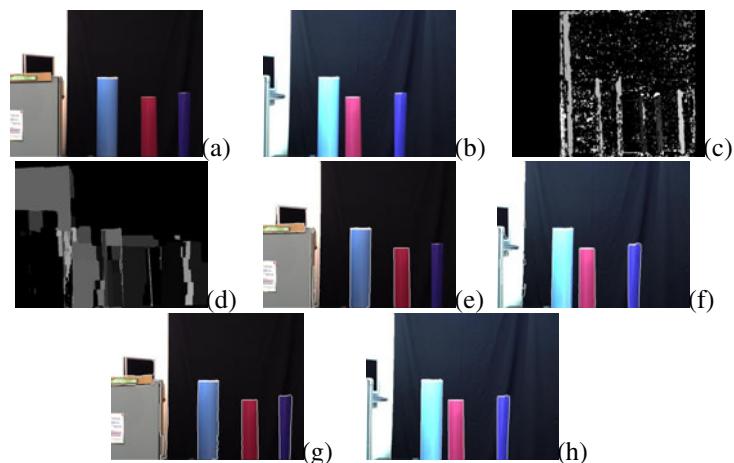


**Fig. 4.** Flowchart

## 5 Experimental Results

Figs. 5(a) and (b) display a pair of stereo images of three color columns. Figs. 5(c) and (d) are the disparity maps using SAD and global method based on graph cuts. Due to the blank wall problem, the disparity maps using these two methods are not satisfactory. Obviously, it is difficult to use these disparity maps to estimate the object distance. Therefore, the presented depth map computation framework for objects combined with color segmentation and region matching was implemented. The segmented results using proposed method in CIE Lab shown in Figs. 5(e) and (f) are worse than the results in CIE ab shown in Figs. 5(g) and (h) because the latter are closer to human perception. CIE ab color space is more suitable in this case.

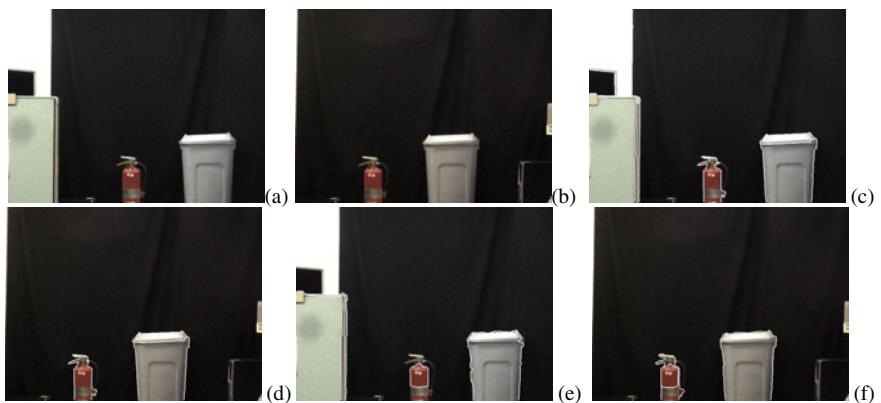
We also implemented the region matching algorithm using our cost function and El Ansari's cost function. El Ansari's cost function did not work well because the leftmost columns in two images differ sharply under the different exposure condition of left and right cameras. On the contrary, our cost function can alleviate this influence since we considered the geometrical cost function with color difference.



**Fig. 5.** Stereo pairs of columns (a) left image, (b) right image, (c) disparity map using SAD, (d) global method based on graph cuts, (e) the segmented left image and (f) the right image using the proposed method in CIE Lab, and (g) the segmented left image and (h) right image using CIE ab color space

**Table 1.** Calculation results (mm)

Regions	Calculated value	Real distance	Error
Cyan column	3170	3000	5.7%
Red column	2160	2040	5.9%
Purple column	2809	2660	5.3%



**Fig. 6.** (a) Left image, (b) right image, (c) and (d) the segmented results using the proposed method in the CIE Lab, (e) and (f) the segmented results using the proposed method in the CIE ab

Finally, the disparity value for each region was obtained after the region matching and then the depth distance of each region was calculated. Table 1 summarized the final results.

Fig.6 shows another example in which the proposed segmentation method in CIE Lab and ab space were implemented for comparison. The segmented results in the Lab space are better than that in the ab space. Our cost function and El Ansari's cost function for region matching were implemented. There was no noticeable difference in performance in this case. However, the computation time of El Ansari's method is much more than ours since we do not need to calculate complex color edge magnitude on the leftmost and rightmost boundary.

## 6 Conclusions

We have developed an object distance estimation framework based on stereovision and color segmentation with color region matching in CIE Lab color space. Experiments show that it is difficult to get a good disparity map in some cases even if the global block matching method based on graph cuts is used. On the contrary, the proposed approach was able to obtain result with adequate accuracy. Experimental results validated the feasibility of the proposed method.

## References

1. Nedevschi, S., Danescu, R., Frentiu, D., Marita, T., Oniga, F., Poco, C.: High Accuracy Stereo Vision System for Far Distance Obstacle Detection. In: 2004 IEEE Intelligent Vehicles Symposium, University of Parma, Parma, Italy, June 14-17 (2004)
2. Lemonde, V., Devy, M.: Object detection with stereovision. Mechatronics and Robotics, Aachen, Germany (September 2004)
3. Broggi, A., Caraffi, C., Fedriga, R.I., Grisleri, P.: Object detection with stereo vision for off-road vehicle navigation. In: 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, vol. 3, pp. 65–72 (June 2005)
4. Saito, T., Morimitsu, N., Sekiguchi, H., et al.: Next Generation Driving Assist System Using New Stereo Camera, FISITA (Septemper 2008)
5. Zhao, J., Whitty, M., Katupitiya, J.: Detection of Non-flat Ground Surfaces Using V-Disparity Images. In: The 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, USA, October 11-15 (2009)
6. Boykov, Y., Veksler, O., Zabih, R.: Fast approximate energy minimization via graph cuts. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(11), 1222–1239 (2001)
7. Sun, J., Zheng, N.-N., Shum, H.-Y.: Stereo matching using belief propagation. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(7), 787–800 (2003)
8. Xia, Y., Gan, Y., Li, W., Ning, S.: A Simple Object detection Approach Based on Stereo Vison in ALV System. In: 2009 IITA International Conference on Control, Automation and Systems Engineering (2009)
9. Hong, L., Chen, G.: Segment-based stereo matching using graph cuts. In: Proc. 2004 IEEE Computer Society Conf. on Computer Vision and Pattern Recognition, Washington, DC (2004)

10. Bleyer, M., Gelautz, M.: Graph-based surface reconstruction from stereo pairs using image segmentation. In: Proc. SPIE, vol. 5665, pp. 288–299 (2005)
11. Bleyer, M., Gelautz, M.: Graph-cut-based stereo matching using image segmentation with symmetrical treatment of occlusions. *Signal Processing: Image Communication* 22, 127–143 (2007)
12. Mühlmann, K., et al.: Calculating Dense Disparity Maps from Color Stereo Images: an Efficient Implementation. *International Journal of Computer Vision* 47(1-3), 79–88 (2002)
13. Deng, Y., Manjunath, B.S.: Unsupervised segmentation of colortexture regions in images and video. *IEEE Transactions of Pattern Analysis and Machine Intelligence* 23(8), 800–810 (2001)
14. Fan, J., Yau, D.K.Y., Elmagarmid, A.K., Aref, W.G.: Automatic image segmentation by integrating color-edge extraction and seeded region growing. *IEEE Transactions on Image Processing* 10(10), 1454–1466 (2001)
15. Shih, F.Y., Cheng, S.: Automatic seeded region growing for color image segmentation. *Image and Vision Computing* 23, 877–886 (2005)
16. Paschos, G.: Perceptually uniform color spaces for color texture analysis: An empirical evaluation. *IEEE Trans. Image Process.* 10(6), 932–937 (2001)
17. Acharya, T., Ray, A.K.: *Image Processing Principles and Applications*. A John Wiley & Sons, Inc., Chichester (2005)
18. Hernandez-Gomez, G., Sanchez-Yanez, R.E., Ayala-Ramirez, V., Correa-Tome, F.E.: Natural image segmentation using the CIELab space. In: 2009 International Conference on Electrical, Communications, and Computers (2009)
19. El Ansari, M., Masmoudi, L., Radoune, L.: A new region matching method for stereoscopic images. *Pattern Recognition Lett.* 21, 283–294
20. El Ansari, M., et al.: A new regions matching for color stereo images. *Pattern Recognition Letters* 28, 1679–1687 (2007)
21. Soquet, N., Aubert, D., Hautiere, N.: Road Segmentation Supervised by an Extended V-Disparity Algorithm for Autonomous Navigation. In: Proceedings of the 2007 IEEE Intelligent Vehicles Symposium, Istanbul, Turkey, June 13-15 (2007)
22. Collins, B.M., Kornhauser, A.L.: Stereo vision for object detection in autonomous navigation, DARPA grand challenge Princeton university technical paper (May 2006)
23. Labayrade, R., Aubert, D.: Robust and Fast Stereovision Based Road Obstacles Detection for Driving Safety Assistance. In: IAPR Workshop on Machine Vision Applications, December 11-13, Nara-ken New Public Hall, Nara (2002)
24. Lee, C.-H., Lim, Y.-C., Kwon, S., Lee, J.-H.: Obstacle localization with a binarized v-disparity map using local maximum frequency values in stereo vision. In: 2008 International Conference on Signals, Circuits and Systems (2008)
25. Bradski, G., Kaehler, A.: *Learning OpenCV – Computer Vision with the OpenCV Library*. O'Reilly Media, Inc., Sebastopol (2008)
26. Bouguet, J.-Y.: Visual methods for three-dimensional modeling, PhD Thesis (1999)

# Multiscale Information Fusion by Graph Cut through Convex Optimization

Yinhui Zhang, Yunsheng Zhang, and Zifen He

Kunming University of Science and Technology, Yunnan, China  
yinhui\_z@yahoo.com.cn

**Abstract.** This paper proposed a novel method for global continuous optimization of maximum a posterior(MAP) during wavelet-domain hidden Markov tree-based(WHMT) multiscale information fusion process. We start with calculating the multiscale classification likelihoods of wavelet coefficients by expectation-maximization(EM) algorithm. Energy function is then generated by combining boundary term estimated by classification likelihoods with regional term obtained by both pixel information and approximation coefficients. Through energy minimization through graph cut via convex optimization, objects are segmented accurately from the images in a global optimization sense. A performance measure for tobacco leaf inspection is used to evaluate our algorithm, the localization accuracy of weak boundary by fusing multiscale information via convex optimization is encouraging.

## 1 Introduction

The scene in which objects to be extracted usually includes objects of varying shapes and sizes. The sharpness of object boundaries is affected by various factors, such as artifacts, noise and inhomogeneous illumination. These factors always lead to boundary blurry that made accurate object identification difficult. After the seminal work by Crouse and Nowak [1], an important multiscale context probabilistic statistical method, namely, wavelet-domain hidden Markov tree models have been successfully used for computer vision inspection, such as image segmentation [2–5], texture analysis [6], feature extraction [7, 8] and edge detection [9].

Work is being done in this field by a number of researchers. The problem of WHMT modeling and WHMT-based texture classification has been discussed in [1, 2]. Ref. [1] proposed a kind of hidden Markov model which is called hidden Markov tree for statistical in wavelet domain. Since the wavelet transform can decorrelate image data by reducing the number of state of wavelet coefficients, thus making WHMT manipulable and useful for statistical modeling. One method to enhance the texture classification performance by using Bayesian probabilistic graph is described in [2]. This method characterizes the clustering property of wavelet coefficients as a non-Gaussian function and the persistence property of wavelet coefficients as a hidden Markov model based on quad-tree

structure. They make use of WHMT is particularly well suited for images containing singularities to provide a good texture classifier. Fan suggested a joint multicontext and multiscale Bayesian image fusion algorithm in [3]. This method is similar to [2] except that the neighborhood information is characterized by several upper scales instead of a single one. However, such fusion could bring large computation burden to the system and make it not practical for real applications. To utilize the reliability of coarse scale classification, E.Mor [4] proposed a WHMT-based image segmentation method by using boundary refinement conditions to segment the finest scale image. In the method suggested by [6], the dependences between color planes and the interactions across scales are modeled by WHMT. However, the WHMT fusion methods proposed in these approaches need supervised training by using singular image regions.

Recently, due to the favorable statistical learning ability of the WHMT model, it has been applied to many unsupervised identification areas, such as writer identification [7], steam feature extraction [8] and edge detection [9]. In [7], Z.He suggested a method for off-line text-independent writer identification using WHMT model. They reported that the WHMT-based model not only can achieve better identification results than that of the Gabor model but also reduce the computation time. R.J.Ferrari et al. [8] suggested a real-time steam detection system in which steam texture pattern is characterized by WHMT derived from the coefficients of the dual-tree complex wavelet transform. Then the parameters of the WHMT model are used as an input feature vector to a support vector machine to detect the presence of steam in video images. In Ref. [9], the authors employed wavelet-domain vector hidden Markov tree to model the statistical properties of multiscale and multidirectional wavelet coefficients to detect region edges in an image. But the dynamical programming algorithm they employed for multiscale fusion has not the capacity to incorporate different visual cues into the object identification framework.

After WHMT model training is done, the original image can be decomposed into a multiscale and multiband representation. In Ref. [2][4][6], each WHMT model has to be trained with a single texture image, i.e., each texture is represented by a corresponding WHMT model. This method is memory consuming and do not work for unknown textures. Moreover, the approaches of WHMT fusion they proposed could not find the global optimum through EM algorithm. In fact, our previous experiments have shown that the drawback of local optimum in characterizing object regions, especially weak boundary, is obvious [10, 11]. In this paper, we intend to fusion multiscale boundary classification likelihoods and regional information of the WHMT model in a global optimum sense for the purpose of unsupervised object identification.

The main contribute in this work is that the WHMT model can fusion both detail and approximation coefficients in a global optimum sense to extract the objects accurately at the pixel resolution level, while the traditional fusion methods via EM always detected on locally minima. The first step in our algorithm is to calculate the multiscale classification likelihoods of detail coefficients. The calculation is implemented by EM learning algorithm. In the second step, graph

cut algorithm via convex optimization is employed to minimize energy function and classification is obtained at the pixel level.

The paper is organized as follows. Section 2 describes the multiscale statistical modeling by WHMT technique. Section 3 describes unsupervised multiscale classification likelihoods estimation. In Section 4, our multiscale information fusion methods are developed. The multiscale energy function construction and pixel classification by energy minimization through graph cut via convex optimization are presented to perform multiscale information fusion of WHMT. Experimental results and discussion are derived in Section 5. In Section 6, conclusions are drawn.

## 2 Multiscale Statistical Modeling by WHMT

In the WHMT model as proposed in [1], we use the probability mass function

$$p(s_i = m) = \begin{bmatrix} p(s_i = 0) \\ p(s_i = 1) \end{bmatrix}$$

of node  $i$  to denote the distribution of hidden states, where  $m \in \{0, 1\}$ . The state transition probability  $\epsilon_i^{p(i)}$  is defined to represent the probability for  $w_i$  to be small(or large) when its parent  $w_{p(i)}$  is small(or large). Since a wavelet coefficient has four children, thus  $\epsilon_i^{p(i)}$  is a  $2 \times 2$  probability matrix representing state transition probability from  $w_{p(i)}$  to its four children

$$\epsilon_i^{p(i)} = \begin{bmatrix} \epsilon(s_i = 0 | s_{p(i)} = 0) & \epsilon(s_i = 0 | s_{p(i)} = 1) \\ \epsilon(s_i = 1 | s_{p(i)} = 0) & \epsilon(s_i = 1 | s_{p(i)} = 1) \end{bmatrix}.$$

For the single object identification problem, the column sums of state transition probability matrix are equal to 1.

With the two models discussed so far, the WHMT model is defined by a parameter set

$$\Theta = \left\{ p_J(s_i = m), \epsilon_i^{p(i)}, \sigma_i^2 | m \in \{0, 1\} \right\} \quad (1)$$

for each  $LH, HL$  and  $HH$  subband. Where  $p_J(s_i = m)$  denotes the probability mass function  $p(s_i = m)$  of node  $i$  at the coarsest scale  $J$ . The complete WHMT model consists of three sub-WHMT models each represents for a subband. For gray image processing, we employ the subband independence assumption, the complete WHMT model is thus

$$f(W|\Theta) = f(W^{LH}|\Theta^{LH})f(W^{HL}|\Theta^{HL})f(W^{HH}|\Theta^{HH}) \quad (2)$$

## 3 Multiscale Classification Likelihoods Estimation

A energy minimization algorithm based on graph cut is developed to implement multiscale classification likelihoods fusion. The proposed graph cut via convex

optimization algorithm is motivated by [12–15] except that our algorithm incorporates multiscale likelihoods information into energy function. In addition, our algorithm need not manual selection of seeds, i.e., the pixel classification processing at the pixel level is fully unsupervised. The implementation of E-step applied to tree-structured WHMT model contains two sweeps: up sweep and down sweep along the tree.

### 3.1 Up Sweep

Each up sweep iteration contains four steps:

- 1) Initialization: For all hidden state variables  $s_i$  at scale  $j = 1$ , calculate the conditional likelihoods of each subtree  $\mathcal{T}_i$  given that it is in state  $m$ :

$$\beta_i(m) = f(\mathcal{T}_i | s_i = m, \Theta).$$

- 2)  $j = j + 1$ .

- 3) For all hidden state variables  $s_i$  at scale  $j$ , compute the conditional likelihoods of each subtree  $\mathcal{T}_i$  given that its parent is in state  $m$ :

$$\beta_{i,p(i)}(m) = f(\mathcal{T}_i | s_{p(i)} = m, \Theta).$$

The conditional likelihoods of each subtree  $\mathcal{T}_{p(i)}$  at the next coarser level is thus calculated by

$$\beta_{p(i)}(m) = g(w_{p(i)}; \sigma_{p(i)}^2) \prod_{i \in C(p(i))} \beta_{i,p(i)}(m)$$

where  $C(p(i))$  represents the four children of node  $p(i)$ . The conditional likelihoods of subtree  $\mathcal{T}_{p(i)\setminus i}$  given  $p(i)$  is in state  $m$  is calculated by

$$\beta_{p(i)\setminus i}(m) = f(\mathcal{T}_{p(i)\setminus i} | s_{p(i)} = m, \Theta).$$

- 4) If  $j = J$ , then stop; else return to step 2).

### 3.2 Down Sweep

Similar to the up sweep, each down sweep also contains four steps:

- 1) Initialization: For each hidden state variable at the coarsest scale  $j = J$ , let  $\alpha_{i,J}(m) = p(s_{i,J} = m)$ ,  $m \in \{0, 1\}$ .

- 2)  $j = j - 1$ .

- 3) For each hidden state variable  $s_i$  at scale  $j$ , compute the joint probability function

$$\alpha_i(m) = p(s_i = m, \mathcal{T}_{j\setminus i} | \Theta).$$

- 4) If  $j = 1$ , then stop; else return to step 2).

According to the chain rule of probability, the desired state probabilities are obtained by

$$p(s_i = m | W, \Theta) = \frac{\alpha_i(m) \beta_i(m)}{\sum_{n=0}^1 \alpha_i(n) \beta_i(n)} \quad (3)$$

and

$$p(s_i = m, s_{p(i)} = n | W, \Theta) = \frac{\beta_i(m) \epsilon_{s_i=m}^{s_{p(i)}=n} \alpha_{p(i)}(n) \beta_{p(i)\setminus i}(n)}{\sum_{n=0}^1 \alpha_i(n) \beta_i(n)}. \quad (4)$$

The M-step finds the optimal parameter  $\Theta^{k+1}$  in terms of the expectation which can be mathematically formulated as

$$\Theta^{k+1} = \arg \max_{\Theta} E [\ln p(W, S | \Theta) | W, \Theta^k]. \quad (5)$$

The EM algorithm iterates until  $\|\Theta^{k+1} - \Theta^k\| \leq \xi$ , where  $\xi$  denotes termination condition and  $\|\cdot\|$  is the usual Euclidean norm. Once the EM algorithm converged, the multiscale classification likelihoods(MCL) of  $w_{i,j}$  given  $\Theta$  is thus estimated by:

$$MCL(w_{i,j} | \Theta) = \sum_{m=0}^1 \beta_{i,j}(m) \alpha_{i,j}(m). \quad (6)$$

## 4 Multiscale Information Fusion

We intend to merge boundary and region classification information in order to improve the object identification accuracy at the pixel resolution level. The idea of multiscale classification likelihoods and approximate coefficients fusion is motivated by our previous experiments reported in [10, 11, 16]. In the following, we will employ convex optimization energy minimization algorithm proposed in [15] to perform optimal pixel classification. The main differences between our algorithm and that in [15] is the presence of the multiscale energy function and the unsupervised foreground-background segmentation process.

### 4.1 Classification Energy

We assign a label  $L_v$  to each node  $v \in V$  in  $G_{st}$  with probability of  $MCL(v | \Theta)$ . Then  $L = (L_1, \dots, L_v, \dots, L_{|V|})$  denotes a classification of the original image pixels. Let  $E$  be the energy of classification  $L$ :

$$E(L) = \lambda \cdot E_R(L) + E_B(L) \quad (7)$$

where  $E_R(L)$  is the regional term by which we impose smoothness constrains on the classification. Similarly,  $E_B(L)$  represents the boundary term by which singularity constrains are imposed. A weight factor  $\lambda$  is assigned between the regional term and boundary term and we let  $\lambda = 1.2$  in the following experiments. In our method, the smoothness and singularity constrains derives from pixel information plus approximation coefficients and multiscale classification likelihoods, respectively.

Since the original images are smoothed by the discrete wavelet transform, thus we can take advantage this fact to obtain the regional term. Motivated by MAP-MRF formulations in [12], we use negative log-likelihood of the intensity

of pixels and approximation coefficients, given intensity histogram models of the object and background to obtain regional term:

$$E_R = -\log(P(I_{v,J}) + P(I_{v,0})) \quad (8)$$

for  $m \in \{0, 1\}$ .

The boundary term can be formulated by multiscale classification likelihoods that we have discussed in Section 3:

$$E_{B,j}(L) = \sum_{\{p,q\} \in \mathcal{N}} B_{\{p,q\},j} \cdot \delta_j \quad (9)$$

for  $j \in \{0, 1, \dots, J\}$ , where  $B_{\{p,q\},j}$  can be viewed as a penalty for the discontinuity between node  $p$  and  $q$ , and

$$\delta_j = \begin{cases} 1 & \text{if } MCL(s_p|\Theta) \neq MCL(s_q|\Theta) \\ 0 & \text{if } MCL(s_p|\Theta) = MCL(s_q|\Theta) \end{cases}$$

We have adopted the following boundary penalties from a function:

$$E_{B,j}(L) \propto \frac{\exp\left(-\frac{(I_{p,j} - I_{q,j})^2}{2\sigma_j^2}\right)}{\|MCL(s_{p,j}=1|\Theta) - MCL(s_{q,j}=1|\Theta)\|} \quad (10)$$

According to the above equation, the boundary penalties are larger when the intensities or multiscale classification likelihoods between node  $p$  and  $q$  are similar. On the other hand, the distinct difference between  $p$  and  $q$  will lead to small boundary penalties.

## 4.2 Fusion through Graph Cut via Convex Optimization

We employ the graph cut via convex optimization method proposed in [15] to implement global energy minimization. The main difference between our algorithm and that in [15] is the construction of multiscale energy function. Considering the directed graph  $G_{st} = (V \cup \{s, t\}, E)$ , the energy minimization is an iterative process, each of which contains three stages.

Let  $x \in R^{|E|}$  denote a vector indicating the flow in each of the edges of the graph, where  $|E|$  is number of edges in  $G_{st}$ . The goal of the optimization is to maximize the inner product under capacity and conservation constrains:

$$\begin{aligned} \max_x \quad & \mathbf{c}^T \mathbf{x} \\ \text{st} \quad & A\mathbf{x} = 0 \\ & -\mathbf{w} \leq \mathbf{x} \leq \mathbf{w} \end{aligned}$$

where  $\mathbf{c} \in R^{|E|}$  is a binary vector with +1 entries for all of the edges emanating from source  $s$  and 0 entries elsewhere.  $A \in R^{|V|}$  denotes the node edge incidence matrix and  $|V|$  is the number of nodes in  $G_{st}$ .  $\mathbf{w}$  is the capacity constrains associated with each of the edges.

Then the optimal value is computed by maximizing the associated Lagrangian dual function as follows:

$$\begin{aligned} & \max_{\lambda, \nu} \quad \mathbf{w}^T(\lambda_+ + \lambda_-) \\ \text{st} \quad & A^T \nu - \mathbf{c} = (\lambda_- - \lambda_+) \\ & \lambda_+ \geq 0, \lambda_- \geq 0 \end{aligned}$$

where Lagrangians  $\lambda$  and  $\nu$  correspond to the capacity and conservation constraints, respectively. For a fixed value of  $\nu$ , the minimum value that  $(\lambda_+ + \lambda_-)_i$  attains is  $|(A^T \nu - \mathbf{c})_i|$ . Thus the optimization problem can be reformulated as

$$\min_{\nu} \| \text{diag}(\mathbf{w})(A^T \nu - \mathbf{c}) \|_1 \quad (11)$$

Notice that the unconstrained formulation reveals the connection between graph cuts and convex optimization and the continuous optimization techniques, such as Newton's method, can be employed to compute the global minimization.

## 5 Experimental Results and Discussion

In our experiments, the tobacco targets are captured by a SPYDER3 line CCD camera and eight frame images each containing  $2048 \times 2048$  pixels are first used to test our algorithm. The preprocessing of one every four downsampling is followed to reduce computing cost, thus there are  $512 \times 512$  pixels in each preprocessed frame. We use a personal computer with CPU 1.86GHz and 1G memory to perform the testing. The programs are written in Matlab7.0.1.

### 5.1 Unsupervised Learning of WHMT Model

The model initialization and unsupervised learning by EM algorithm have been discussed in Section 3. The single parameter that needs to be selected is the termination condition in the EM iterations, which is selected as  $10^{-2}$  during the unsupervised learning experiments.

To evaluate the learning time, we calculate the average elapsed time of the eight frames when different termination conditions are selected. The test results are shown in Tab.1. We note that the total learning time increase rapidly with the decrease of termination condition. Though the learning time of the three subbands seems similar, the average learning time of HH subband tends to account for the main part of the total learning time with the decreasing of  $\xi$ . For example, the learning time of HH subband exceeds 50% of the total learning time when  $\xi = 10^{-5}$ . Moreover, the longer learning time is a disastrous in the case of real time target identification. Fortunately, in our experiments we find that there seems no benefit for too lower termination condition, i.e., unsupervised learning with larger termination condition ( $\xi = 10^{-2}$  for example) could also achieve high identification accuracy.

### 5.2 Experimental Results and Discussion

We adopted four methods to perform multiscale classification likelihoods and approximation parameters fusion through energy minimization. The former three

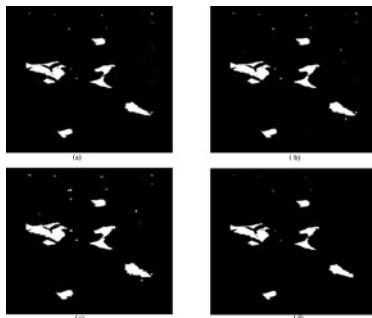
**Table 1.** Average elapsed time of eight frames during learning(seconds)

$\xi$	LH	HL	HH	Total time
$10^{-1}$	0.69	0.65	0.65	1.99
$10^{-2}$	3.20	3.09	3.30	9.59
$10^{-3}$	5.67	4.04	5.12	14.83
$10^{-4}$	5.03	5.09	9.55	19.67
$10^{-5}$	6.39	9.07	19.27	34.73

methods only consider boundary term. In method1, we fusion classification likelihoods of scale1, i.e., the energy term is given by  $E(L) = E_{B,j}(L)$ , where  $j = 1$ . In method2 we obtain boundary penalty term by  $E(L) = \sum_{j=1}^2 E_{B,j}(L)$ . In method3 we fusion scale1, 2 and 3 by  $E(L) = \sum_{j=1}^3 E_{B,j}(L)$ . The last method considers both boundary term and regional term by setting the energy to  $E(L) = \lambda \cdot E_{R,J}(L) + \sum_{j=1}^3 E_{B,j}(L)$ .

The classification results of the four methods are shown in Fig.1. In our four fusion methods, the iteration termination condition  $\xi$  is set to  $10^{-2}$ . From Fig.1(a) we note that the fusion of multiscale classification likelihoods of scale1 can achieve a fairly good identification accuracy. However, the fusion of higher scales in succession will deteriorate the identification accuracy. This phenomenon can be observed in Fig.1(b) and Fig.1(c), where the boundary of tobacco leave detected becomes coarser as  $j$  is increased. The main reason is that the fusion of higher scales penalize too much the boundary discontinuities while in absence of regional information. From Fig.1(d) we note that the fusion of multiscale classification likelihoods and approximate parameters result in the best identification results among the four fusion methods. Note that the boundary of the tobacco targets localized is fairly smoother than the other three methods. Further more, our fusion of approximation parameters yielding more robust target detection results.

To evaluate the performance of our algorithm quantitatively, we employed three numerical criteria proposed in [3]. The identification accuracy is represented by Pa



**Fig. 1.** Identification results of frame1 when  $\xi = 10^{-2}$ . (a)-(d): the classification results of method1-method4, respectively.

which is defined as the percentage of pixels classified correctly. The percentage of detected boundaries that are consistent with the truth boundaries is denoted by Pb, showing specificity. The percentage of true boundaries that can be detected is denoted by Pc, showing sensitivity of the identification algorithm. The averaged identification results of eight frames are shown in Tab.2.

Note that the quantitative evaluate results in Tab.2 are consistent with our observations from Fig.1. Though the classification accuracy of the former three methods is very similar, the boundary specificity and sensitivity reduced with the fusion of higher scale likelihoods. For example, we fusion scale1, 2 and 3 in method3, the classification specificity and sensitivity dropped to 6.20% and 7.11%, respectively. Among our four fusion methods, method4 has the highest criteria values which reached to 98.01%, 34.71% and 36.02%, respectively. In addition, our method does not need to train the WHMT model by singular image regions, i.e., our method is fully unsupervised.

**Table 2.** The quantitative evaluation results of our method

Method	Pa(%)	Pb(%)	Pc(%)
1	96.32	15.49	18.27
2	95.27	11.17	12.01
3	92.31	6.20	7.11
4	98.01	34.71	36.02

## 6 Conclusions

In this work, we have proposed a multiscale information fusion method of wavelet-domain hidden Markov tree model for accurate object localization. In contrast with earlier work, our method need not supervised training and could fusion the MAP estimation presented in both wavelet and approximation coefficients in a global optimum sense. After representing the multiscale classification likelihoods and approximation coefficients by WHMT model, the energy term is generated, which consists of both boundary and regional term. Then we employed energy minimization method based on graph cut via convex optimization to calculate the minimum cut of the arc-weighted directed graph constructed at the pixel resolution level. Finally, we identified the tobacco targets and tested the identification accuracy, specificity and sensitivity of the proposed method. Quantitative evaluation of the algorithm shows that our method is encouraging in terms of classification accuracy, specificity and sensitivity.

## Acknowledgments

This work was supported by the National Science Foundation of China(NSFC) under Grant 60962007 and by the Yunnan Education Office Foundations under Grant 6Y0145D and 08Y0091.

## References

1. Crouse, M.S., Nowak, R.D., Baraniuk, R.G.: Wavelet-based statistical signal processing using hidden Markov models. *IEEE Transactions on Signal Processing* 46, 886–902 (1998)
2. Choi, H., Baraniuk, R.G.: Multiscale image segmentation using wavelet-domain hidden Markov models. *IEEE Transactions on Image Processing* 10, 1309–1321 (2001)
3. Fan, G.: Wavelet domain statistical image modeling and processing. PhD Thesis, Oklahoma University (2001)
4. Mor, E., Aladjem, M.: Boundary refinements for wavelet-domain multiscale texture segmentation. *Image and Vision Computing* 23, 1150–1158 (2005)
5. Ye, Z., Lur, C.: Wavelet-based unsupervised SAR image segmentation using hidden Markov tree models. In: Proceedings of the Int'l Conference on Pattern Recognition, pp. 729–732 (2002)
6. Xu, Q., Yang, J., Ding, S.: Color texture analysis using the wavelet-based hidden Markov model. *Pattern Recognition Letters* 26, 1710–1719 (2005)
7. He, Z., You, X., Tang, Y.Y.: Writer identification of Chinese handwriting documents using hidden Markov tree model. *Pattern Recognition* 41, 1295–1307 (2008)
8. Ferrari, R.J., Zhang, H., Kube, C.R.: Real-time detection of steam in video images. *Pattern Recognition* 40, 1148–1159 (2007)
9. Sun, J., Gu, D., Chen, Y., Zhang, S.: A multiscale edge detection algorithm based on wavelet domain vector hidden Markov tree model. *Pattern Recognition* 37, 1315–1324 (2004)
10. Zhang, Y.H., Zhang, Y.S., Tang, X.Y., He, Z.F.: Unsupervised image sequence segmentation based on hidden Markov tree model. In: Proceedings of the Chinese Control Conference, pp. 495–499 (2008)
11. Zhang, Y.H., Zhang, Y.S., He, Z.F., Tang, X.Y.: Automatic inspection of tobacco leaves based on MRF image model. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Paragios, N., Tanveer, S.-M., Ju, T., Liu, Z., Coquillart, S., Cruz-Neira, C., Müller, T., Malzbender, T. (eds.) ISVC 2007, Part II. LNCS, vol. 4842, pp. 662–670. Springer, Heidelberg (2007)
12. Boykov, Y., Veksler, O., Zabin, R.: Fast approximate energy minimization via graph cuts. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23, 1222–1239 (2001)
13. Boykov, Y., Kolmogorov, V.: An experimental comparison of min-cut/max-flow algorithms for energy minimization in vision. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 1124–1137 (2004)
14. Boykov, Y.: Graph cuts and efficient N-D image segmentation. *Int'l Journal of Computer Vision* 70, 109–131 (2006)
15. Bhushnurm, A., Taylor, C.J.: Graph cuts via l1 norm minimization. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30, 1866–1871 (2008)
16. Zhang, Y.H., Zhang, Y.S., He, Z.F., Tang, X.Y.: Multiscale fusion of wavelet-domain hidden Markov tree through graph cut. *Image and Vision Computing* 27, 1402–1410 (2009)

# A Fast Level Set-Like Algorithm for Region-Based Active Contours

Martin Maška, Pavel Matula, Ondřej Daněk, and Michal Kozubek

Centre for Biomedical Image Analysis, Faculty of Informatics

Masaryk University, Brno, Czech Republic

[xmaska@fi.muni.cz](mailto:xmaska@fi.muni.cz)

**Abstract.** Implicit active contours are widely employed in image processing and related areas. Their implementation using the level set framework brings several advantages over parametric snakes. In particular, a parameterization independence, topological flexibility, and straightforward extension into higher dimensions have led to their popularity. On the other hand, a numerical solution of associated partial differential equations (PDEs) is very time-consuming, especially for large 3D images. In this paper, we modify a fast level set-like algorithm by Nilsson and Heyden [14] intended for tracking gradient-based active contours in order to obtain a fast algorithm for tracking region-based active contours driven by the Chan-Vese model. The potential of the proposed algorithm and its comparison with two other fast methods minimizing the Chan-Vese model are demonstrated on both synthetic and real image data.

## 1 Introduction

A well-known approach to image segmentation is the active contour model (also called *snakes*) introduced by Kass et al. [10]. The problem of image segmentation is transformed to the minimization of a specially designed energy functional. Starting with a parametric contour enclosing an object of interest, the contour is evolved under the influence of internal and external forces towards the object boundary, where the functional reaches its local minimum.

Implicit active contours [2–4] have been developed as an alternative to parametric snakes. Their solution is usually carried out using the level set framework [15, 16], in which the contour is represented implicitly as the zero level set (also called *interface*) of a scalar, higher-dimensional function  $\phi$ . This representation has several advantages over the parametric one. In particular, it avoids parameterization problems, the topology of the contour is handled inherently, and the extension into higher dimensions is straightforward.

Depending on the minimized energy functional, implicit active contours are broadly classified as either *gradient-based* or *region-based*. The gradient-based models [2, 3] are driven mainly by the image gradient. The information inside the regions is ignored. In contrast, the region-based models [4, 19] consider the

homogeneity of regions instead of edges. In both cases, the contour evolution is often governed by a partial differential equation in the following general form:

$$\phi_t + F|\nabla\phi| = 0 , \quad (1)$$

where  $F$  is a speed function describing the motion of the interface in the normal direction. A basic PDE-based solution using explicit finite difference schemes results in a significant computational burden limiting the use of this approach in near real-time applications.

Many approximations, aimed at speeding up the basic level set framework, have been proposed in last two decades. In the family of gradient-based implicit active contours, the narrow band [1], sparse-field [22], and fast marching method [17] have become popular. Later, other interesting approaches based on the additive operator splitting scheme [8] or a pointwise scheduled propagation of the implicit contour [5, 14] have emerged. Shi and Carl [18] proposed a fast algorithm that is able to track the gradient-based as well as region-based implicit active contours, provided the speed function can be decomposed into data-dependent and regularization terms. Other techniques developed to minimize popular Chan-Vese model [4] are based on the  $k$ -means clustering [7], threshold dynamics [6], or graph cuts [23]. We also refer the reader to the work by Lie et al. [11] and Wang et al. [21].

In this paper, we modify a fast level set-like algorithm by Nilsson and Heyden [14], that has turned out to be about two orders of magnitude faster [12] than the sparse-field method [22] and the Deng and Tsui algorithm [5], to obtain a fast algorithm minimizing the Chan-Vese model. Instead of storing the interface points in a heap-sorted queue and a pointwise scheduled propagation of the implicit contour, the proposed algorithm stores the interface points in a list data structure and propagates the whole contour in each iteration, which conform better with a global character of the speed function used in the Chan-Vese model. A comparison of the proposed algorithm with two other fast approaches [9, 23] minimizing the Chan-Vese model shows its speed and low memory demands.

The organization of the paper is as follows. In Section 2, the theoretical background and formulation of the Chan-Vese model are reviewed. Section 3 is devoted to the Nilsson and Heyden algorithm and its modification. Experimental results are demonstrated in Section 4. We conclude the paper with a discussion and suggestions for future work in Section 5 and 6, respectively.

## 2 Chan-Vese Model

This section reviews briefly the theoretical background and formulations of the Mumford-Shah and Chan-Vese models.

In [13], Mumford and Shah introduced a functional formulation of image segmentation. The basic idea is to find a pair  $(u, C)$  for a given input image  $u_0 : \Omega \rightarrow \mathbb{R}$ , where  $u$  is a piecewise smooth approximation of  $u_0$ ,  $C \subset \Omega$  is a

smooth and closed segmenting contour, and  $\Omega$  denotes the image domain. The Mumford-Shah model can be written as

$$E_{MS}(u, C) = \mu|C| + \lambda \int_{\Omega} (u_0 - u)^2 dx + \beta \int_{\Omega \setminus C} |\nabla u|^2 dx , \quad (2)$$

where  $\mu$ ,  $\lambda$ , and  $\beta$  are positive constants. The minimization of this functional is a very difficult task, since one seeks for the contour  $C$  as well as the image  $u$ . No unique solution exists in general.

A piecewise constant approximation to the Mumford-Shah model was proposed by Chan and Vese [4]. Assuming that  $u$  is a piecewise constant function being constant in two possibly disconnected regions  $\Omega_1$  and  $\Omega_2$  separated by a contour  $C$  ( $\Omega = \Omega_1 \cup \Omega_2 \cup C$ ), the Chan-Vese model is formulated as

$$E_{CV}(C, c_1, c_2) = \mu|C| + \lambda_1 \int_{\Omega_1} (u_0 - c_1)^2 dx + \lambda_2 \int_{\Omega_2} (u_0 - c_2)^2 dx , \quad (3)$$

where  $\mu$  is nonnegative constant,  $\lambda_1$  and  $\lambda_2$  are positive constants, and  $c_1$  and  $c_2$  represent the constant level inside the regions  $\Omega_1$  and  $\Omega_2$ , respectively. Embedding the contour  $C$  in a higher-dimensional function  $\phi$  with  $C$  as its zero level set, the functional can be minimized using the level set framework. The associated Euler-Lagrange equation has the form:

$$\phi_t + \delta_{\varepsilon}(\phi) \left[ \mu \cdot \operatorname{div} \left( \frac{\nabla \phi}{|\nabla \phi|} \right) - \lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2 \right] = 0 , \quad (4)$$

where

$$c_1 = \frac{\int_{\Omega} u_0 H_{\varepsilon}(\phi) dx}{\int_{\Omega} H_{\varepsilon}(\phi) dx} \quad \text{and} \quad c_2 = \frac{\int_{\Omega} u_0 (1 - H_{\varepsilon}(\phi)) dx}{\int_{\Omega} (1 - H_{\varepsilon}(\phi)) dx} . \quad (5)$$

The symbols  $H_{\varepsilon}$  and  $\delta_{\varepsilon}$  denote regularized versions of the Heaviside and Dirac delta functions. Careful attention has to be paid to the regularization of these functions, since it affects the model performance. Provided  $\delta_{\varepsilon}$  is nonzero in the whole domain, the Chan-Vese model has the tendency to compute a global minimizer. On the contrary, the choice of  $\delta_{\varepsilon}$  with a compact support results only in a local minimizer and, therefore, the dependency on the initialization. Notice that (4) has not the general form of (1). However, in order to extend the evolution to all level sets of  $\phi$ , it is possible to replace  $\delta_{\varepsilon}(\phi)$  by  $|\nabla \phi|$ . Chan and Vese worked with  $\delta_{\varepsilon}(\phi)$  to remain close to the initial minimization problem [4].

To conclude, the Chan-Vese model is able to detect objects with weak boundaries and sharp corners and does not require pre-smoothing of the initial image. In addition, the minimization of this model using the level set framework enables automatic detection of holes in the objects starting with an initial contour not necessarily surrounding the objects to be detected. On the other hand, a numerical solution of the associated Euler-Lagrange equation is a nontrivial and very time-consuming task, especially for large 3D images.

### 3 Proposed Algorithm

A principle of the proposed algorithm is explained in this section. First, we briefly describe the basic idea of the Nilsson and Heyden algorithm, since the proposed one builds on this approach. Modifications to the Nilsson and Heyden algorithm that result in a fast algorithm minimizing the Chan-Vese model are introduced in Section 3.2. We conclude this section with a few remarks on the limitations of the proposed algorithm.

#### 3.1 Nilsson and Heyden Algorithm

A fast approximation of the level set framework exploiting a pointwise scheduled propagation of the implicit contour was introduced in the work by Nilsson and Heyden [14]. Instead of evolving the whole interface in a small constant time step, a point  $p$  of the interface with the minimal departure time is moved to the outside or inside of the interface depending on the sign of the speed function at this point. Simultaneously, its local neighbourhood (4-neighbourhood in 2D and 6-neighbourhood in 3D, respectively) is updated accordingly. The departure time of the interface point corresponds to the time at which the propagation of this point is expected to occur. More precisely, the departure time  $T_d(p)$  of the interface point  $p$  is defined as

$$T_d(p) = T_a(p) + \frac{1}{\max\{|F(p)|, \varepsilon\}} , \quad (6)$$

where  $T_a(p)$  is the arrival time (the time at which the interface arrived to the point  $p$ , this time is initialized to 0 for all points of the initial interface) and  $F(p)$  is the speed function. The max-operation in the denominator avoids the division by zero ( $\varepsilon$  is a small number). Furthermore, considering the level set function as a mapping of the set membership of each point (i.e. the points of the interface are represented by the value 0, interior points by -1, and exterior ones by 1), the need of its periodical reinitialization vanishes. Due to this simplification, the curvature of the interface can be roughly approximated in an incremental manner. These ideas in conjunction with a heap-sorted queue for the departure times of interface points result in a fast algorithm for tracking implicit contours. We refer the reader to the original paper [14] for further details.

#### 3.2 Modifications to the Nilsson and Heyden Algorithm

This section explains how to modify the Nilsson and Heyden algorithm in order to obtain an efficient numerical algorithm for (4). Remind that this algorithm only evolves the interface and was proposed originally for tracking gradient-based implicit active contours [2, 3], in which the speed at each interface point only depends on its local neighbourhood. But the speed of the interface in the normal direction given, according to (4), as

$$F = \mu\kappa - \lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2 \quad (7)$$

has a global character. A local change of the interface in one point can change the global values  $c_1$  and  $c_2$ , that would require recomputing the speed function at all interface points. Therefore, storing the interface points in the heap-sorted queue ceases to be efficient. Recomputing the speed at all interface points leads to rebalancing the whole heap-sorted queue.

Similarly to the work by Shi and Carl [18], the proposed algorithm stores the interface points in a list data structure. Furthermore, we modify the local scope of one iteration of the Nilsson and Heyden algorithm. Instead of propagating only one interface point with the minimal departure time, the proposed algorithm propagates all the interface points in each iteration. Due to this modification, the arrival and departure times can no longer be considered. At the beginning of each iteration, the values  $c_1$  and  $c_2$  are recomputed and the speed at each interface point is evaluated according to [7]. Subsequently, the original local propagation of each interface point is performed. Note that the local propagation of each interface point allows for the values  $c_1$  and  $c_2$  to be recomputed efficiently, since we know exactly which points move to the inside and outside of the interface. Therefore, the values  $c_1$  and  $c_2$  can be updated in an incremental manner similarly to the interface curvature.

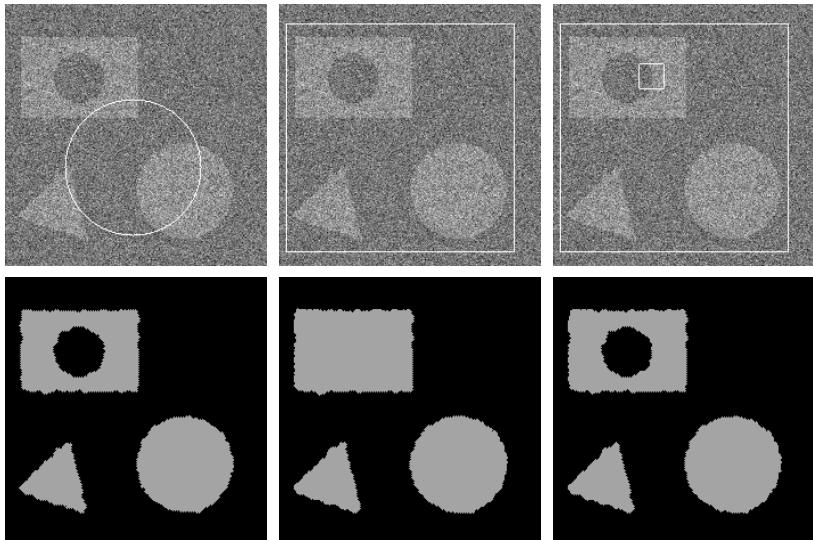
### 3.3 Limitations

Before we demonstrate the potential of the proposed algorithm, a few remarks on its limitations should be mentioned. First, it does not achieve subpixel accuracy due to the simplified representation of the implicit function. However, this is usually not a critical issue for segmentation purposes, since images are discretized in rectangular grids as well. Second, the original Chan-Vese algorithm has the tendency to compute a global minimizer [4], while the proposed algorithm acts only as a local minimizer, since the interface only is evolved. Therefore, new internal contours cannot come up and the final contour depends much more on the initialization.

## 4 Experimental Results

In this section, we present several results and comparisons on both synthetic and real image data to demonstrate the potential of the proposed algorithm. The experiments have been performed on a common workstation (Intel Core2 Duo 2.0 GHz, 2 GB RAM, Windows XP Professional).

We start with a noisy synthetic image of size  $256 \times 256$  pixels containing basic geometric objects (Fig. II) to demonstrate the dependency of the final contour on the initialization. The optimal segmentation should contain three objects – a rectangle with a circular hole, a triangle, and a circle. The main problem is to detect the circular hole inside the rectangle. If the initial contour does not cross the hole itself or at least the rectangle, there is no way how to propagate the contour inside the rectangle to detect the hole, since the proposed algorithm only evolves the interface. Quite bumpy results are caused by both the high level



**Fig. 1.** Dependency on the initialization. Top row: Initial contours. Bottom row: Segmentation results for  $\mu = 0.3$ ,  $\lambda_1 = \lambda_2 = 1$ .

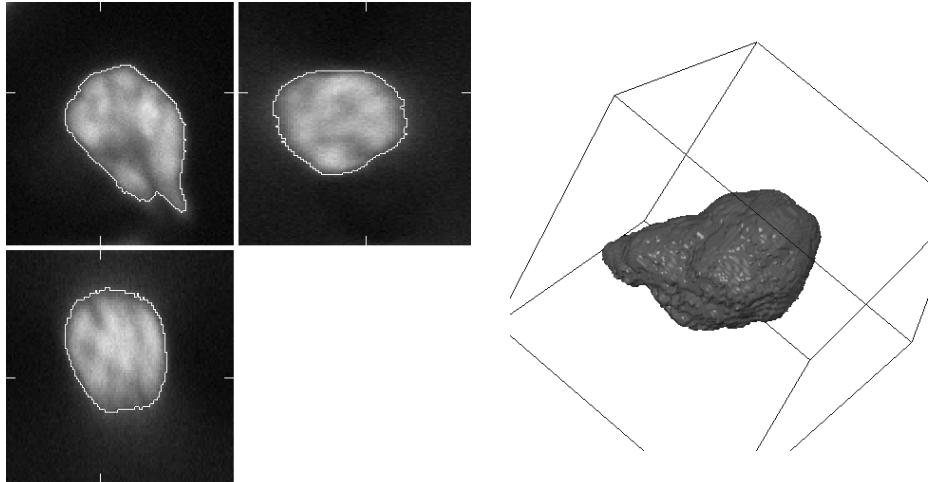
of noise in the image and the rough approximation of the curvature used in the proposed algorithm. The execution time was about 0.09 seconds in average.

In contrast to [6, 9] considering that  $\lambda_1 = \lambda_2$ , the proposed algorithm is not limited by such a constraint. A possible application, in which applying different weights for the foreground and background data terms could be profitable, is depicted in Fig. 2 showing different segmentation of a MRI brain scan of size  $256 \times 256$  pixels. The same weights ( $\lambda_1 = \lambda_2 = 1$ ) results only in the separation of the brain tissue from dark background, whereas the higher weight for the image foreground ( $\lambda_1 = 10$ ,  $\lambda_2 = 1$ ) leads to the detection of the white matter. The computation took 0.08 and 0.13 seconds, respectively.

We conclude this section with a comparison of the proposed algorithm with two other fast methods minimizing the Chan-Vese model. For this purpose, the



**Fig. 2.** Segmentation of a MRI brain scan ( $\mu = 0.4$ ). Left: Initial contour. Centre: Segmentation result for  $\lambda_1 = \lambda_2 = 1$ . Right: Segmentation result for  $\lambda_1 = 10$ ,  $\lambda_2 = 1$ .



**Fig. 3.** Segmentation of 3D cell nucleus ( $\mu = 0.1$ ,  $\lambda_1 = \lambda_2 = 1$ ). Left: Three orthogonal cross-sections of the input image overlaid with the final contour. Ticks show the positions of the other two cross-sections. Right: Rendered 3D model of the final contour.

**Table 1.** Execution time and memory consumption of the hybrid algorithm, graph cuts, and the proposed algorithm for both 2D as well as 3D images. Non-applicable entries are denoted by the symbol NA.

Image	Size	CPU (sec)			RAM (MB)		
		Hybrid	GC	Proposed	Hybrid	GC	Proposed
Geometric objects	256 × 256	0.56	0.08	0.09	15	20	13
Brain tissue	256 × 256	0.16	0.06	0.08	15	20	13
White matter	256 × 256	NA	0.08	0.13	NA	20	13
Cell nuclei	654 × 762	2.22	0.44	0.42	23	62	20
	1316 × 1035	7.89	1.29	1.34	37	144	30
	120 × 128 × 76	7.59	0.59	0.79	34	81	28
	425 × 303 × 60	83.09	4.44	6.34	143	447	116
	559 × 446 × 70	375.25	13.26	14.47	305	990	239

hybrid algorithm [9] combining the  $k$ -means clustering [7] with the threshold dynamics [6] and the algorithm based on graph cuts [23] were chosen. We focused on two objectives – the execution time and memory consumption. Several 2D and 3D images of different sizes were considered in this comparison. They were acquired using a fluorescence microscope and contain nuclei of HL60 cell line. To ensure the objectivity of results, all the compared methods were run with the same parameters. An example of segmentation results is depicted in Fig. 3. The measured values of the execution time and memory consumption are listed in Table 1. The symbol NA denotes non-applicable entries, i.e. scenarios in which the hybrid algorithm cannot be applied, since it only supports the same weights for the foreground and background data terms to be used.

## 5 Discussion

The final evaluation of the proposed algorithm is introduced in this section. We discuss, namely, the experimental results presented in Sect. 4 in detail.

The compared methods produce visually similar segmentation results. The Hausdorff distance of the segmentation results was at most two pixels (voxels) with respect to the city-block metric. However, they differ significantly in the execution time and memory consumption. In comparison to the hybrid algorithm [9], the proposed algorithm is about one order of magnitude faster. The difference is considerable, in particular, for larger 3D images for which the speed-up factor is more than 25. Furthermore, the proposed algorithm is also slightly more memory-efficient, from about 15 up to 20 percent, than the hybrid one. Compared with the graph cuts, the proposed algorithm is from about 10 up to 20 percent slower. However, the difference is in the order of seconds at most, as illustrated in Table II. On the other hand, the proposed algorithm has significantly lower memory demands than the graph cuts. The measured values indicate that the proposed algorithm is about four times more memory-efficient than the graph cuts. Note that for large images of size about  $1024 \times 1024 \times 100$  voxels, that are commonly produced by various biomedical scanners and microscopes as well, the proposed algorithm consumes only about 2 GB of memory in comparison to 8 GB consumed by the graph cuts, which may limit analyses of such large images using the graph cuts on common workstations.

Since the proposed algorithm only behaves as a local minimizer, careful attention has to be paid to the initialization. In general, no initialization technique is optimal for an arbitrary scenario. It is specific to a particular application and depends on the complexity of objects to be detected. In addition to the initial contours placed manually, we used simple unimodal thresholding producing a satisfactory initialization for the segmentation of cell nuclei.

For future work, we suggest several straightforward extensions of the proposed algorithm. Simultaneous tracking of multiple region-based implicit contours, as proposed in [20], or imposing topology-preserving constraints on the evolving contours could be profitable, namely, in biomedical applications. The preliminary results of our ongoing research are very promising in this area.

## 6 Conclusion

We have modified a fast level set-like algorithm by Nilsson and Heyden intended for tracking gradient-based active contours to obtain a fast algorithm for tracking region-based active contours driven by the Chan-Vese model. Instead of storing the interface points in a heap-sorted queue and a pointwise scheduled propagation of the implicit contour, that have turned out to be inefficient for a global character of the speed function used in the Chan-Vese model, the proposed algorithm stores the interface points in a list data structure and propagates the whole interface in each iteration. The comparison of the proposed algorithm with two other fast methods showed its speed and low memory demands. On the other hand, it may converge to a local minimum depending on the initialization.

**Acknowledgments.** This work has been supported by the Ministry of Education of the Czech Republic (Projects No. MSM-0021622419, No. LC535 and No. 2B06052).

## References

1. Adalsteinsson, D., Sethian, J.A.: A fast level set method for propagating interfaces. *Journal of Computational Physics* 118(2), 269–277 (1995)
2. Caselles, V., Catté, F., Coll, T., Dibos, F.: A geometric model for active contours in image processing. *Numerische Mathematik* 66(1), 1–31 (1993)
3. Caselles, V., Kimmel, R., Sapiro, G.: Geodesic active contours. *International Journal of Computer Vision* 22(1), 61–79 (1997)
4. Chan, T.F., Vese, L.A.: Active contours without edges. *IEEE Transactions on Image Processing* 10(2), 266–277 (2001)
5. Deng, J., Tsui, H.T.: A fast level set method for segmentation of low contrast noisy biomedical images. *Pattern Recognition Letters* 23(1-3), 161–169 (2002)
6. Esedoglu, S., Tsai, Y.H.R.: Threshold dynamics for the piecewise constant Mumford-Shah functional. *Journal of Computational Physics* 211(1), 367–384 (2006)
7. Gibou, F., Fedkiw, R.: A fast hybrid k-means level set algorithm for segmentation. In: *Proceedings of the 4th Annual Hawaii International Conference on Statistics and Mathematics*, pp. 281–291 (2005)
8. Goldenberg, R., Kimmel, R., Rivlin, E., Rudzsky, M.: Fast geodesic active contours. *IEEE Transactions on Image Processing* 10(10), 1467–1475 (2001)
9. Hubený, J., Matula, P.: Fast and robust segmentation of low contrast biomedical images. In: *Proceedings of the 6th IASTED International Conference on Visualization, Imaging and Image Processing*, pp. 189–196 (2006)
10. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. *International Journal of Computer Vision* 1(4), 321–331 (1987)
11. Lie, J., Lysaker, M., Tai, X.C.: A binary level set model and some applications to Mumford-Shah image segmentation. *IEEE Transactions on Image Processing* 15(5), 1171–1181 (2006)
12. Maška, M., Hubený, J., Svoboda, D., Kozubek, M.: A comparison of fast level set-like algorithms for image segmentation in fluorescence microscopy. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Paragios, N., Tanveer, S.-M., Ju, T., Liu, Z., Coquillart, S., Cruz-Neira, C., Müller, T., Malzbender, T. (eds.) *ISVC 2007, Part II. LNCS*, vol. 4842, pp. 571–581. Springer, Heidelberg (2007)
13. Mumford, D., Shah, J.: Optimal approximation by piecewise smooth functions and associated variational problems. *Communications on Pure and Applied Mathematics* 42(5), 577–685 (1989)
14. Nilsson, B., Heyden, A.: A fast algorithm for level set-like active contours. *Pattern Recognition Letters* 24(9-10), 1331–1337 (2003)
15. Osher, S., Fedkiw, R.: *Level Set Methods and Dynamic Implicit Surfaces*. Springer, New York (2003)
16. Osher, S., Sethian, J.A.: Fronts propagating with curvature dependent speed: Algorithms based on Hamilton–Jacobi formulation. *Journal of Computational Physics* 79(1), 12–49 (1988)
17. Sethian, J.A.: A fast marching level set method for monotonically advancing fronts. *Proceedings of the National Academy of Sciences* 93(4), 1591–1595 (1996)

18. Shi, Y., Karl, W.C.: A real-time algorithm for the approximation of level-set-based curve evolution. *IEEE Transactions on Image Processing* 17(5), 645–656 (2008)
19. Tsai, A., Yezzi, A., Willsky, A.S.: Curve evolution implementation of the Mumford-Shah functional for image segmentation, denoising, interpolation, and magnification. *IEEE Transactions on Image Processing* 10(8), 1169–1186 (2001)
20. Vese, L.A., Chan, T.F.: A multiphase level set framework for image segmentation using the Mumford and Shah model. *International Journal of Computer Vision* 50(3), 271–293 (2002)
21. Wang, X.F., Huang, D.S., Xu, H.: An efficient local chan-veese model for image segmentation. *Pattern Recognition* 43(3), 603–618 (2010)
22. Whitaker, R.T.: A level-set approach to 3D reconstruction from range data. *International Journal of Computer Vision* 29(3), 203–231 (1998)
23. Zeng, Y., Chen, W., Peng, Q.: Efficiently solving the piecewise constant Mumford-Shah model using graph cuts. Tech. rep., Department of Computer Science, Zhejiang University, China (2006)

# A Novel Hardware Architecture for Rapid Object Detection Based on Adaboost Algorithm

Tinghui Wang<sup>1</sup>, Feng Zhao<sup>1</sup>, Jiang Wan<sup>1</sup>, and Yongxin Zhu<sup>2</sup>

<sup>1</sup> Digilent Electronic Technology Co. Ltd.

{steve.wang,frank.zhao,jiang.wan}@digilentchina.com

<sup>2</sup> Shanghai Jiaotong University

zhuoyongxin@sjtu.edu.cn

**Abstract.** This paper proposed a novel hardware architecture for rapid object detection based on Adaboost learning algorithm with Haar-like features as weak classifiers. A 24x24 pipelined integral image array is introduced to reduce calculation time and eliminate the problem of the huge hardware resource consumption in integral image calculation and storage. An expansion of the integral image array is also proposed to increase the parallelism at a low cost of hardware resource consumption. These methods resulted in an optimized detection process. We further implemented the process on Xilinx XUP Virtex II Pro FPGA board, and achieved an accuracy of 91.3%, and a speed of 80 fps at clock rate of 100 MHz, for 352x288 CIF image.

## 1 Introduction

As the first step of object recognition with applications in recognizing human faces, detecting pedestrians and cars, etc. in 2D images, object detection is always a hot and challenging topic - the algorithm should be robust and the procedure should be as fast as possible. Recent years, FPGA-based implementation of object detection algorithm was introduced to accelerate the procedure. Researchers have designed state-of-the-art hardware architectures for object detection using various methods, which are classified into four categories: knowledge-based, feature invariant, template matching and appearance-based [1].

Proposed by Viola and Jones [2], rapid object detection based on Adaboost classification algorithm with Haar-like features is one of the most popular algorithms used in hardware implementation, as the method can achieve a high accuracy as well as a fast speed. With the help of cascade structure of classifiers and a new image representation – integral image, very fast feature calculation and object detection can be achieved.

Current state-of-the-art object detection system can achieve a detection rate for real-time videos and cameras with approximately 30 frames per second, usually demonstrated on face detection application. As we want to integrate more complicated function into the system, it needs to be much faster. Most of the literatures these years mainly focused on the optimization of feature calculation and cascade structure of classifiers, since it is thought to be the most time

consuming part in the detection system if the input window contains a face. The trade-off between detection speed, stage and classifier amount, and detection rate is further explained by Viola<sup>[3]</sup> in 2004. Recently, Hiromoto<sup>[4]</sup> also discussed about the parallelism of cascade classifier structure at a system level.

However, we would like to argue that the pre-processing stage for cascade classifiers, such as integral image calculation, is the real time consuming part, especially for the frames that there is not a face. Actually, a high parallelism is needed in pre-processing stage instead of the classification part. There are literatures in recent years that try to accelerate the integral image calculation. Theocharides<sup>[5]</sup> proposed a structure called CDTU (Collection and Data Transfer Unit) array to improve the calculation, but CDTUs consume massive hardware resources, so that it is not possible to be implemented on FPGA chips. The simulation result in their paper reported that they obtained a rough estimate of 52 frames per second targeting 500 MHz clock cycle. Lai<sup>[6]</sup> proposed a piped register module for integral image calculation with 640 columns and 22 rows. According to their report based on their face detection system, they achieved a detection rate of 130 frames per second, however, with only 52 weak classifiers, a great sacrifice in accuracy. Researchers have also tried other means to realize the same algorithm. Hefenbrock<sup>[7]</sup> proposes a multi-GPU implementation for the face detection system, which achieved 15.2 FPS with 4 GPUs.

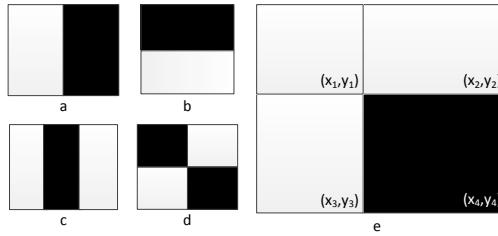
In this paper, we adopt a piped integral image calculation array, which eliminates the massive memory consumption of integral image and reduces the amount of the data needed to be transmitted between off-chip memory and the detection system. Motivated by and demonstrated on the task of face detection, we implemented a face detection system based on training data of OpenCV with 25 stages and 2913 features. Tested on 16000 pictures in Shanghai IsVision Sample Database, we achieved a detection rate of 80 frames per second on Virtex-II Pro XUP FPGA Board at 100 MHz clock rate.

In the next section, we will review the rapid object detection algorithm proposed by Viola. Then we analyze and optimize the detection procedure, and elaborate the hardware architecture of detection system, followed by the implementation statistics and test results of the face detection system we implemented on Xilinx XUP Virtex-II Pro XUP FPGA board. Section V gives our conclusion.

## 2 Rapid Object Detection Algorithm

Rapid object detection based on Adaboost algorithm is first proposed by Viola<sup>[2]</sup>. Adaboost algorithm is a machine learning algorithm that builds a strong classifier by assigning proper weights to weak classifiers. The weak classifier are built based on Haar-like features, which are a set of simple features shown in Fig. II(a to d). They get their name from their intuitive similarity of Haar wavelets. The result of Haar-like feature function is the difference between the summation of pixels in black region and the summation of pixels in white region.

In order to speed up the Haar-like feature function calculation, Viola proposed a new image representation called Integral Image, also known as Summed Area



**Fig. 1.** Basic Type of Haar-like Features (*a,b* - two rectangles; *c* - three rectangles; *d* - four rectangles; *e* - SAT calculation)

Table (SAT).  $SAT(x, y)$  of some position  $(x, y)$  is the summation of all the pixels in the left-up plane of current location  $(x, y)$ :

$$SAT(x, y) = \sum_{\substack{0 \leq x' \leq x \\ 0 \leq y' \leq y}} Pixel(x', y') \quad (1)$$

With the advantage of summed area table, the summation of pixels in a rectangle (the black area in Fig. 1(e)) needs only four data references:

$$\begin{aligned} \sum_D P(x, y) = & SAT(x_1, y_1) + SAT(x_4, y_4) \\ & - SAT(x_3, y_3) - SAT(x_2, y_2) \end{aligned} \quad (2)$$

Weak classifiers are built based on these Haar-like features. The hypothesis of weak classifiers

$$h(x, f, p, \theta) = \begin{cases} 1 & \text{if } p \cdot f(x) < p \cdot \theta \\ 0 & \text{otherwise,} \end{cases} \quad (3)$$

where  $h(x, f, p, \theta)$  is the weak classifier function,  $p$  is the polarity which indicated the direction of inequality,  $x$  is the pixel array of sub-window,  $f$  is the feature function described above,  $\theta$  is the threshold assigned with some simple machine learning algorithm during training.

The final strong classifier is

$$C(x) = \begin{cases} 1 & \sum_{t=1}^T w_t \cdot h_t(x) \geq \theta_t \\ 0 & \text{otherwise,} \end{cases} \quad (4)$$

where  $x$  is a  $24 \times 24$ -pixels sub-window,  $C(x)$  is the final result of the strong classifier,  $w_t$  is the weight assigned to the  $t^{\text{th}}$  classifier,  $h_t(x)$  is the weak classifier function and  $\theta_t$  is the threshold assigned to each classifier.

Viola also introduced a cascade structure of classifiers to accelerate the object detection procedure. The final strong classifier is divided into stages, while each stage is a boosted strong classifier. For the first several stages, the classifiers are trained to reject most of the negative image areas, while the following stages

are composed of more complex classifiers that are trained to achieve low false positive rate. Usually, there is only several faces in millions of sub-windows. The cascade architecture can speed up the whole detection process dramatically, as most false sub-windows can be filtered out after one stage or two, and only the sub-windows that contain faces need to go through all the stages.

### 3 Analysis and Optimization of Detection Procedure

Current object detection system based on cascade structure of Haar-like feature classifiers follows a detection procedure as follows: Firstly, the integral image of the whole frame is calculated and stored in off-chip memory; Then a sub-window (usually 24x24 pixels) traverse through the whole frame; Run the detection classifiers on all the sub-windows; Down-scale the image and repeat the above procedure again; And finally, output the sub-windows that contain the target object.

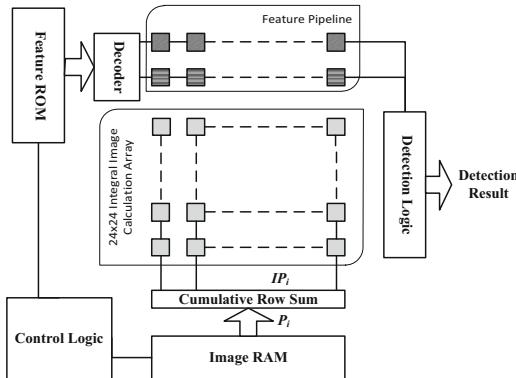
For most sub-windows, they do not contain the object. Due to the cascade structure of classifiers, they need not go through all the stages before the correct classification is done. According to Viola, the first stage of cascade classifiers, which contains only 2 features in Viola's design, can rejects about 50% of non-face sub-windows and the next stage can reject 80% of non-faces while detecting almost 100% of faces. A large part of the time consumed by these sub-windows is wasted on the calculation of integral image and data exchange between off-chip memory and the detection circuit.

To process a sub-window of 24x24 pixels, a minimum memory of 10386 bits (about 1K Bytes, 17 bits per pixel times 576 pixels) are needed. Based on this analysis, we designed an integral image calculation array, elaborated in next section, that can process the sub-window in one clock cycle with minimum hardware resource consumption. Due to the pipe-lined nature of this structure, only 24 bytes of data (24 pixels) are needed to be transmitted from off-chip memory to detection system at a time, which happen simultaneously with the classification process of last sub-window. The detection procedure is modified according to hardware design as follows:

- Step 1.** Pick a sub-window ( $24 \times 24$  in our demonstration application on face detection) as a potential candidate of the object to be detected.
- Step 2.** Calculate the integral image of the sub-window.
- Step 3.** Apply the strong classifier to the sub-window to determine whether the sub-window contains the object we want to detect.
- Step 4.** If all sub-windows (potential areas of the target object) have been checked, down-scale the image by a scale factor (1.2 in our system) and move to next step. Otherwise, pick the next sub-window (usually move the sub-window 1 pixel downward or move 1 pixel right if it reaches the end of a column and start from the top) and go back to step 4 again.
- Step 5.** If the size of down-scaled graph is smaller than the detectable resolution, the system outputs the result. Go to step 2 otherwise.

## 4 Hardware Architecture

The block diagram of our architecture is shown in Fig. 2. Our system consists of six major blocks: Control Logic, Image RAM, Feature ROM, Feature Pipeline, Integral Image Calculation Array, and Feature Detection Logic. Current video frame that is to be detected is stored in Image RAM, and the parameters of cascade classifiers are stored in Feature ROM. When the system starts to detect the target object, pixels of the sub-window are fed, line by line, into the piped Integral Image Calculation Array, where the integral image is accomplished within one clock cycle. After the integral image of the sub-window is generated, features are pumped into the Feature Pipeline, calculated, and then compared with the threshold of the stage. If the sub-window passed all stages, a decision that the sub-window contains the object will be made. Otherwise, the sub-window is thought not containing the target object.

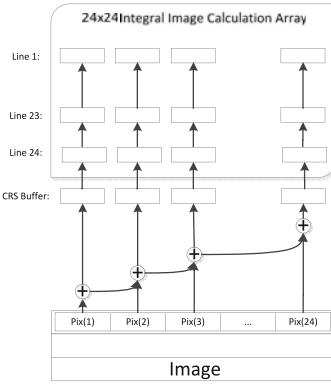


**Fig. 2.** Hardware Architecture Overview

### 4.1 Piped Integral Image Calculation Array

As the key contribution of this paper, the piped integral image array we designed is shown in Fig. 3. The array calculates a new integral image based on the old integral image in the array by taking in a new row of 24 pixels wide as the sub-window moves downward in the image being processed.

To build a piped integral image calculation array, we need to find the relationship between the old integral image with the new integral image after a line of 24 pixels enters the sub-window. Here, we use  $x$  to denote the row index while  $y$  to denote the column index.  $P(x, y)$  is the pixel value of the past sub-window, while  $P'(x, y)$  is the pixel value of the new sub-window, whose integral image is about to be calculated.  $SAT(x, y)$  is the old summed area table and  $SAT'(x, y)$  is the new table to be calculated.



**Fig. 3.** Piped Integral Image Calculation Array

From the definition of notations:

$$P'(x, y) = P(x + 1, y), \text{ for } 1 \leq x \leq 23 \quad (5)$$

Define the Cumulative Row Summation (CRS) of the row  $x$  as

$$CRS(y) = \sum_{j=1}^y P'(x, j) \quad (6)$$

For the first 23 lines of the integral image to be calculated:

$$\begin{aligned} SAT'(x, y) &= \sum_{i=1, j=1}^{x, y} P'(i, j) \\ &= \sum_{m=2; n=1}^{x+1; y} P(m, n) \\ &= \sum_{m=1; n=1}^{x+1; y} P(m, n) - \sum_{n=1}^y P(1, n) \\ &= SAT(x + 1, y) - SAT(1, y) \end{aligned} \quad (7)$$

The result shows that the values of the first 23 lines in the new summed area table can be calculated by subtracting values of line 1 from line 2 to 24 in the old integral image and then shift the array one line up.

For the last line of the new integral image,

$$\begin{aligned} SAT'(24, y) &= \sum_{i=1; j=1}^{23; y} P'(i, j) + \sum_{j=1}^y P(24, j) \\ &= SAT(24, y) - SAT(1, y) + CRS(y) \end{aligned} \quad (8)$$

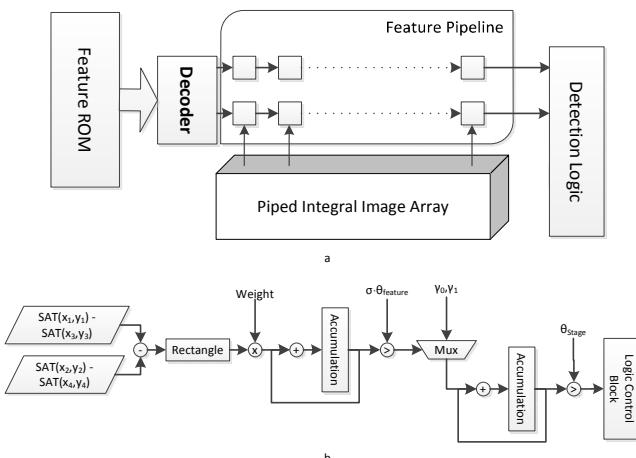
The result means that the last line of the new integral image can be derived by adding the accumulative sum of the new row with the last row and then subtracted by the first row in old integral image. Thus the array works as follows:

- Step 1.** Subtract the first row from all the rest rows in the integral image calculation array.
- Step 2.** Calculate the cumulative row sum (CRS) of the new row and store the results in the CRS buffer.
- Step 3.** Add the last row (row 24) to the CRS buffer.
- Step 4.** Shift one line up and we get the new integral image.

All these four steps, all arithmetic operations (add and minus), can be integrated into a combined logic circuit. If the time interval is long enough for the propagation of the logic circuit, the only clock signal we need here is the shift operation in the last step.

## 4.2 Parallel Pipelined Feature Calculation

In order to utilize the integral image stored in the array and accelerate the detection speed, a Feature Pipeline with 24 cells is introduced (Fig. 4a). Each cell is connected to the corresponding column of the array through the data path (with multiplexer and control logic). Each cell contains the information of the rectangle of the features to be calculated and shifts rightwards. When the first match of column number between integral image calculation array and rectangle information inside the cell occurs, the integral image value of two vertexes are extracted from the array and a subtraction operation is applied. Another subtraction of the other two vertexes is applied when the second match occurs. As in Equation 2, the summation of the pixels in the rectangle is the difference between the above two subtraction results.



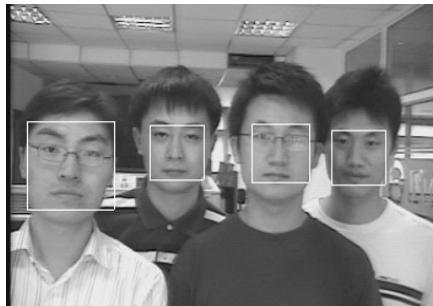
**Fig. 4.** Pipelined Feature Calculation (a - feature pipeline structure; b - Detection Logic)

After the data of one feature propagates through the 24 cells, the result of feature function is acquired and then passed on to the feature detection logic block (as shown in Figure 4b), where the result of each feature is compared with a feature threshold ( $\theta_{feature}$ ) to select the positive or the negative contributor ( $\gamma_1/\gamma_0$ ). The contributors are then summed up and compared with a stage threshold ( $\theta_{stage}$ ). If the sub-window passes all the classification stages, it is recognized as the object and vice versa.

The architecture is also expandable to achieve a higher parallelism with a rather small increase in hardware resource consumption. Two subwindows can be processed simultaneously if we add one line to integral image array and another feature pipeline to the system. As a result, the speed of detection can be doubled when the hardware resource consumption just goes up for about 5%. Moreover, the speed of our system is accelerated by 8 times due to the adoption of Float-Point-to-Fixed-Point-Transition for all the parameters.

## 5 Results

We designed a face detection system using Verilog HDL based on the hardware object detection architecture proposed above and implemented it on Xilinx Virtex-II Pro XUP FPGA Board. In the implementation, we utilized the classifier parameters of Open Computer Vision Library (OpenCV), who trained a strong frontal face classifier with 25 stages and 2913 weak classifiers. We also implemented a video camera on our FPGA board with an input of CIF format ( $352 \times 288$ , Y/Cb/Cr). The detection result is transmitted out directly through SVGA port. The result is shown in Fig. 5. We are working on a colorful version for future demonstration.



**Fig. 5.** Example of Face Detection

Our system achieved a detection rate of 91.3% and a false positive rate of 3.2% while it was tested on 16000 pictures of Shanghai IsVision Sample Database (5000 positive and 11000 negative). It also achieved an average detection speed of 80 fps when tested lively with video camera as the input. The system is quite fast compared to the system reported in the previous literature, such as 15 frames

per second for  $120 \times 120$ -pixel input image on the same platform [8], and 30 frames per second for CIF image input on Virtex-V LX330 platform [4]. The details of our test result are shown in Table 1.

**Table 1.** Test results for V2Pro Implementation

Platform	Xilinx XUP V2 Pro
Clock Frequency	100 MHz
Input Image Resolution	$352 \times 288$
Detection Rate	91.3%
False Positive Rate	3.2%
Detection Speed	80 <i>fps</i>

Moreover, we try to achieve a high detection rate and detection speed consuming a minimum amount of hardware resource as well. In our system, we consumed 12438 Slices and 15749 4-input LUTs on Virtex-II FPGA chips (as shown in Tab 2), compared with 26604 6-input LUTs in [4] and 32438 6-input LUTs in [9].

**Table 2.** System Hardware Resource Consumption

Logic Utilization	Used	Available	Utilization
Slices	12438	13696	90%
Slice Flip Flops	10571	27392	38%
4-input LUTs	15749	27392	57%
Block RAMs	87	136	63%

Compared with the multi-GPU design proposed by Hefenbrock [7], we achieved a better detection speed with a lower cost and lower power consumption as well. As we almost utilized all the resources available on XC2VP30 FPGA chip on Virtex-II Pro Development System, the whole board costs 399 USD (academic price) to compared to 1599 USD (industry price) compared to 300 USD to 500 USD for a single GTX 285 GPU as reported in [7]. As argued in section three, the parallel processing of multiple subwindows is achievable with a small increase in hardware resource consumption.

## 6 Conclusion

In this paper, we proposed a novel hardware architecture for object detection based on Adaboost Algorithm with Haar-like feature as classifiers. In order to speed up the detection procedure, especially for sub-windows that do not contain a face, and minimize the memory consumption, we designed a piped integral image calculation array, implemented a feature calculation pipeline and proposed

a parallel processing structure of multiple subwindows. The face detection system we implemented on Virtex-II Pro XUP FPGA Board based on the proposed architecture achieved a detection speed of 80 fps with CIF image input and a detection rate of 91.3%.

## Acknowledgement

Special thanks to Shanghai IsVision Technologies Co., Ltd for providing database of test samples.

## References

1. Yang, M., Kriegman, D., Ahuja, N.: Detecting faces in images: A survey. *IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI)* 24, 34–58 (2002)
2. Viola, P., Jones, M.: Rapid object detection using a boosted cascade of simple features. In: *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2001*, vol. 1, pp. I-511–I-518 (2001)
3. Viola, P., Jones, M.: Robust real-time face detection. *International Journal of Computer Vision* 57, 137–154 (2004)
4. Hiromoto, M., Sugano, H., Miyamoto, R.: Partially parallel architecture for adaboost-based detection with haar-like features. *IEEE Trans. Circuits Syst. Video Techn.* 19, 41–52 (2009)
5. Theocarides, T., Vijaykrishnan, N., Irwin, M.: A parallel architecture for hardware face detection. In: *IEEE Computer Society Annual Symposium on VLSI (ISVLSI)*, pp. 452–453. IEEE Computer Society, Los Alamitos (2006)
6. Lai, H.-C., Savvides, M., Chen, T.: Proposed fpga hardware architecture for high frame rate (?100 fps) face detection using feature cascade classifiers. In: *First IEEE International Conference on Biometrics: Theory, Applications, and Systems (BTAS 2007)*, pp. 1–6 (2007)
7. Hefenbrock, D., Oberg, J., Thanh, N.T.N., Kastner, R., Baden, S.B.: Accelerating viola-jones face detection to fpga-level using gpus. In: *Annual IEEE Symposium on Field-Programmable Custom Computing Machines*, pp. 11–18 (2010)
8. Wei, Y., Bing, X., Chareonsak, C.: Fpga implementation of adaboost algorithm for detection of face biometrics. In: *IEEE International Workshop on Biomedical Circuits and Systems (BIOCAS)*, pp. S1/6–17–20 (2005)
9. Cho, J., Benson, B., Mirzaei, S., Kastner, R.: Parallelized architecture of multiple classifiers for face detection. In: *IEEE International Conference on Application-Specific Systems, Architectures and Processors*, pp. 75–82 (2009)

# Using Perceptual Color Contrast for Color Image Processing

Guangming Xiong<sup>1</sup>, Dah-Jye Lee<sup>2</sup>, Spencer G. Fowers<sup>2</sup>,  
Jianwei Gong<sup>1</sup>, and Huiyan Chen<sup>1</sup>

<sup>1</sup> School of Mechanical Engineering, Beijing Institute of Technology, Beijing, China

<sup>2</sup> Dept. of Electrical and Computer Engineering, Brigham Young University,  
Provo, Utah, USA

**Abstract.** Many grayscale image processing techniques such as edge and feature detection, template matching, require the computations of image gradients and intensity difference. These computations in grayscale are very much like measuring color difference between two colors. The goal of this work is to determine an efficient method to represent color difference so that many existing grayscale image processing techniques that require the computations of intensity difference and image gradients can be adapted for color without significantly increasing the amount of data to process and without significantly altering the grayscale-based algorithms. In this paper, several perceptual color contrast measurement formulas are evaluated to determine the most applicable metric for color difference representation. Well-known edge and feature detection algorithms using color contrast are implemented to prove its feasibility.

## 1 Introduction

Recently, many researchers use contrast information as a measure for color difference. Mojsilovic et al. computed color difference for colors from the Fibonacci palette and extended gradient-based grayscale edge detectors to process color-mapped images [1]. Chen et al. adopted another form of contrast, which is defined over the CIE uniform color space and is closely related to human visual perception [2]. Similar to Chen's work, Liu proposed a color-edge detection method based on discrimination of noticeable color contrasts [3]. Moreno et al. presented a new method for color edge detection based on tensor voting framework in which perceptual color differences were estimated by means of an optimized version of the CIEDE2000 formula [4]. Chou and Liu used the CIEDE2000 color difference equation to measure the perceptual redundancy inherent in each wavelet coefficient of various color spaces [5]. Fondón et al. employed CIEDE2000 to build a number of distance images which represent the similarity between reference colors and the other colors present in the image for segmentation of skin cancer images [6]. Li et al. used CIEDE2000 to calculate color difference between two pixels and apply it to Sobel operator [7]. Because color distance or difference between two color points at any part of a uniform color space corresponds to the perceptual difference between the two colors by the human vision system, computing image gradients in grayscale is very much like measuring color difference in color image.

Many grayscale image processing techniques such as edge detection and block matching require the computation of image gradients or grayscale difference. The ultimate goal of this work is to determine an efficient method to represent rich color information so that existing grayscale image processing techniques can be adapted for color image processing without significantly increasing the amount of data to process and without significantly altering the grayscale-based algorithms. More specifically, the main concept is to represent color in a UCS (Uniform Color Space) and use color contrast measurement directly for image gradients and difference computations.

The rest of this paper is organized as follows. In Section 2, several perceptual color contrast measurement metrics are discussed. Some of them are selected for implementation and comparison. Section 3 presents the concept of color gradient and demonstrates its ability to measure color difference. The implementations of edge detection and feature detection and block matching algorithms using selected color contrast metrics are then presented in Sections 4, 5 and 6. Finally, the paper is concluded in Section 7.

## 2 Selections of Color Difference Formulas

A color space is considered a perceptually uniform color space if the Euclidean distance between two color points closely corresponds to the perceptual difference determined by the human vision system [8]. In 1976 the CIE recommended CIE Lab as an approximately uniform color space because of its Euclidean distance  $\Delta E^*_{76}$ .

$$\Delta E^*_{76} = \sqrt{(L_2^* - L_1^*)^2 + (a_2^* - a_1^*)^2 + (b_2^* - b_1^*)^2} \quad (1)$$

The CIE 1994 color difference equation (CIE94) which is defined in the L\*C\*h color space, is then given as

$$\Delta E^*_{94} = \left[ \left( \frac{\Delta L^*}{k_L S_L} \right)^2 + \left( \frac{\Delta C^*_{ab}}{k_C S_C} \right)^2 + \left( \frac{\Delta H^*_{ab}}{k_H S_H} \right)^2 \right]^{1/2} \quad (2)$$

$S_L$ ,  $S_C$ , and  $S_H$  (weighting functions) correct the lack of uniformity of CIE Lab space, while  $k_L$ ,  $k_C$ , and  $k_H$ , correct for the influence of experimental viewing conditions.

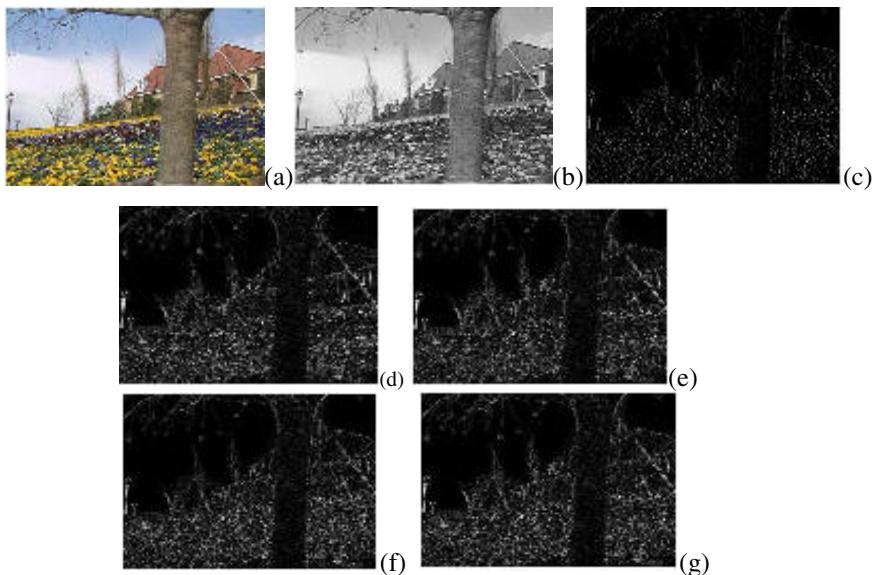
The latest color difference metric, CIEDE2000, was incorporated to precisely measure color difference in CIE Lab [9]. Cui et al. argued that CIEDE2000 does not include an associated uniform color space because it is not a Euclidean distance [10]. Based on similar concept, new Euclidean formulas including DIN99 [11], CIECAM02 [12], and GP color difference [13] formulas based on a number of variants of uniform OSA-UCS color space were developed.

It is worth mentioning that CIEDE2000 is currently viewed as the most advanced color difference formula since the aforementioned new formulas perform either at best the same or not quite as well as CIEDE2000 [10, 13]. Although these new formulas are considered much simpler than CIEDE2000, the conversion from CIE XYZ, a commonly used standard, to each of their own associated color space is much more complicated than the conversion to a standard CIE Lab color space. Taking GP

color difference formula as an example, it is a Euclidean distance and a comparatively simple equation in OSA-UCS space. However, it is complex to convert CIE XYZ to its associated OSA-UCS color space [13]. Similar to GP formula, the CIECAM02 formula, although relatively simple, the conversion to its associated CIECAM02 color space is not a straightforward computation task [12], which makes it not suitable for color image processing. In this work, only those formulas that are based on the standard CIE Lab color space including CIE76, CIE94, CIEDE2000, and DIN99D are implemented and evaluated.

### 3 Color Image Gradients

Computation of image gradients is the first step to performing many image processing algorithms. Computation of image gradients is equivalent to convolving a small mask with the input image. The simplest mask used for calculating gradients is [-1, 0, 1]. This mask can be used for both horizontal and vertical gradients for edge detection. In grayscale mode, it is basically performed by finding the grayscale difference between the pixel on the left and the pixel on the right of the center pixel for horizontal gradient or the grayscale difference between the pixel above and the pixel below the center pixel for vertical gradient. The same calculation can be applied to color image by calculating color difference between two pixels using perceptual color contrast.



**Fig. 1.** Horizontal gradients, (a) original color image, (b) original grayscale image, (c) grayscale gradient, (d) CIE76 (e) CIE94, (f) CIEDE2000, and (g) DIN99D

Figure 1 shows the horizontal gradients using the grayscale method and the four formulas for color difference calculation selected in Section 2. The gradient values are normalized to 0 to 255 for display. The gradient values are zero for image areas that

have uniform intensity. Figure 1 demonstrates that perceptual color contrast can be used for image gradients computation and it performs better than the grayscale difference. Instead of relying on visual inspection to determine the performance, a statistical method was used to quantify the result. We first compare the number of pixels that are considered zero gradients. A better gradient computation method should be more sensitive to intensity or color change and detect fewer zero gradient pixels. Histograms of the calculated gradients were calculated for comparison. Grayscale method is the least sensitive of all, detecting around 72,000 zero gradient pixels. CIE76 is the formula most suitable for gradient calculation, detecting less than 10,000 zero gradient pixels. All four color difference formulas perform better than grayscale.

## 4 Color Edge Operators

Our first example of using perceptual color contrast for image processing is edge detection. Two most commonly used edge detectors, Sobel and Canny, are converted for color edge detection to prove that perceptual color contrast allows us to use the original grayscale algorithms without the need of developing a complicated color version of the algorithms for these edge detectors. Our results show that the perceptual color contrast approach performs better or at least as well as the grayscale algorithms depending on the input image, especially in regions that have high color contrast but with very uniform grayscale values.

### 4.1 Color Sobel Operator

The Sobel operator calculates the horizontal and vertical gradients to determine edge magnitude or strength as

$$M = \sqrt{G_x^2 + G_y^2} \quad (3)$$

Where  $Gx_{i,j}$  and  $Gy_{i,j}$  denote the horizontal and vertical gradients of pixel  $(i, j)$  and are defined as

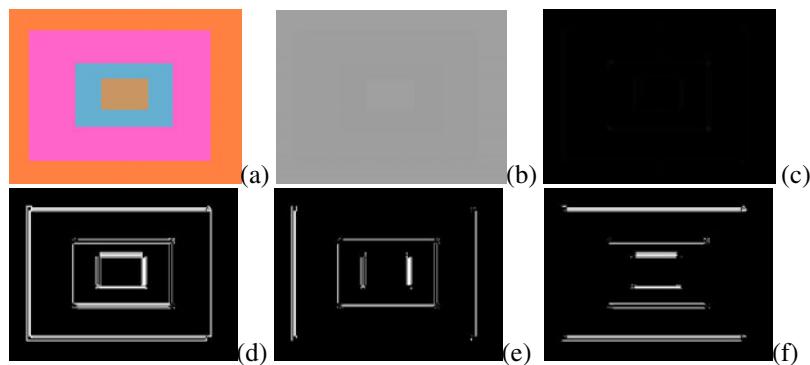
$$Gx_{i,j} = \Delta E(p_{i-1,j+1}, p_{i-1,j-1}) + 2\Delta E(p_{i,j+1}, p_{i,j-1}) + \Delta E(p_{i+1,j+1}, p_{i+1,j-1}) \quad (4)$$

$$Gy_{i,j} = \Delta E(p_{i-1,j-1}, p_{i+1,j-1}) + 2\Delta E(p_{i-1,j}, p_{i+1,j}) + \Delta E(p_{i-1,j+1}, p_{i+1,j+1}) \quad (5)$$

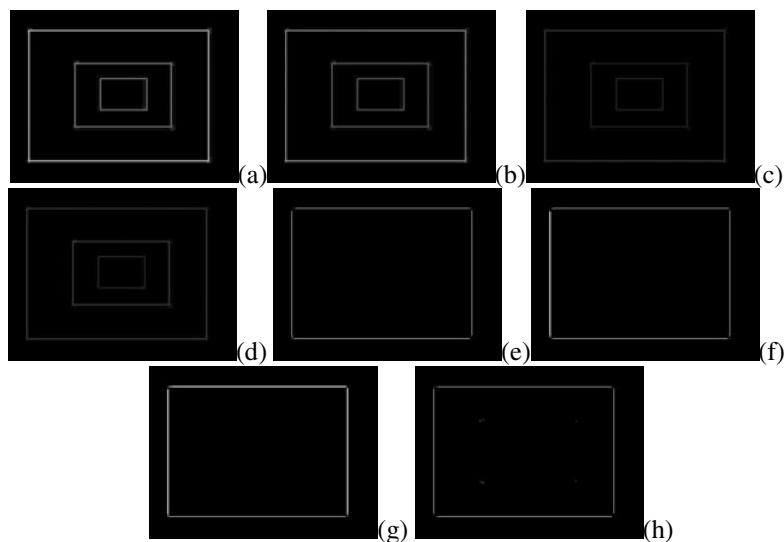
In grayscale mode,  $\Delta E$  represents the grayscale difference between the two pixels in the parenthesis. In color mode with the same equations,  $\Delta E$  represents the color difference between the two pixels calculated by a color-difference formula. We computed the color horizontal and vertical gradients  $Gx_{i,j}$  and  $Gy_{i,j}$  using all four selected formulas CIE76, CIE94, CIEDE2000, and DIN99D for comparison.

Figure 2 shows a simple computer-generated color pattern that has high color contrast (Figure 2(a)) but with fairly uniform grayscale values (Figure 2(b)) to illustrates the advantage of color Sobel methods over grayscale. The edge strength in grayscale (Figure 2(c)) can hardly be seen and makes edge detection almost impossible. Figures 2 (d), (e), and (f) show the binarized edge detection results using different thresholds. Since there is not much distinct grayscale difference between two

adjacent color rectangles, using grayscale Sobel to detect any particular rectangular boundary is not possible. On the contrary, color Sobel operator based on perceptual color contrast is able to obtain different edge strength for different rectangular boundary, which makes the detection of any particular rectangular boundary a relatively easy task. Figure 3 shows edge strength results using the four selected color-difference formulas. The outer (largest) rectangular boundary has the strongest edge then the two smaller ones. Any particular rectangular boundary can be easily detected by applying proper high and low thresholds. Figure 3 shows the detection result of the outer boundary. It is noted that CIE76 and CIE94 are able to distinguish color contrast (more distinct edge strength for different boundaries) better than CIE2000 and DIN99D.



**Fig. 2.** Edge detection of a simple color pattern, (a) original color pattern, (b) grayscale version of (a), (c) edge strength using grayscale Sobel, (d)–(f) edge detection using different thresholds



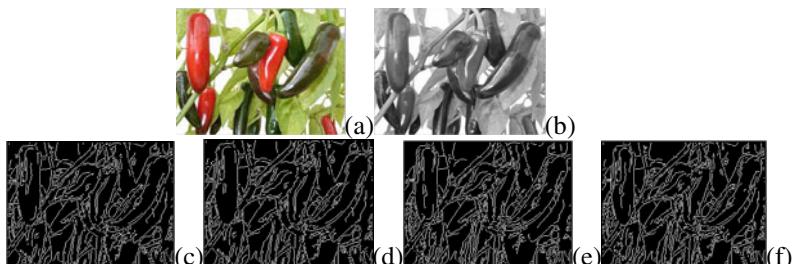
**Fig. 3.** Edge strength using (a) CIE76, (b) CIE94, (c) CIE2000, and (d) DIN99D and edge detection using (e) CIE76, (f) CIE94, (g) CIE2000, and (h) DIN99D

## 4.2 Color Canny Operator

Another example of using perceptual color contrast for edge detection is color Canny operator. Canny edge operator was developed by John F. Canny in 1986. It is a very powerful and also popular edge detector. It consists of the following four steps.

- 1) Noise reduction using a filter based on the first derivative of a Gaussian.
- 2) Intensity gradients computation to calculate edge strength and edge normal.
- 3) Non-maximum suppression to thin the edges.
- 4) Edge tracing with hysteresis to connect broken edges.

As mentioned previously, our goal is to use perceptual color contrast for color gradients computation so that with minimal modification, grayscale-based algorithms can be used for color image processing. Of the four steps shown above, our color Canny operator requires only slight modification of Step 2. All other steps are the same as the original grayscale version. Figure 4 shows the result using our color Canny detector. There is no noticeable performance difference among the four selected color difference formulas. However, the computation times of the CIE94, CIEDE2000, and DIN99D Canny operators are 3, 22, and 5 times of the CIE76 Canny operator respectively.



**Fig. 4.** Edge strength of the “Peppers” image using different Canny operators. (a) Original color pepper image, (b) grayscale image, (c) CIE76 Canny, (d) CIE94 Canny, (e) CIE2000 Canny, and (f) DIN99D Canny.

## 5 Color Harris Corner Detector

Corner or feature detection is an important step for many computer vision algorithms. Harris corner detector is one of the famous detectors for feature detection. It consists of the following steps [14].

- 1) For each pixel  $(x, y)$  in the image  $I$ , calculate the autocorrelation matrix  $M$  of a small window as

$$M = \begin{bmatrix} \sum E_x^2 & \sum E_x E_y \\ \sum E_x E_y & \sum E_y^2 \end{bmatrix} \quad (6)$$

where  $E_x$  and  $E_y$  represent the horizontal and vertical gradients.

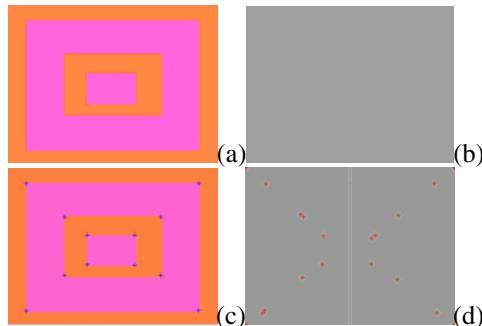
- 2) Construct the cornerness map by calculating the cornerness measure  $C(x, y)$  for each pixel  $(x, y)$  as

$$C(x, y) = \det(M) - k(\text{trace}(M))^2 \quad (7)$$

where  $k$  is a constant coefficient.

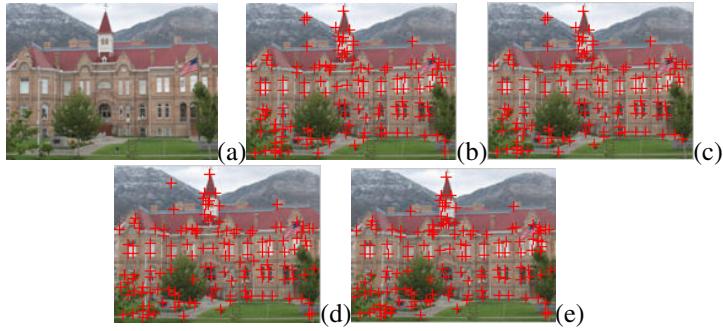
- 3) Perform non-maximal suppression to find local maxima.  
 4) Threshold the cornerness map by setting all  $C(x, y)$  below a threshold value to zero. All non-zero points remaining in the cornerness map are considered corners.

Of the four steps shown above, our color Harris corner detector requires only slight modification in Step 1. All other steps are the same as the original grayscale version. A computer-generated color pattern that has high color contrast (Figure 5(a)) but with fairly uniform grayscale values (Figure 5(b)) is generated to illustrate the advantage of using our color corner detectors. When the threshold is set to be 0.05 of the maximum of  $C(x, y)$ , all four color Harris corner detectors successfully detect all twelve corners (Figure 5(c)). The grayscale Harris corner detector fails to find any corners using the same threshold. This is due to the grayscale cornerness measures are much lower than the color cornerness measures. The threshold must be set to be very close to zero in order for the grayscale corner detector to find all twelve corners but also with a few false corners (Figure 5(d)).



**Fig. 5.** (a) Original color pattern, (b) grayscale version of (a), (c) corners detected using color Harris corner detectors, and (d) corners detected using grayscale Harris corner detector

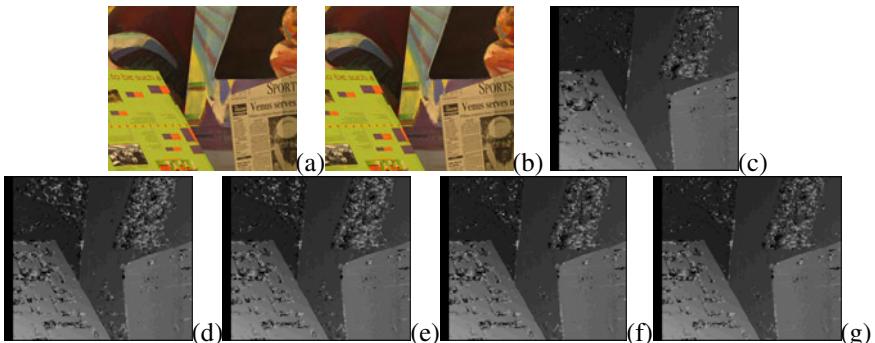
Figure 6 shows the results of a real scene using color Harris corner detectors. The overall performance of all four formulas is very close. CIE2000 and DIN99D have detected a few more corners than the other two formulas. However, the computation times of the CIE94, CIE2000, and DIN99D corner detectors are 2.13, 13.0, 3.4 times of the CIE76 Corner operator.



**Fig. 6.** Color detection results using Harris corner detectors. (a) original image, (c) CIE76 Corner, (d) CIE94 Corner, (e) CIE2000 Corner, and (f) DIN99D Corner.

## 6 Color SAD Block Matching

The last example of this study is color SAD block matching for solving the correspondence problem for stereo vision applications. A large number of algorithms for solving stereo vision correspondence problem have been developed. Most of them use grayscale values. Recent work in [15,16] claimed that color correspondence methods perform much better than grayscale methods. However, these color methods are much more complex than the grayscale algorithms. In this section, we use our perceptual color contrast concept to develop a color SAD algorithm to demonstrate that it is suitable for solving the correspondence problem. Other more powerful stereo matching algorithms can be easily converted to color.



**Fig. 7.** Venus (a) left image, (b) right image, (c) grayscale SAD, (d) CIE76 SAD, (e) CIE94 SAD, (f) CIE2000 SAD, and (g) DIN99D SAD

The SAD algorithm is an area-based correspondence matching method. It computes the summation of the absolute intensity differences of pixels in a small window centered at  $(x, y)$ . The only difference between the grayscale SAD and color SAD is that the grayscale difference is replaced by the color difference measured by one of those selected color difference formulas.

Four stereo image pairs, Tsukuba, Venus, Teddy, and Cones [17,18] were used to test the proposed color SAD methods. We implemented these color SAD methods using window size 5. Only the results for Venus were shown in this paper in Figure 7 owing to page limit. We scaled the resulting disparity maps by integer factors mentioned in [19]. Also, we submitted our results to the web [19] for evaluation in order to compare the performance of different color difference formulas. The evaluation results show that CIE2000 SAD and DIN99D SAD performed slightly better than CIE94 SAD and CIE76 SAD. However, the computation times of the CIE94 SAD, CIEDE2000 SAD, and DIN99D SAD are 2.9399, 12.7567, 1.6476 times of the CIE76 SAD respectively.

## 7 Conclusions

We have presented a new approach for directly converting grayscale image processing techniques that require the computation of grayscale difference into their color version using perceptual color contrast. We have also achieved color performance without significantly altering the grayscale-based algorithms. Our experimental results demonstrate the feasibility of this approach.

Based on our findings in this work, we feel very confident that perceptual color contrast can be easily adapted for many powerful grayscale image processing techniques that require the computation of grayscale difference such as color optical flow, color target recognition, color feature detection and feature tracking.

## References

1. Mojsilovic, A., Soljanin, E.: Color Quantization and Processing by Fibonacci Lattices. *IEEE Transactions on Image Processing* 10, 1712–1725 (2001)
2. Chen, H., Chien, W., Wang, S.: Contrast-Based Color Image Segmentation. *IEEE Signal Processing Letters* 11, 641–644 (2004)
3. Liu, K.C.: Color-edge Detection Based on Discrimination of Noticeable Color Contrasts. *International Journal of Imaging Systems and Technology* 19, 332–339 (2009)
4. Moreno1, R., Garcia, M.A., Puig, D., Julia, C.: Robust Color Edge Detection through Tensor Voting. In: *IEEE International Conference on Image Processing (ICIP)*, Cairo, Egypt, pp. 2153–2156 (2009)
5. Chou, C., Liu, K.: Performance Analysis of Color Image Watermarking Schemes Using Perceptually Redundant Signal Spaces. In: *International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Pasadena, CA, USA, pp. 651–654 (2006)
6. Fondón, I., Serrano, C., Acha, B.: Segmentation of Skin Cancer Images based on Multistep Region Growing (2009), <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.145.1619>
7. Li, X., Zhang, X.: A Perceptual Color Edge Detection Algorithm. In: *International Conference on Computer Science and Software Engineering*, Wuhan, China, pp. 297–300 (2008)
8. Acharya, T., Ray, A.K.: *Image Processing Principles and Applications*. A John Wiley & Sons, Inc., Chichester (2005)

9. Luo, M.R., Cui, G., Rigg, B.: The Development of the CIE 2000 Color Difference Formula: CIEDE 2000. *Color Research and Application* 26, 340–350 (2001)
10. Cui, G., Luo, M.R., Rigg, B., Roesler, G., Witt, K.: Uniform Color Spaces Based on the DIN99 Color-Difference Formula. *Color Research and Application* 27, 282–290 (2002)
11. DIN Deutsche Institut für Normung e.V., DIN 6176: Farbmehratische Bestimmung von Farbabständen bei Körperfärbungen nach der DIN99-Formel, Berlin (2000)
12. Luo, M.R., Cui, G., Li, C.: Uniform Color Spaces Based on CIECAM02 Color Appearance Model. *Color Research and Application* 31, 320–330 (2006)
13. Huertas, R., Melgosa, M.: Performance of a Color-difference Formula Based on OSA-UCS Space Using Small–medium Color Differences. *Journal of the Optical Society of America A* 23, 2077–2084 (2006)
14. Harris, C., Stephens, M.: A combined corner and edge detector. In: *Proceedings of the 4th Alvey Vision Conference*, pp. 147–151 (1988)
15. Compa, P., Satorre, R., Rizo, R., Molina, R.: Improving Depth Estimation Using Color Information in Stereo Vision. In: *IASTED International Conference on Visualization, Imaging, and Image Processing*, Benidorm, Spain, pp. 377–389 (2005)
16. Cabani, I., Toulminet, G., Bensrhair, A.: Self-adaptive Color Edges Segmentation and Matching for Road Obstacle Detection. In: *IEEE Intelligent Vehicles Symposium*, Tokyo, Japan, pp. 58–63 (2006)
17. Scharstein, D., Szeliski, R.: A Taxonomy and Evaluation of Dense Two-frame Stereo Correspondence Algorithms. *International Journal of Computer Vision* 47, 7–42 (2002)
18. Scharstein, D., Szeliski, R.: High-accuracy Stereo Depth Maps Using Structured Light. In: *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Madison, vol. 1, pp. 195–202 (2003)
19. <http://vision.middlebury.edu/stereo/submit/>

# GPU Acceleration of Robust Point Matching

Chad Mourning, Scott Nykl, Huihui Xu,  
David Chelberg, and Jundong Liu

School of Electrical Engineering and Computer Science  
Ohio University, Stocker Center  
Athens, Ohio, USA, 45701

**Abstract.** Robust Point Matching (RPM) is a common image registration algorithm, yet its large computational complexity prohibits registering large point sets in a timely manner. With recent advances in General Purpose Graphical Processing Units (GPGPUs), commodity hardware is capable of greatly reducing the execution time of RPM when non-rigidly aligning thousands of data points. In this paper, we identify areas where parallelism can be exploited in the RPM algorithm, and investigate a GPU-based approach to accelerate the implementation. Other common RPM implementations are compared with our solution. Experiments on synthetic and real data sets show that our approach achieves close to linear speed-up with respect to total computational power over the widely used Matlab implementation. Our tests indicate that utilizing our implementation on current state of the art GPU technology would enable the use of vastly greater point set sizes.

## 1 Introduction

Image registration is the process of establishing correspondence between two or more images. It is one of the most important problems in a variety of fields including medical imaging, computer vision, remote sensing, etc.

Image registration algorithms can be classified into two categories, namely, feature-based and intensity-based methods. In the former, features are extracted from input images, and a similarity metric is employed to measure the feature alignment under a class of parametrized or more generally non-parametrized transformations. The latter determines the optimal coordinate transformations directly from the image data or a derived image-like representation of the same.

Salient points extracted from image pairs often make a simple yet very powerful form of feature. They not only identify significant locations where correspondence should be established, but also provide a foundation to build more sophisticated representations such as curves and surfaces. Numerous point-set based registration solutions [1–3] have been proposed in the literature. Iterative Closest Point (ICP) matching [4] is a classical solution for registering rigid point sets. ICP starts with an initial guess of the relative rigid-body transform between the two inputs, and iteratively refines it by repeatedly generating correspondences between pairs of points and minimizing a least-square error metric. Recently, several ICP extensions [4, 5] have been proposed to handle non-rigid point matching. The major drawbacks of ICP (as well as its variants) reside in its lack of

robustness. First, ICP optimization is prone to be trapped in local minima and therefore often fails to lead to a proper convergence. Second, ICP requires a good initial guess of the alignment to converge to the desired solution. Third, if the inputs contain a considerable number of non-corresponding points, or outliers, the matching performance of ICP greatly deteriorates.

The Robust Point Matching (RPM) algorithm proposed by Chui *et. al.* [2] handles the robustness issue much better than ICP. RPM utilizes a softassign/deterministic annealing approach [6, 7] to estimate point correspondences and alignments simultaneously. Instead of matching the points based on candidate distances as in ICP, RPM fuzzily assigns a correspondence probability to each pair based on a Gaussian weighted distance metric. Meanwhile, a deformation field parametrized by thin-plate splines [8, 11] is computed based on these weighted probabilities. At each iteration, the width of the Gaussian functions, which corresponds to the temperature parameter in deterministic annealing, is decreased, and a new correspondence with increased certainty is established. This procedure iterates until a certain predetermined temperature is reached. After thresholding, a binary correspondence is obtained. Robustness is achieved by treating the points with no matching partners as outliers and automatically eliminating them from the parameter estimation procedure.

The primary drawback of the RPM-TPS method is its slowness. Dominating the computation cost of the algorithm is the spline transformation update routine that requires an inversion operation of an  $m \times m$  matrix, where  $m$  is the number of points in the reference data set. Implementations (especially the original Matlab implementation) suffer from slow execution, when the number of points becomes reasonably large. Recent developments [9, 10] based on RPM also have the same problem.

To utilize RPM in the areas of neuroimaging, GIS, etc, where massive data sets are usually involved, it is necessary to parallelize and optimize the algorithm to reduce its running time. In this paper, we develop a GPU-based approach and demonstrate the improvements made over the prior implementations. Our solution takes advantage of the parallelism existing in the RPM algorithm, and distribute the major computational load onto different GPUs. Close-to-linear speedup with respect to total computational power over the original Matlab RPM-TPS implementation [2] is achieved.

The remainder of this paper is organized as follows: Section 2 outlines the major steps of the RPM algorithm; Section 3 describes the proposed GPU acceleration scheme; Section 4 presents experimental results on synthetic and real medical data sets; Section 5 discusses future work and conclusions drawn from the work.

## 2 The RPM-TPS Algorithm

This section summarizes Chui's work as presented in [2]. We present it here for the reader's convenience.

Let  $V = \{v_1, \dots, v_K\}$  and  $X = \{x_1, \dots, x_N\}$  (in  $\mathbb{R}^2$  or  $\mathbb{R}^3$ ) be the two input point sets. The registration goal is to find a non-rigid transformation  $f$  that best matches reference set  $V$  and target set  $X$ .

## 2.1 Joint Optimization of the Mapping and Correspondence

The RPM-TPS algorithm estimates the mapping  $f$  and point correspondences simultaneously. A correspondence matrix  $M$  of size  $(K + 1) \times (N + 1)$  is constructed to record the probability of each point pair  $(v_a \in V, v_i \in X)$  being matched. If point  $v_a$  is eventually paired with  $v_i$ ,  $m_{ai}$  will be equal to 1. Otherwise it will be 0. The extra row at  $(K + 1)$  and column at  $(N + 1)$  of  $M$  take care of outliers. To solve the joint estimation problem, a *fuzzy assignment-least square* energy function is minimized as follows,

$$E(M, f) = \sum_{i=1}^N \sum_{\alpha=1}^K m_{ai} \|x_i - f(v_a)\|^2 + \lambda \|Lf\|^2 + T \sum_{i=1}^N \sum_{\alpha=1}^K m_{ai} \log m_{ai} - \zeta \sum_{i=1}^N \sum_{\alpha=1}^K m_{ai} \quad (1)$$

where  $m_{ai}$  satisfies  $\sum_{i=1}^{N+1} m_{ai} = 1$  for  $i \in 1, \dots, N$  and  $\sum_{a=1}^{K+1} m_{ai} = 1$  for  $a \in 1, \dots, K$  with  $m_{ai} \in [0, 1]$ .

The Thin Plate Spline (TPS) representation is chosen in RPM-TPS as a specific non-rigid transformation model, with  $\|Lf\|^2$  being the imposed smoothness constraint on the estimate of  $f(v_a)$ .  $\lambda$  and  $\zeta$  are constant weighting factors. Before the system converges to an exact correspondence, elements in the correspondence matrix  $m$  are kept as real numbers in the interval  $[0, 1]$  that indicate the probabilities of each point pair  $(v_a, v_i)$  to be matched. A deterministic annealing scheme is adopted in the iterative procedure to gradually increase the correspondence certainties, and this global-to-local estimation fashion brings robustness to the RPM-TPS algorithm.

As shown in [11], with a fixed weighting parameter  $\lambda$ , eqn. (1) has a unique minimizer  $f$  that can be decomposed into two matrices  $d$  and  $w$ , representing the global affine and local non-affine transformation respectively,

$$f(v_a, d, w) = v_a \cdot d + \phi(v_a) \cdot w \quad (2)$$

The vector  $\phi(v_a)$  is the TPS kernel defined by  $\phi_b(v_a) = \|v_b - v_a\|^2 \log \|v_b - v_a\|$ .

## 2.2 Numerical Solution of the RPM-TPS Algorithm

The TPS minimization scheme proposed in [11] is employed in the RPM-TPS algorithm to estimate the  $d$  and  $w$  in eqn. (2). The affine and non-affine warping spaces are separated using a QR decomposition. The minimization of eqn. (1) can then be reformulated as

$$E_{TPS} = \|Q_2^T Y - Q_2^T \phi Q_2\|^2 + \|Q_1^T Y - R d - Q_1^T \phi Q_2 \gamma\|^2 + \lambda \gamma^T Q_2^T \phi Q_2 \gamma \quad (3)$$

where  $w = Q_2 \gamma$  and  $\gamma$  is a  $(K - D - 1) \times (D + 1)$  matrix.

Eqn. (3) can then be minimized in an alternating fashion, first on  $\gamma$  and then on  $d$ . Several mathematical manipulations results in the following equations, which are used to compute  $d$  and  $w$ :

$$\hat{w} = Q_2(Q_2^T \Phi Q_2) + \lambda I_{(K-D-1)})^{-1} Q_2^T Y \quad (4)$$

$$\hat{d} = R^{-1}(Q_1^T Y - \Phi \hat{w}) \quad (5)$$

The estimation of  $(d, w)$  is conducted iteratively in RPM-TPS, together with the updating of the correspondence matrix  $M$ . An annealing scheme controls the dual-update procedure with a gradually reduced temperature parameter  $T$ . During each iteration, the matrix  $M$  is updated as follows,

$$m_{ai} = \frac{1}{T} e^{-\frac{(x_i - f(v_a))^T (x_i - f(v_a))}{2T}} \quad (6)$$

$$m_{K+1,i} = \frac{1}{T_0} e^{-\frac{(x_i - v_{K+1})^T (x_i - v_{K+1})}{2T_0}} \quad (7)$$

$$m_{a,N+1} = \frac{1}{T_0} e^{-\frac{(x_{N+1} - f(v_a))^T (x_{N+1} - f(v_a))}{2T_0}} \quad (8)$$

### 3 GPU Acceleration of RPM-TPS

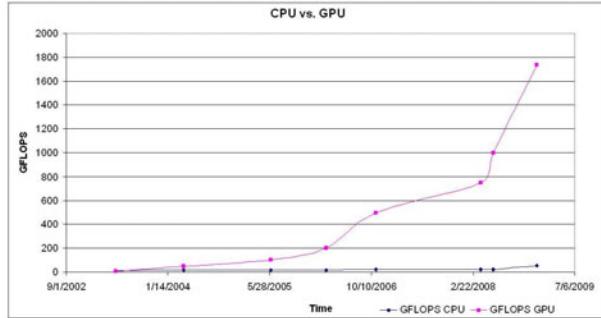
The 21st century has experienced a parallel-processing revolution thanks in large part to the massively multi-pipelined graphics cards used for modern video gaming. The total processing power of the modern GPU has grown significantly faster than the CPU. This fact is easily seen in Fig. 1, using data distributed by Intel and NVidia. NVidia has created a special purpose C-extension, CUDA (Compute Unified Device Architecture)[12, 13], to help easily implement algorithms that exploit the full power of the GPU. CUDA compatible graphics cards are inexpensive and readily available.

Applications most suitable for GPU parallelization are those that operate on large volumes of data that can be processed independently. Operations that are inherently serial or data sets too small to saturate the GPU's stream processors are not well suited for GPU parallelization.

#### 3.1 GPU Implementation of RPM-TPS

As mentioned in section 2 and in Chui et. al. [2], RPM-TPS attempts to match reference set  $V$  of size  $K$  onto target set  $X$  of size  $N$  in an iterative fashion; both points sets are of dimensionality  $D$ , which for our experiments was 3. Each iteration requires three main steps: 1) generate a  $K \times N$  fuzzy correspondence matrix  $M$  as shown in eq.(1), 2) compute the affine transform matrix  $d$  and non-affine warping matrix  $w$  (eqs. (3), (5), (4)) and 3) transform each point  $v$  in set  $V$  by the  $d$  and  $w$  transformation matrices.

Each iteration requires operations on several large matrices. In order to compute eqs. (4) and (5), a QR decomposition must first be performed on the current reference



**Fig. 1.** Intel CPU processing power vs. NVidia GPU processing power

set  $V$ , which, in matrix form, is a  $K \times (D + 1)$  matrix [2]. The QR decomposition generates three matrices  $Q_1$ ,  $Q_2$ , and  $R$  of sizes  $K \times (D + 1)$ ,  $N \times (K - D - 1)$ , and  $(D + 1) \times (D + 1)$ , respectively. As shown in eqs. (3), (4), and (5), matrix  $Q_2$  is involved in 10 different matrix multiplications and is on the order of  $N \times K$ . In total, 3 of these multiplications result in a product that is approximately  $K \times K$ . Assuming the multiplication of two  $7000 \times 7000$  matrices, the product alone, on a 32-bit machine requires 186.92MB or of memory.

Furthermore eq. (4) requires an inversion of a  $(K - D - 1) \times (K - D - 1)$  matrix. In our larger experiments, this means that a  $7000 \times 7000$  matrix needs to be inverted once per iteration.

The large computational costs associated with each iteration of the RPM-TPS algorithm when operating on large data sets justify a parallel approach, especially for large matrix operations. Such operations are ideal candidates for GPU-based implementations. A modern GPU can perform many more floating point operations per second (FLOPS) than a modern CPU as shown in Fig. II. Our RPM-TPS implementation exploits this GPU potential via NVidia's CUDA API.

Our matrix operations, including multiplication, inversion, and QR decomposition are all performed on the GPU. These three operations, with appropriate algorithms, may be efficiently implemented in parallel on modern GPUs.

As mentioned in section 4, our tests were performed on a NVidia<sup>TM</sup> GeForce 9650M with 1GB of memory. This GPU contains 32 stream processor units (SPUs). As the SPU count increases, so does the potential parallelism. Furthermore, the larger the SPU count, the larger the input data can be before saturating the GPU hardware. 32 SPUs is a modest count as NVidia<sup>TM</sup> higher end cards, such as the Tesla S1070 contain 960 SPUs.

## 4 Experimental Results

We conducted tests using our GPU-based implementation, Chui's Matlab implementation [2], and our non-optimized CPU-based implementation. The results compare one iteration's execution time of each implementation against the size of the matched point

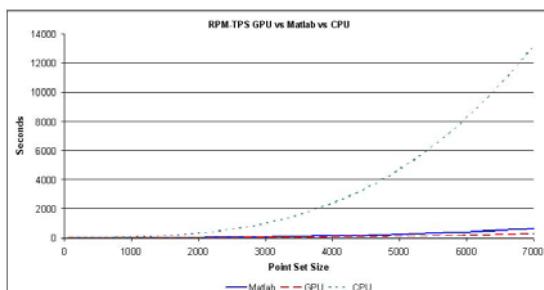
sets. Chui's RPM-TPS implementation defaults to 425 iterations to complete convergence [2]. All tests were conducted on an Intel®Core™2 Duo CPU operating 2.4 GHz with 4 GB of memory with a NVidia™GeForce 9650M with 1 GB of memory.

The experiments were conducted on two different types of data. The first set of data consisted of synthetic data while the second set consisted of actual medical data, specifically, brain surfaces and two sub-cortical structures. The synthetic data consisted of random point sets of desired sizes that were used to collect timing statistics and computational efficiency information. Our GPU accelerated RPM-TPS implementation was then run on the aforementioned brain structures to show the accuracy of the algorithm.

#### 4.1 Synthetic Data

Our synthetic experimental data was collected using a randomly generated point set of size  $N$ ;  $N$  varied from 50 to 7000 in increments of 50. The randomly generated point set  $X$  (of size  $N$ ) served as the target point set;  $X$  was then arbitrarily skewed along the  $X$ ,  $Y$ , and  $Z$  dimensions to create a distorted point set  $V$ . Each test case used RPM-TPS to non-rigidly align reference point set  $V$  on to target point set  $X$ . Because RPM-TPS requires the same computation for each iteration, we ran 4 iterations and averaged them together resulting in the average iteration at size  $N$ . Our GPU implementation successfully completed all test cases yielding 140 data points. Due to time constraints our CPU implementation only ran one iteration at size  $N$  for the first 111 test cases. The later test cases required prohibitively large amounts of time. The Matlab implementation was only able to run test cases up to an  $N$  of 4600 (92 data points); at a larger  $N$ , Matlab threw an exception and terminated execution. For the following figures, each of these data sets were fit against a cubic regression trend line to interpolate the data between the observed values, and in the cases of the CPU and Matlab, to extrapolate up to an  $N$  of 7000.

As shown in Fig. 2 the test case point sets ranged from matching two 3D point sets of size 50 to matching two 3D points of size 7000. The average time, in seconds, to complete one iteration of the RPM-TPS algorithm is plotted against the corresponding point set size.

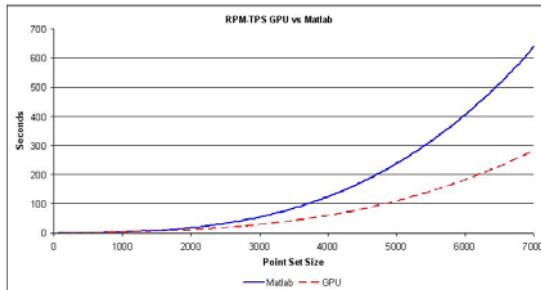


**Fig. 2.** GPU, Matlab, and CPU-based implementations of RPM-TPS. The  $x$  axis defines the size of the two matched point sets. The  $y$  axis defines the average time, in seconds, needed to complete one iteration of the RPM-TPS algorithm.

As expected, our non-optimized CPU-based implementation performed significantly slower than the GPU and Matlab implementations. This is a result of naive matrix operations on large matrices required during each iteration of the RPM-TPS algorithm. One iteration matching two point sets of size 500 required roughly 3.2 sec; thus the total RPM-TPS execution time matching two 500 point sets, running 425 iterations, was approximately 23 minutes. Similarly, the GPU and Matlab required 2.48 minutes and 2.72 minutes, respectively.

## 4.2 Analysis of Synthetic Data

Figure 3 directly compare our GPU implementation against Chui’s Matlab implementation [2]. Once the size of the target and reference point sets reached approximately 450, the GPU execution time fell below that of Matlab. Furthermore, as illustrated by Fig. 3, the GPU’s performance, relative to Matlab, continued to increase linearly as the point set size grew linearly. Using RPM-TPS to align two 7000 3D point sets required 284.546 sec/iteration on the GPU whereas Matlab required 642.280 sec/iteration, a factor of 2.26 times faster. Extrapolating this trend to Chui’s default convergence threshold of 425 iterations, the GPU completed in 33.60 hours whereas Matlab completed in 75.82 hours. The GPU is initially slower than the Matlab implementation because data must be transferred from main memory to the GPU’s memory and a GPU kernel must be dispatched to operate on the data; after the GPU kernel has completed, the result must be retrieved from the GPU back to main memory.

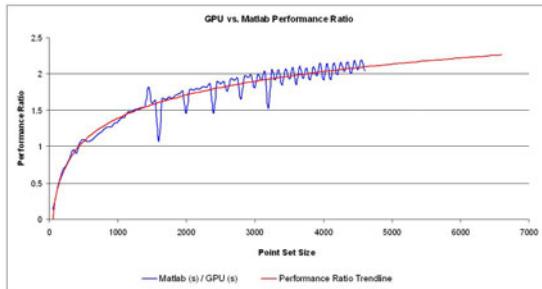


**Fig. 3.** GPU vs. Matlab execution time

As shown in Fig. 1 over the past several years, the GPU’s GFLOPS count has greatly increased beyond that of the CPU; as a result, a GPU-based implementation has a larger computational potential than a corresponding CPU-based implementation. Our NVidia™ GeForce 9650M contains 32 stream processor units (SPUs), each running at 625 MHz and capable of a maximum of three floating point operations per cycle yielding a peak GFLOPS count of 60. Our 2.4 GHz Core 2 Duo capable of a maximum of four SSE floating point operations per cycle yield a peak GFLOP count of 19.2.

The ratio of our GPU GFLOPS count to our CPU GFLOPS count is  $\frac{60}{19.2} = G = 3.125$ . The ratio of our GPU-based performance to Matlab’s performance was  $\frac{642.280 \text{ sec/iteration}}{284.546 \text{ sec/iteration}} = P = 2.26$  at our largest data set size of 7000 points. Therefore,

at 7000 data points, the performance ratio to the GFLOP ratio is  $\frac{P}{G} = E = 0.7232$ . Another way of stating this is the increase in GFLOPS by a factor of  $G$  yields a factor of  $P$  increase in GPU performance which implies our GPU implementation is  $E$  times as efficient as Matlab's implementation. Theoretically, if our GPU implementation was as efficient per GFLOP as Matlab, this ratio would be one, and if our implementation was more efficient than Matlab per GFLOP, this ratio would be larger than one. Matlab utilizes highly efficient matrix operations and makes use of CPU specific optimizations such as SSE that GPUs do not support. Furthermore, numerical implementations for CPUs have existed much longer than current CUDA-based numerical implementations; as a result, Matlab's implementation appears to efficiently utilize the available GFLOPS count of the CPU. Fortunately, modern GPUs have a GFLOPS count several magnitudes greater than modern CPUs as shown in Fig. II this computational power yields significant performance increases. Theoretically, running this same experiment using an 1800 GFLOPS GPU instead of the 60 GFLOPS GPU implies  $\frac{1800}{19.2} = G' = 93.75$ ; assuming the current efficiency factor remains constant,  $G' \times E = P' = 67.80$ . Thus the performance jumps from from a factor of  $2.26X$  to  $67.80X$ .



**Fig. 4.** A plot of the performance ratios ( $P$ ) between Matlab's performance and the performance of our GPU implementation for all point sizes from 50 to 4600. Also contains a trend line extended to 7000 points.

Assuming Matlab to be a perfectly efficient implementation, one would imagine that we should be able to achieve a  $P$  equal to  $G$ , however this is not the case. Our current implementation only accelerates the matrix multiplications, matrix inversions, and QR decomposition. While the remaining portions of the algorithm may not be well suited for a GPU implementation, there are other benefits to such an implementation. The current implementation must send the matrices and retrieve the results each time an aforementioned operation is performed. A purely GPU implementation of this algorithm would only require that the initial point sets be sent to the card, and only the final reference set positions be returned. This technique could provide for further substantial acceleration.

#### 4.3 Medical Imaging Data

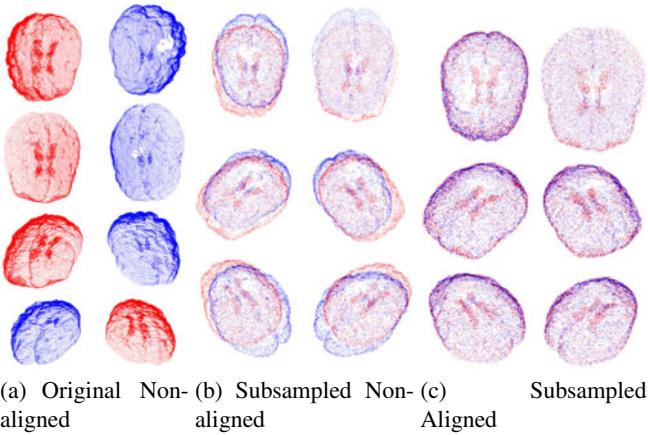
RPM-TPS is commonly used to register brain structures in neuroimaging analysis. A pair of 3D clinical MR images obtained from the University of XXX Hospital, were

used in further tests. The data are 0.94mm x 0.94mm x 3mm in spacing and 256 x 256 x 41 in volume size. Brain surface and sub-cortical structure segmentation were obtained using the FMRIB's Automated Segmentation Tool (FAST) and Integrated Registration and Segmentation Tool (FIRST) in FSL library [14].

Table 1 shows a comparison of the complete registration times for the Caudate and Hippocampus structures. Fig. 5(a) shows four views of the brain surface that are to be aligned with RPM-TPS; each brain surface contains approximately 45,000 data points. Prior to alignment, each brain surface was reduced to 7,000 data points via random subsampling of the original 45,000 point set brain surfaces. The non-aligned brain surfaces are shown in Fig. 5(b), the aligned brain surfaces are shown in Fig. 5(c).

**Table 1.** Registration Times of Brain Structures

Implementation	Hippocampus (1200 Points)	Caudate (617 Points)
GPU	1580.02 sec	293.02 sec
Matlab	2033.12 sec	343.91 sec
CPU	30111.25 sec	4499.90 sec



**Fig. 5.** Non-aligned vs aligned brain surfaces. Shown from left-to-right, top-to-bottom is the top view, bottom view, front left view, front right view, back left view, and back right view, respectively

## 5 Discussion and Future Work

This section discusses topics beyond the present scope of this paper. We are continuing to investigate these important issues.

One alternative acceleration strategy is to subsample the points in a large data set and use only those points for matching. This will effectively reduce the computation

per iteration of the algorithm and decrease the overall time the registration will take to complete. However, it is also possible that the subsampled registration may be less accurate than the registration computed using the full data set.

To determine loss of accuracy, a synthetic data set could be created using a known point set and a known non-affine deformation field  $w$  described in eqn. (4). The data set is subsampled to the appropriate size so that the run time of its registration is the same as the run time of the full data set registration using the acceleration method presented in this paper.

## References

1. Besl, P.J., McKay, N.D.: A method for registration of 3-d shapes. *IEEE Trans. Pattern Anal. Mach. Intell.* 14, 239–256 (1992)
2. Chui, H., Rangarajan, A.: A new point matching algorithm for non-rigid registration. *Computer Vision and Image Understanding* 89, 114–141 (2003)
3. Rangarajan, A., Mjolsness, E., Pappu, S., Davachi, L., Goldman-Rakic, P.S., Duncan, J.S.: A robust point matching algorithm for autoradiograph alignment. In: Höhne, K.H., Kikinis, R. (eds.) VBC 1996. LNCS, vol. 1131, pp. 277–286. Springer, Heidelberg (1996)
4. Fitzgibbon, A.W.: Robust registration of 2d and 3d point sets. In: British Machine Vision Conference, pp. 411–420 (2001)
5. Penney, G.P., Edwards, P.J., King, A.P., Blackall, J.M., Batchelor, P.G., Hawkes, D.J.: A stochastic iterative closest point algorithm (stochasticicp), p. 762 (2001)
6. Gold, S., Rangarajan, A.: A graduated assignment algorithm for graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 18, 377–388 (1996)
7. Gold, S., Rangarajan, A., ping Lu, C., Mjolsness, E.: New algorithms for 2d and 3d point matching: Pose estimation and correspondence. *Pattern Recognition* 31, 957–964 (1997)
8. Bookstein, F.L.: Principal warps: Thin-plate splines and the decomposition of deformations. *IEEE Trans. Pattern Anal. Mach. Intell.* 11, 567–585 (1989)
9. Chui, H., Rangarajan, A., Zhang, J., Leonard, C.M.: Unsupervised learning of an atlas from unlabeled point-sets. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 26, 160–172 (2004)
10. Kirchner, M., Saussen, B., Steen, H., Steen, J., Hamprecht, F.: amsrpm: Robust point matching for retention time alignment of lc/ms data with r. *Journal of Statistical Software* 18 (2007)
11. Wahba, G.: Spline models for observational data. CBMS-NSF Regional Conference Series in Applied Mathematics, vol. 59. Society for Industrial and Applied Mathematics (SIAM), Philadelphia (1990)
12. Corporation, N.: Nvidia cuda compute unified device architecture programming guide (2007), <http://developer.nvidia.com/cuda>
13. Zeller, C.: Cuda performance. In: ACM SIGGRAPH GPGPU Course Notes (2007)
14. Smith, S., et al.: NueroImage, pp. 208–219 (2004)

# A Wavelet-Based Face Recognition System Using Partial Information

H.F. Neo<sup>1</sup>, C.C. Teo<sup>1</sup>, and Andrew B.J. Teoh<sup>2</sup>

<sup>1</sup> Faculty of Information Science and Tehnology,  
Multimedia University,  
75450 Melaka

<sup>2</sup> Biometrics Engineering Research Center,  
Yonsei University,  
Seoul, South Korea

{hfneo,ccteo}@mmu.edu.my, bjteoh@ieee.org

**Abstract.** This paper aims to integrate part-based feature extractor, namely Non-negative matrix factorization (NMF), Local NMF and Spatially Confined NMF in wavelet frequency domain. Wavelet transform, with its approximate decomposition is used to reduce the noise and produce a representation in the low frequency domain, and hence making the facial images insensitive to facial expression and small occlusion. 75% ratio of full-face images are used for training and testing since they contain sufficient information as reported in a previous study. Our experiments on *Essex-94* Database demonstrate that feature extractors in wavelet frequency domain perform better than without any filters. The optimum result is obtained for SFNMF of  $r^* = 60$  with Symlet orthonormal wavelet filter of order 2 in the second decomposition level. The recognition rate is equivalent to 98%.

**Keywords:** Part-based, Face Recognition, Non-negative Matrix Factorization, Wavelet Transform.

## 1 Introduction

Over the recent years, face recognition technology has progressed and is being increasingly applied in various security applications such as public security and law enforcement. However, imperfect face identification from live video cameras or video footage remains a big challenge because the non-cooperative faces could be blocked by other subjects either accidentally or intentionally. Under such environments, partial face recognition may be the only viable alternative for individual identification. Unlike full-face, partial face usually merge with other subjects and this makes it very hard to extract all of the necessary parameters and information for recognition purposes. Nevertheless, the researchers have continued to examine the reliability and workability of this technology. [1] investigated the effect of partial occlusion of facial expression recognition using Gabor wavelets texture information extraction, a supervised image decomposition method based on Discriminant Non-negative Matrix Factorization and a shape-based method that exploits the geometrical displacement of facial features. [2]

presented FAce Recognition against Occlusions and Expression Variations (FARO) based on partitioned iterated function systems. [3] addressed the golden section rule and the maximum margin criterion while [4] made use of Lophoscopic PCA to handle large expression, partial occlusions and other distortions facial images. [5] investigated the main reason for the significant degradation of partial face recognition. On top of that, [6] claimed that 75% of face information are good enough to produce demonstrably recognition accuracy. The faces could achieve similar performance as full face information.

Linear subspace projection ie. Principal Component Analysis (PCA), Linear Discriminant Analaysis (LDA), Non-negative Matrix Factorization (NMF) map the high dimensionality images into a lower-dimensional manifold and treat the images as a whole. This way, some researchers argue that the representation of face is lack the intuition of combining parts to form a whole [7]. Wachsmuth et al. [8] have drawn psycholigical and physiological evidence for part-based object representations in the brain. Thus, part-based representation is very useful and meaningful as it is intuitive with the notion of combining parts to form a whole. [9] proposed Non-Negative Matrix Factorization (NMF) for learning of face features. Some researchers made used of NMF for noise-robust feature extraction in speech recognition [10], eye detection [11] and others have created variant of NMF for instance Weighted Fisher NMF [12]. We made used of NMF and its variants ie. Local NMF (LNMF) [13] and Spatially Confined NMF (SFNMF) [14] to reduce the dimensionality of the raw image and at the same time to preserve as many salient features as possible.

On the other hand, it is worth to mention that the strength of partial face recognition is that the global processing space is less and thus it reduces the computational load if compare with full face recognition. However, the global processing space of 75% face information is still considered big  $O(d^2)$  where  $d$  is the number of pixels in the training images. However, if  $N$ , the number of training images is smaller than  $d$ , the computational complexity would be reduced to  $O(N^2)$ . Besides, PCA, LDA and NMF suffer from poor discriminatory power although they give good representation of the face images. Given two images of the same person and different person, the similarity measure is still high. This means these methods get poor discriminatory power [15]. Hence, we propose to apply our feature extraction methods in wavelet subband to overcome the limitations mentioned. By decomposing an image, the resolution of the subimages are reduced and in turn, the computational complexity are reduced significantly.

The outline of the paper is organized as follow: Section 2 describes wavelet transform based features and its properties. In section 3 we define Non-negative Matrix Factorization (NMF) and its variants – Local NMF and Spatially Confined NMF. The experimental results and discussion are presented in Section 3 and finally conclusion is discussed in Section 4.

## 2 Wavelet Transform

Wavelet transform (WT) is an excellent scale analysis tool which can be applied to any signal with finite energy. It transforms image into multiresolution representation,

which enables one to efficiently compute a small-sized feature representation that is particularly desirable for face recognition [16]. We have proposed to adopt WT due to:

- High temporal localization for high frequencies while offering good frequency resolution for low frequencies. [17] shown that the low-frequency approximation subband is suitable for face descriptor for recognition.
- Wavelet coefficients as the feature representation of face [16]. If an orthonormal wavelet basis was chosen, the coefficients computed are independent to each other creating a set of distinct features of the original signal.
- Shorter computational time. WT decomposes image into a lower dimension multiresolution representation, corresponding to different frequency ranges which minimizes computational overhead.

The wavelet decomposition of a signal  $f(x)$  can be obtained by convolution of signal with a family of real orthonormal basis,  $\psi_{a,b}(x)$  [18]:

$$(W_\psi f)(a,b) = a^{-\frac{1}{2}} \int_{\mathfrak{R}} f(x) \psi\left(\frac{x-b}{a}\right) dx, f(x) \in L^2(\mathfrak{R}) \quad (1)$$

where  $a, b \in \mathfrak{R}$  and  $a \neq 0$  are the dilation parameter and the translation parameter respectively.

Figure 1 shows the image decomposition level. The *LL* band is a coarser approximation to the original image. For one level decomposition, *LH* and *HL* bands record the changes of the image along horizontal and vertical directions while *HH* band shows the higher frequency component of the image. We can further decompose *LL* band to the second level.

<i>LL</i>	<i>LH</i>	1	2	
3	4	6		
5		7		

(a) Level 1 decomposition (b) Level 2 decomposition

**Fig. 1.** Graphical Representation of Wavelet decomposition in (a) level one and (b) two

### 3 Non-negative Matrix Factorization and Its Variants

#### 3.1 Non-negative Matrix Factorization (NMF)

NMF finds an approximate factorization, where  $X$  is the raw face data into non-negative factors  $W$  and  $H$ . The non-negativity constraints make the representation purely additive (allowing no subtractions), in contrast to many other linear representations such as PCA. This ensures that the components are combined to form a whole in the non-subtractive way [9].

Given an initial database expressed by a  $n \times m$  matrix  $X$ , where each column is an  $n$ -dimensional non-negative vector of the original database ( $m$  vectors), it is possible to find two new matrices ( $W$  and  $H$ ) in order to approximate the original matrix:

$$X \approx \tilde{X} \equiv WH, \text{ where } W \in \Re^{n \times r}, H \in \Re^{r \times m} \quad (2)$$

We can rewrite the factorization in terms of the columns of  $X$  and  $H$  as:

$$x_j \approx \tilde{x}_j = Wh_j, \text{ where } x_j \in \Re^n, h_j \in \Re^r \text{ for } j = 1, \dots, n \quad (3)$$

The dimensions of the factorized matrices  $W$  and  $H$  are  $n \times r$  and  $r \times m$ , respectively. Assuming consistent precision, a reduction of storage is obtained whenever  $r$ , the number of basis vectors, satisfies  $(n + m)r < nm$ . Each column of matrix  $W$  contains basis vectors while each column of  $H$  contains the weights needed to approximate the corresponding column in  $X$  using the basis from  $W$ .

In order to estimate the factorization matrices, an objective function has to be defined. We have used the column of  $X$  and its approximation of  $X=WH$  subject to this objective function:

$$\Theta_{NMF}(W, H) = \sum_{j=1}^n \|x_j - Wh_j\|^2 = \|X - WH\|^2 \quad (4)$$

This objective function can be related to the likelihood of generating the images in  $X$  from the basis  $W$  and encoding  $H$ . An iterative approach to reach a local minimum of this objective function is given by the following rules [19]:

$$W_{ia} \leftarrow W_{ia} \sum_{\mu} \frac{X_{i\mu}}{(WH)_{i\mu}} H_{a\mu} \quad (5)$$

$$W_{ia} \leftarrow \frac{W_{ia}}{\sum_j W_{ja}} \quad (6)$$

$$H_{a\mu} \leftarrow H_{a\mu} \sum_i W_{ia} \frac{X_{i\mu}}{(WH)_{i\mu}} \quad (7)$$

Initialization is performed using positive random initial conditions for matrices  $W$  and  $H$ . Convergence of the process is also ensured.

### 3.2 Local Non-negative Matrix Factorization (LNMF)

LNMF aims to improve the locality of the learned features by imposing additional constraints. It incorporates the following three additional constraints into the original NMF formulation [13].

(i) LNMF attempts to minimize the number of basis components required to represent  $X$ . This implies that a basis component should not be further decomposed into more components.

(ii) LNMF attempts to maximize the total “activity” on each component. The idea is to retain the basis with the most important information.

(iii) LNMF attempts to produce different basis as orthogonal as possible, in order to minimize the redundancy between different basis.

LNMF incorporates the above constraints into the original NMF formulation and defines the following constrained divergence as the objective function [13]:

$$\Theta_{LNMF}(W, H) = \sum_i^m \sum_j^n X_{ij} \log \frac{X_{ij}}{[WH]_{ij}} - X_{ij} + [WH]_{ij} + \alpha C_{ij} - \beta \sum_i D_{ii} \quad (8)$$

where  $\alpha, \beta > 0$  are constants and  $C = W^T W$  and  $D = H H^T$ . The structure of the LNMF update for  $W$  is nearly identical to that in Equation 5, 6; differing only in the coefficient matrix  $H$ . The update for  $H$  now uses an element-by-element square root to satisfy the three additional constraints [13]:

$$H_{a\mu} \leftarrow \sqrt{H_{a\mu} \sum_i W_{ia} \frac{X_{i\mu}}{(WH)_{i\mu}}} \quad (9)$$

### 3.3 Spatially Confined Non-negative Matrix Factorization (SFNMF)

SFNMF method is implemented through a series of simple image processing operations to its corresponding NMF basis image. Firstly, a number of  $r$  original NMF basis are selected. Each basis is processed off-line to detect the spatially confined regions. The maximum values of the basis image are identified by adjusting the threshold of a histogram of pixel values and followed by the morphological dilation operation to find a blob region. As a result, SFNMF basis images where only pixels in the detected regions have grey values copied from the corresponding pixels in the original NMF image are created. The remaining pixels are set to zero [14].

SFNMF basis image only represents spatially confined regions. This is intuitive with the idea of recognition by components where spatially confined regions correspond to the important facial features regions such as eyes, eyebrows, nose and lips.

### 3.4 Subspace Face Recognition

NMF, LNMF, and SFNMF bases are learned from a set of training images similar to other subspace project methods. Let  $v$  denote the projection vector, the columns of  $W$  are NMF, LNMF or SFNMF basis images. During recognition process, given an input face image,  $X_{test}$ , it is projected to  $v = W^T X_{test}$ . Thereafter, classification is performed by ' $X v$ ', which were computed from a set of training images by using the  $L_2$  norm or known as Euclidean distance metric.

## 4 Experimental Results and Discussion

A prototype was developed using MATLAB 7.0 for experimental testing, and installed on a 1.60GHz Intel machine with 1Gb of RAM machine. All testings are done using *Faces-94* Essex University Face Database [20]. This database has 153

subjects with 20 images per person. Face images are of size 180x200 in portrait format and after normalization, it becomes 75% of its full-face. The first 53 subjects with 10 images are used for bases training with a total of 530 images. Another 100 subjects with 20 images are used for testing in the probe set with a total of 2000 images. In our experiments, False Reject Rate (FRR) and False Accept Rate (FAR) tests are performed. On top of that, Total Success Rate (TSR) is obtained as follow:

$$\text{TSR} = \left( 1 - \frac{\text{FA} + \text{FR}}{\text{Total number accesses}} \right) \times 100\% \quad (10)$$

where FA = number of accepted imposter claims and FR = number of rejected genuine claims.

For the FAR test, the first image of each subject in the testing set is matched against the first impression of all other faces and the same matching process was repeated for subsequent images, leading to 99,000 (4950 x 20) imposter attempts. For the FRR test, each image of each subject is matched against all other images of the same subject, leading to 19000 (190 attempts of each subject x 100) genuine attempts. The experiments are conducted by using the NMF and its variants in its optimal  $r$  ( $r^*$ ), integrated with Wavelet Transform.

Commonly speaking, all wavelet transforms with smooth, compactly support orthogonality can be used in our study. It is found that the selection of different wavelets does not seriously affect the performance of this approach. We have chosen haar, Daubechies and Symlet filters in Subband  $L_2$  to be integrated with NMF and its variants as feature extractors. Subband  $L_3$  shows the poorest performance due to down sampling process which gets rid of the face feature structures of the coarser images; thus deteriorates the discriminative power of the wavelet transform [21].

Table 1 shows our previous findings in [6]. The comparison of various ratios of faces, which are 25%, 50% (equivalent to right and left face), and 75% of the full-face and 100% full-face prove that 75% face data are good enough for recognition.

**Table 1.** Recognition for 75% facial images using NMF and its variants

Feature extractor	TSR(%)	Number of $r$
NMF	95.59	40
LNMF	97.02	20
SFNMF	96.72	60

The original NMF achieves TSR of 95.59% with  $r = 40$ , followed by our proposed method, SFNMF which achieves 96.72 with  $r = 60$  and LNMF with 97.02% with  $r = 20$ . We notice that the number of  $r$  range from 20 to 40 are relatively large, this is due to the reason when the ratio of the image is big, larger  $r$  is necessary to gain sufficient information to describe a particular face [6].

Next, we will further test NMF, LNMF and SFNMF with the optimal  $r^*$  in wavelet frequency domain. The results are depicted in Table 2, 3 and 4 respectively.

**Table 2.** NMF with  $r^*=40$  in wavelet subband  $L_2$ 

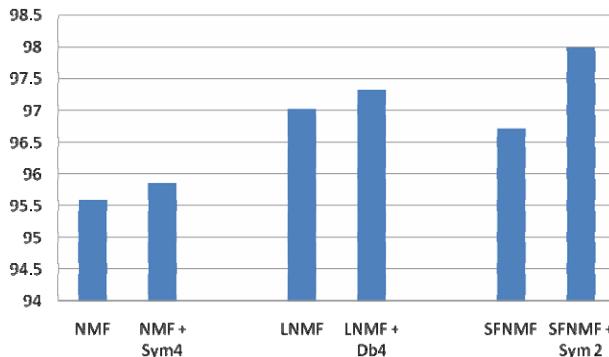
<b>Filter</b>	<b>FAR (%)</b>	<b>FRR (%)</b>	<b>TSR (%)</b>
Haar	4.96	4.98	95.03
Daubechies 2	4.83	4.80	95.17
Daubechies 4	5.00	5.02	94.99
Daubechies 6	5.30	5.29	94.70
Daubechies 8	5.32	5.38	94.67
Daubechies 10	5.54	5.51	94.46
Symlet 2	4.76	4.71	95.25
Symlet 4	4.13	4.18	95.86
Symlet 6	4.68	4.71	95.32
Symlet 8	5.02	5.07	94.97
Symlet 10	5.62	5.69	94.37

**Table 3.** LNMF with  $r^*=20$  in wavelet subband  $L_2$ 

<b>Filter</b>	<b>FAR (%)</b>	<b>FRR (%)</b>	<b>TSR (%)</b>
Haar	3.03	3.11	96.96
Daubechies 2	4.00	4.00	96.00
Daubechies 4	2.66	2.71	97.33
Daubechies 6	3.12	3.11	96.88
Daubechies 8	3.09	3.16	96.90
Daubechies 10	3.44	3.42	96.57
Symlet 2	3.66	3.73	96.33
Symlet 4	3.27	3.38	96.72
Symlet 6	3.40	3.47	96.59
Symlet 8	4.65	4.67	95.34
Symlet 10	3.89	3.91	96.11

**Table 4.** SFNMF with  $r^*=40$  in wavelet subband  $L_2$ 

<b>Filter</b>	<b>FAR (%)</b>	<b>FRR (%)</b>	<b>TSR (%)</b>
Haar	2.88	2.98	97.10
Daubechies 2	2.56	2.62	97.43
Daubechies 4	2.86	2.93	97.13
Daubechies 6	4.17	4.27	95.81
Daubechies 8	3.52	3.60	96.47
Daubechies 10	2.96	3.07	97.03
Symlet 2	2.00	2.09	97.99
Symlet 4	4.21	4.18	95.79
Symlet 6	3.61	3.64	96.39
Symlet 8	2.97	2.84	97.05
Symlet 10	3.10	3.24	96.88



**Fig. 2.** Visual representation of NMF and its variants recognition accuracy in wavelet domains

It is notice that NMF in Symlet 4 filter gives the best TSR of 95.86% with FAR = 4.13% and FTT = 4.18%. On the other hand, LNMF works best in Daubechies 4 filter with TSR of 97.33% with FAR = 2.66% and FRR = 2.71%. Lastly, SFNMF integrated well with Symlet 2 achieving TSR of 97.99% with FAR = 2% and FRR = 2.09%.

Additionally, we illustrated the summary of the finding the Figure 2. The feature extractors in wavelet frequency domain perform much better than sole plain NMF. Besides, our proposed method, SFNMF in wavelet frequency domain has the robust performance compared to pure NMF and Local NFM.

## 5 Conclusion

This paper presents the extension of our previous work in [14], which was on finding the best face ratio performance in face recognition. From the results, it was shown that the face ratios of 75% are able to compensate the information loss in a full-face. In other words, 75% imageries already exists sufficient information, and is suitable for face recognition.

Hence, we made use of 75% face ratio to test the feature extractor in wavelet subband  $L_2$ . Wavelet transform produces lower dimension multiresolution representation that alleviates heavy computational load, and also generates noise and minor distortion insusceptible to face wavelet-based template. We conclude that our proposed method, Spatially Confined NMF in wavelet frequency domain is able to achieve significant recognition rate that is equivalent to 98%.

**Acknowledgments.** The authors wish to thank Ministry Of Science, Technology and Innovation Malaysia. This work is supported by the e-Science grant no. 01-02-01-SF0114.

## References

1. Kotsia, I., Buciu, I., Pitas, I.: An Analysis Of Facial Expression Recognition Under Partial Facial Image Occlusion. *Image and Vision Computing* 26(7), 1052–1067 (2008)
2. Marsico, M.D.: FARO: Face Recognition Against Occlusions And Expression Variations. *IEEE Transactions on Systems, man, and Cybernetics* 40, 121–132 (2010)
3. Tan, X., Chen, S., Zhou, Z., Liu, J.: Face Recognition Under Occlusions And Variant Expressions With Partial Similarity. *IEEE Transations on Information Forensics and Security* 4(4), 217–230 (2009)
4. Tarres, F., Rama, A.: A Novel Method For Face Recognition Under Partial Occlusion Or Facial Expression Variations. In: Proceedings of the 7th WSEAS International Conference on Signal Processing, pp. 37–42 (2008)
5. Ekenel, H.K., Stiefelhagen, R.: Why Is Facial Occlusion A Challenging Problem? In: Tistarelli, M., Nixon, M.S. (eds.) *ICB 2009. LNCS*, vol. 5558, pp. 299–308. Springer, Heidelberg (2009)
6. Neo, H.F., Teo, C.C., Teoh, A.B.J.: A Study On Optimal Face Ratio For Recognition Using Part-Based Feature Extractor. In: *3<sup>rd</sup> IEEE Conference on Signals-Image Technologies and Internet-based System*, pp. 735–741 (2007)
7. Biederman, I.: Recognition-By-Components: A Theory Of Human Image Understanding. *Psychological Review* 94(2), 115–147 (1987)
8. Wachsmuth, E., Oram, M.W., Perrett, D.I.: Recognition Of Objects And Their Component Parts: Responses Of Single Units In The Temporal Cortex Of The Macaque. *Cereb. Cortex* 22(4), 509–522 (1994)
9. Lee, D.D., Seung, H.S.: Learning The Parts Of Objects By Non-Negative Matrix Factorization. *Nature* 401, 788–791 (1999)
10. Schuller, B., Weninger, F., Wollmer, M., Sun, Y., Rigoll, G.: Non-Negative Matrix Factorization As Noise-Robust Feature Extractor For Speech Recognition. In: *Proc. of IEEE ICASSP*, pp. 4562–4565 (2010)
11. Park, C.W., Park, K.T., Moon, Y.S.: Eye Detection Using Eye Filter And Minimisation Of NMF-Based Reconstruction Error In Facial Image. *IEEE Electronics Letters* 46(2), 130–132 (2010)
12. Zhang, Y., Guo, J.: Weighted Fisher Non-negative Matrix Factorization for Face Recognition. In: *Second International Symposium on Knowledge Acquisition and Modeling*, vol. 1, pp. 232–235 (2009)
13. Li, S.Z., Hou, X.W., Zhang, H.J., Cheng, Q.: Learning spatially localized, parts-based representation. In: *IEEE CVPR*, pp. 207–212 (2001)
14. Neo, H.F., Andrew, T.B.J., David, N.C.L.: A Novel Spatially Confined Non-Negative Matrix Factorization for Face Recognition. In: *IAPR Conference on Machine Vision Applications*, Tsukuba Science City, Japan, pp. 16–18 (2005)
15. Mazloom, M., Kasaei, S., Alemi, H.: Construction and Application of SVM Model and Wavelet-PCA for Face Recognition. In: *Second Internal Conference on Computer and Electrical Engineering*, vol. 1, pp. 391–398 (2009)
16. Sellahewa, H., Jassim, S.A.: Image-Quality-Based Adaptive Face Recognition. *IEEE Trans. On Instrumentation and Measurement* 59(4), 805–813 (2010)
17. Sellahewa, H., Jassim, S.A.: Illumination And Expression Invariant Face Recognition: Toward Sample Quality-Based Adaptive Fusion. In: *Proc. 2<sup>nd</sup> IEEE Int. Conf. Biometrics, Theory, Application and System*, pp. 1–6 (2008)

18. Mallat, S.G.: A Theory For Multiresolution Signal Decomposition: The Wavelet Representation. *IEEE Trans. Pattern Anal. Mach. Intell.* 11(7), 674–693 (1989)
19. Lee, D.D., Seung, H.S.: Algorithms For Non-Negative Matrix Factorization. *Proceedings of Neural Information Processing Systems* 13, 556–562 (2001)
20. Vision Group of Essex University- Face Database,  
<http://cswww.essex.ac.uk/allfaces/index.html>
21. Foon, N.H., Pang, Y., Jin, A.T.B., Ling, D.N.C.: An Efficient Method for Human Face Recognition Using Wavelet Transform and Zernike Moments. In: *Proceedings of the International Conference on Computer Graphics, Imaging and Visualization*, pp. 65–69 (2004)

# A Study of Hierarchical Correlation Clustering for Scientific Volume Data

Yi Gu and Chaoli Wang

Michigan Technological University

**Abstract.** Correlation study is at the heart of time-varying multivariate volume data analysis and visualization. In this paper, we study hierarchical clustering of volumetric samples based on the similarity of their correlation relation. Samples are selected from a time-varying multivariate climate data set according to knowledge provided by the domain experts. We present three different hierarchical clustering methods based on quality threshold, k-means, and random walks, to investigate the correlation relation with varying levels of detail. In conjunction with qualitative clustering results integrated with volume rendering, we leverage parallel coordinates to show quantitative correlation information for a complete visualization. We also evaluate the three hierarchical clustering methods in terms of quality and performance.

## 1 Introduction

Finding connection among time-varying multivariate data is critically important in many areas of scientific study. In the field of visualization, researchers have investigated relationships among variables and developed techniques to visualize them. One effective solution is to cluster voxels based on correlation similarity. This allows users to observe how those voxels that have similar correlation behaviors distribute over space and detect possible patterns. When the size of volume data is large, we can select samples for computation to gain an overall impression of the correlation relation in a cost-effective manner. Many research efforts adopted the standard correlation coefficients to study the linear correlation between variables, yet little work is done to build a hierarchy for coarse-to-fine exploration of data correlation. Hierarchical clustering can show cluster within clusters and much as in multiresolution visualization, it provides us a flexible means to adaptively examine the data. In this paper, we present three different hierarchical clustering methods for correlation classification and perform a comparative study of their quality and performance using a climate data set. The evaluation includes side-by-side qualitative comparison of clustering results and quantitative comparison using silhouette plot. We conclude this paper by making our recommendation and pointing out our future research.

## 2 Related Work

Analyzing and visualizing time-varying multivariate data remains a significant challenge in visualization research. Over the years, researchers have applied the

standard pointwise correlation in their analysis [1–4]. New user interfaces were also developed to visualize multivariate data relationships [1, 2]. To the best of our knowledge, hierarchical clustering of time-varying volume data based on the similarity of multivariate correlation has not been investigated, which is the focus of this work.

Parallel coordinates have become a popular technique for visualizing relationships among a large collection of variables. An important issue for parallel coordinates is to order dimensions to reveal multivariate data patterns. One way to achieve this is based on the evaluation of similarity between dimensions. Ankerst et al. [5] provided global and partial similarity measures for two dimensions. They defined the dimension arrangement problem as an optimization problem to minimize the summation of the dissimilarity of all consecutive pairs of axes. Yang et al. [6] built a hierarchical dimension structure and allowed dimension reordering and filtering. We utilize parallel coordinates to show quantitative correlation information for volume samples that are clustered hierarchically. To effectively present relationships among samples, we define two correlation-based distance measures for dimension clustering and ordering.

### 3 Sample Selection

Given a large time-varying multivariate data set, computing the correlation among all voxels over all time steps could be very expensive. A viable alternative is to sample in space and time. This is feasible because in general, the correlation pattern with respect to close neighboring reference locations are similar and not all time steps are necessary in order to detect the correlation pattern. Thus, we can compute the correlation for selected samples at selected time steps and perform clustering to gain an overview of the correlation relationships.

We can adopt uniform or random sampling depending on the need. The domain knowledge about the data can also help us choose a customized sampling scheme. For the climate data set we experiment with, the domain scientists provide the knowledge to assist us in choosing spatial samples and time steps.

## 4 Hierarchical Correlation Clustering

### 4.1 Correlation Matrix

We use the Pearson product-moment correlation coefficient to evaluate the linear correlation between the time series at two sampling locations  $X$  and  $Y$

$$\rho_{XY} = \frac{1}{T} \sum_{t=1}^T \left( \frac{X_t - \mu_X}{\sigma_X} \right) \left( \frac{Y_t - \mu_Y}{\sigma_Y} \right), \quad (1)$$

where  $T$  is the number of time steps.  $\mu_X$  ( $\mu_Y$ ) and  $\sigma_X$  ( $\sigma_Y$ ) are the mean and standard deviation of  $X$  ( $Y$ ), respectively.  $\rho_{XY}$  is in  $[-1, 1]$ . The value of 1 (-1)

means that there is a perfect positive (negative) linear relationship between  $X$  and  $Y$ . The value of 0 shows that there is no linear relationship between  $X$  and  $Y$ . For all the samples given, we can build a correlation matrix  $\mathbf{M}$  with  $\mathbf{M}_{i,j}$  recording  $\rho_{X_i X_j}$ . If  $X_i$  and  $X_j$  are drawn from the same variable (two different variables), then  $\mathbf{M}$  is the self-correlation (cross-correlation) matrix.

## 4.2 Distance Measure

Before clustering the samples, we need to define the distance between two samples  $X$  and  $Y$ . In this paper, we take two different distance measures which both take the correlation matrix  $\mathbf{M}$  as the input. The first distance measure only considers  $\mathbf{M}_{i,j}$  for samples  $X_i$  and  $X_j$ , and we define the distance as

$$d_s(X_i, X_j) = 1 - |\mathbf{M}_{i,j}|. \quad (2)$$

That is, the distance indicates the strength of linear correlation between  $X_i$  and  $X_j$ . When  $X_i$  and  $X_j$  are perfectly correlated (regardless of the sign), then  $d_s(X_i, X_j)$  gets its minimum of 0. If  $X_i$  and  $X_j$  have no linear correlation, then  $d_s(X_i, X_j)$  gets its maximum of 1. The second distance measure considers two rows  $\mathbf{M}_{i,k}$  and  $\mathbf{M}_{j,k}$  for samples  $X_i$  and  $X_j$ , and we define the distance as

$$d_v(X_i, X_j) = \sqrt{\sum_{k=1}^N (\mathbf{M}_{i,k} - \mathbf{M}_{j,k})^2}, \quad (3)$$

where  $N$  is the number of samples. We compute  $d_v(X_i, X_j)$  for all pairs of samples and normalize them to  $[0, 1]$  for our use.

## 4.3 Hierarchical Clustering

The correlation matrix and distance measure defined above can be used to cluster the samples in a hierarchical manner. In general, there are two approaches to build such a hierarchy, *agglomerative* or *divisive* [2]. The agglomerative (or “bottom-up”) approach starts with each sample in its own cluster and merges two or more clusters successively until a single cluster is produced. The divisive (or “top-down”) approach starts with all samples in a single cluster and splits the cluster into two or more clusters until certain stopping criteria are met or each sample is in its own cluster. The advantages of hierarchical clustering are that it can show “cluster within clusters” and it allows the user to observe clusters according to the depth-first-search or breadth-first-search traversal order. We refer interested readers to the work of Zimek [8] for the mathematical background of correlation clustering. In this paper, we experiment with three hierarchical clustering methods based on quality threshold, k-means, and random walks to investigate the correlation relation among samples at different levels of detail.

**Hierarchical Quality Threshold.** This is a bottom-up hierarchical clustering approach which uses a list of distance thresholds  $\{\delta_0, \delta_1, \delta_2, \dots, \delta_l\}$  to create a hierarchy of at most  $l + 1$  levels in  $l$  iterations. These thresholds must satisfy the following conditions:  $\delta_i < \delta_j$  if  $i < j$ ;  $\delta_i \in (0, 1)$  for  $1 < i < l - 1$ ; and  $\delta_0 = 0$ ,  $\delta_l = 1$ . At the beginning, each sample is in its own cluster. At the first iteration, we build a candidate cluster for each sample  $s$  by including all samples that have their distance to  $s$  smaller than threshold  $\delta_1$ . Then, we save the cluster with the largest number of samples as the first true cluster and remove all samples in this cluster from further consideration. In the true cluster, sample  $s$  is treated as its representative sample. We repeat with the reduced set of samples until all samples are classified. At the second iteration, we use threshold  $\delta_2$  to create the next level of hierarchy. The input to this iteration is all representative samples gathered from the previous iteration. We continue this process for the following iterations until we finish the  $l$ th iteration or until we only have one cluster left in the current iteration.

**Hierarchical k-Means.** The popular k-means algorithm classifies  $N$  points into  $k$  clusters,  $k < N$ . Generally speaking, the algorithm attempts to find the natural centers of  $k$  clusters. In our case, the input is the  $N \times N$  correlation matrix  $\mathbf{M}$  where each row in  $\mathbf{M}$  represents a  $N$ -dimensional sample to be classified. The k-means algorithm randomly partitions the input points into  $k$  initial clusters and chooses a point from each cluster as its centroid. Then, we reassign every point to its closest centroid to form new clusters. The centroids are recalculated for the new clusters. The algorithm repeats until some convergence condition is met. We extend this k-means algorithm for the top-down hierarchical clustering where we take each cluster output from the previous iteration as the input to the k-means algorithm and construct the hierarchy accordingly. This process continues until a given number of levels is built or the average distortion within every cluster is less than the given threshold.

**Random Walks.** Random walks [9] are also a bottom-up hierarchical clustering algorithm. In our case, this algorithm considers the  $N \times N$  correlation matrix  $\mathbf{M}$  as a fully connected graph where we treat  $|\mathbf{M}_{ij}|$  as the weight for edge  $e_{ij}$ . At each step, a walker starts from vertex  $v_i$  and chooses one of its adjacent vertices to walk to. The probability that an adjacent vertex  $v_j$  is chosen is defined as  $\mathbf{P}_{ij} = |\mathbf{M}_{ij}|/d_i$ , where  $d_i = \sum_{j=1}^N |\mathbf{M}_{ij}|$ . In this way, we can compute a random walk probability matrix  $\mathbf{P}^t$  to record the possibility starting from  $v_i$  to  $v_j$  in  $t$  steps. With  $\mathbf{P}^t$ , we define the distance between  $v_i$  and  $v_j$  as

$$r_{ij}^t = \sqrt{\sum_{k=1}^N \frac{(\mathbf{P}_{ik}^t - \mathbf{P}_{jk}^t)^2}{d_k}}, \quad (4)$$

Random walks start with every vertex in its own cluster. Then the algorithm iteratively merges two clusters with the minimum mean distance into a new

cluster, and updates all the distances between clusters. This process continues until we only have one single cluster left. The probability of going from a cluster  $C$  to  $v_j$  in  $t$  steps is defined as

$$\mathbf{P}_{Cj}^t = \frac{1}{|C|} \sum_{i \in C} \mathbf{P}_{ij}^t, \quad (5)$$

and the distance between two clusters  $C$  and  $D$  is defined as

$$r_{CD}^t = \sqrt{\sum_{k=1}^N \frac{(\mathbf{P}_{Ck}^t - \mathbf{P}_{Dk}^t)^2}{d_k}}. \quad (6)$$

## 5 Evaluation

To evaluate the effectiveness of different hierarchical clustering algorithms, we generate the same or very similar number of clusters for all methods for a fair comparison. A straightforward comparison is to directly compare the clustering results side by side in the volume space. The limitation of this comparison is that it is subjective, which can be complemented by a quantitative comparison.

Silhouette plot [10] is a technique to verify the quality of a clustering algorithm and it works as follows. For each point  $p_i$  in its cluster  $C$ , we calculate  $p_i$ 's average similarity  $a_i$  with all other points in  $C$ . Then for any cluster  $C_j$  other than  $C$ , we calculate  $p_i$ 's average similarity  $d_{ij}$  with all points in  $C_j$ . Let  $b_i$  be the minimum of all  $d_{ij}$  for  $p_i$ , and the corresponding cluster be  $C_k$  (i.e.,  $C_k$  is the second best cluster for  $p_i$ ), we define the silhouette value for  $p_i$  as

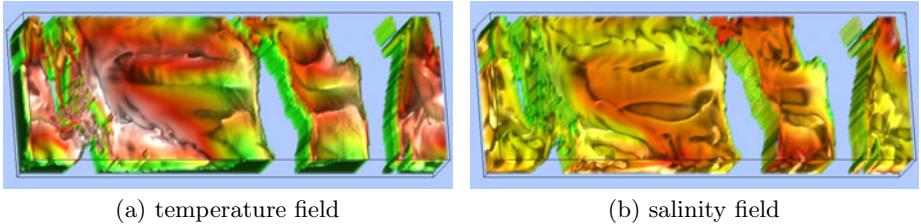
$$s_i = \frac{b_i - a_i}{\max(a_i, b_i)}. \quad (7)$$

$s_i$  is in the range of  $[-1, 1]$ . If  $s_i$  is close to 1 (-1), it means  $p_i$  is well (poorly) clustered. If  $s_i$  is near 0, it means  $p_i$  could be in either cluster. If  $\max(s_i) < 0.25$  for all the points, it indicates that these points are poorly clustered. There are two possible reasons. One reason is that the points themselves could not be well separated or clustered. Another reason is that the clustering algorithm does not perform well. To draw the silhouette plot, we sort  $s_i$  for all the points in each cluster and display a line segment for each point to show its silhouette value. By comparing the silhouette plots for all three clustering algorithms, we can evaluate their effectiveness in a quantitative manner.

## 6 Results and Discussion

### 6.1 Data Set

We conducted our hierarchical correlation clustering study using the tropical oceanic data simulated with the National Oceanic and Atmospheric Administration (NOAA) Geophysical Fluid Dynamics Laboratory (GFDL) CM2.1 global



**Fig. 1.** Snapshots of the temperature and salinity fields at the first time step. Green, yellow, and red are for low, medium, and high scalar values, respectively.

coupled general circulation model. The equatorial upper-ocean climate data set covers a period of 100 years, which is sufficient for our correlation study. The data represent monthly averages and there are 1,200 time steps in total. The spatial dimension of the data set is  $360 \times 66 \times 27$ , with the  $x$  axis for longitude (covering the entire range), the  $y$  axis for latitude (from  $20^{\circ}\text{S}$  to  $20^{\circ}\text{N}$ ), and the  $z$  axis for depth (from 0 to 300 meters). Figure 1 shows the two fields, temperature and salinity, which we used in our experiment. The results we reported are based on the temperature and salinity cross correlation.

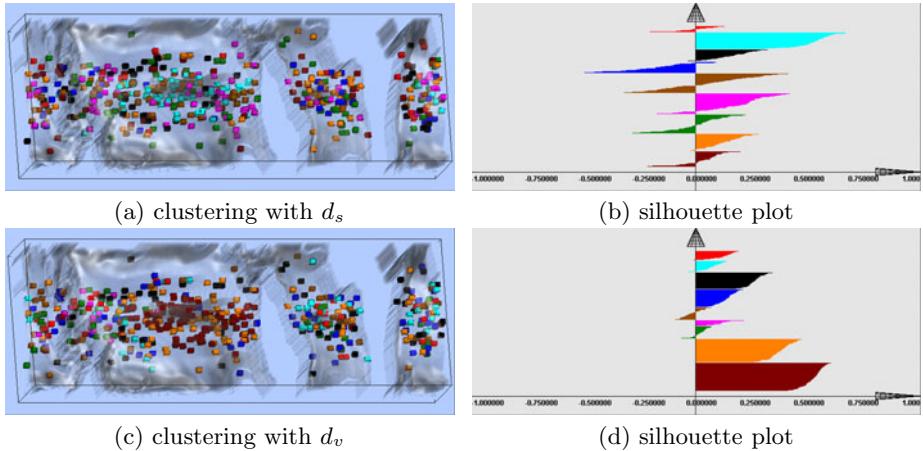
## 6.2 Sampling in Space and Time

For this climate data set, the NOAA scientists provided us with the following knowledge for sample selection. First, voxels belong to the continents are not considered. Second, voxels near the Earth's equator are more important than voxels farther away. As such, the simulation grid along the latitude is actually non-uniform: it is denser near the equator than farther away. Third, voxels near the sea surface are more important than voxels farther away. We incorporated such knowledge into sample selection. Specifically, we used a Gaussian function for the latitude (the  $y$  axis) and an exponential function for the depth (the  $z$  axis) to compute the probability of a voxel being selected. This treatment allows us to sample more voxels from important regions. It also agrees well with the computational grid used in simulation. In our experiment, we sampled two sets of voxels (500 and 2000) from the volume for correlation clustering.

We only took a subset of time steps from the original time series to reduce the computation cost in the correlation study. As suggested by the scientists, we strode in time to reduce the data volumes with fairly independent samples: we took the first time step, then chose every 12th time step (i.e., we picked the volumes corresponding to the same month). A total of 100 time steps were selected to compute the correlation matrix.

## 6.3 Distance Measure Comparison

In Figure 2, we show the comparison of two distance measures  $d_s$  and  $d_v$  on the clustering performance while all other inputs are the same. Although it is

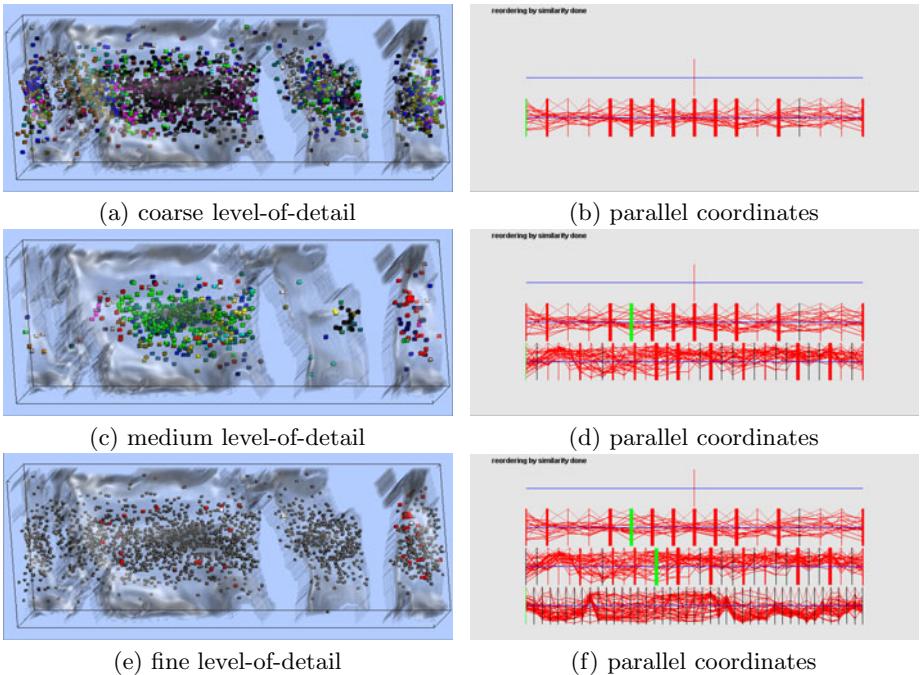


**Fig. 2.** Comparison of two distance measures  $d_s$  and  $d_v$  with random walks. Both have 500 samples and produce nine clusters which are highlighted with different colors in (a) and (c). From (b) and (d), we can see that  $d_v$  performs better than  $d_s$ .

not obvious from the clustering results, the silhouette plots shown in (b) and (d) clearly indicate that  $d_v$  is better than  $d_s$ . In this example, 45.4% of samples have their silhouette value larger than 0.25 when using  $d_v$ , compared with 22.4% of samples using  $d_s$ . For samples with silhouette value less than 0.0, it is 8.8% with  $d_v$  and 26.8% with  $d_s$ . The reason that  $d_v$  performs better is because given two samples,  $d_v$  considers correlations between all samples while  $d_s$  only considers the correlation between the two samples. The same conclusion can be drawn for the other two hierarchical clustering algorithms. We thus used  $d_v$  as the distance measure in all the following test cases.

#### 6.4 Level-of-Detail Correlation Exploration

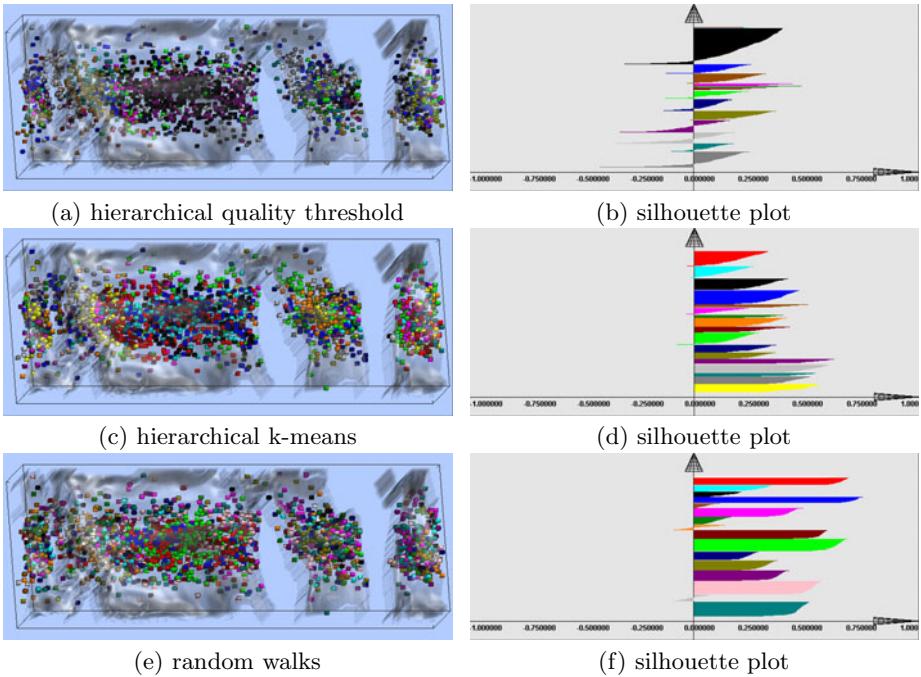
Figure 3 shows the level-of-detail exploration of correlation clusters with the hierarchical quality threshold algorithm. Samples that are not in the current level being explored can be either hidden or de-emphasized as shown in (c) and (e), respectively. Parallel coordinates show the correlation relation quantitatively. In our case, the number of axes in a level equals the number of samples. The thickness of each axis is in proportion to the number of samples it contains in the next level, which provides hint for user interaction. The user can simply click on an axis to see the detail or double click to return. For each level in the parallel coordinates, we sort the axes by their similarity so that sample correlation patterns can be better perceived. The samples along the path from the root to the current level are highlighted in white and green in the volume and parallel coordinates views, respectively. By linking the parallel coordinates view with the volume view, we enable the user to explore the hierarchical clustering results in a controllable and coordinated fashion.



**Fig. 3.** Level-of-detail exploration of correlation clustering of 2000 samples with hierarchical quality threshold. (a) shows all the samples, (c) shows only samples in the current level, and (e) de-emphasizes samples that are not in the current level using gray color and smaller size. Parallel coordinates show the qualitative correlation relationships among samples accordingly. The axes in the current level are reordered by their similarity with each axis corresponding to a (representative) sample.

## 6.5 Clustering Algorithm Comparison

In Figure 4 and Table II, we compare the three hierarchical clustering algorithms. Their silhouette plots clearly indicate that hierarchical quality threshold performs the worst. While hierarchical k-means and random walks have comparable performances in terms of percentages of samples with silhouette value larger than 0.25 and smaller than 0.0. Random walks have more samples with silhouette value larger than 0.5 and it takes much less time to compute compared with hierarchical k-means. Therefore, the random walks algorithm is the best in terms of quality and performance tradeoff. Unlike quality threshold and k-means algorithms, random walks do not require parameters such as the threshold or number of clusters to start with, which also makes it appealing for use. On the other hand, we observed that the timing of quality threshold is very sensitive to the number of levels and the threshold chosen for each level. The output and resulting quality are also very unstable. The quality of k-means is fairly good except that it requires much more time to compute.



**Fig. 4.** Comparison of the three hierarchical clustering algorithms with 2000 samples. The numbers of clusters generated are 17, 18, and 17 for quality threshold, k-means, and random walks respectively. From (b), (d), and (f), we can see that random walks produce the best result while quality threshold produces the worst result.

**Table 1.** Comparison of three hierarchical clustering algorithms. The two timings (percentages) in the speed (quality) entry are for the clustering time in second on an AMD Athlon dual-core 1.05 GHz laptop CPU (samples with silhouette value larger than 0.25) with 2000 and 500 samples, respectively.

	quality threshold	k-means	random walks
strategy	agglomerative	divisive	agglomerative
parameters	# levels threshold for each level	# initial clusters termination threshold	none
randomness	no	yes	yes
tree style	general	general	binary
speed	unstable (184.4s, 22.2s)	slow (673.9s, 30.1s)	fast (188.4s, 4.5s)
quality	bad (22.0%, 67.6%)	good (65.1%, 60.8%)	good (72.3%, 45.4%)
stability	unstable	stable	stable

## 7 Conclusions and Future Work

We have presented a study of hierarchical correlation clustering for time-varying multivariate data sets. Samples are selected from a climate data set based on

domain knowledge. Our approach utilizes parallel coordinates to show the quantitative correlation information and silhouette plots to evaluate the effectiveness of clustering results. We compare three popular hierarchical clustering algorithms in terms of quality and performance and make our recommendation. In the future, we will evaluate our approach and results with the domain scientists. We also plan to investigate the uncertainty or error introduced in our sampling in terms of clustering accuracy.

## Acknowledgements

This work was supported by Michigan Technological University startup fund and the National Science Foundation through grant OCI-0905008. We thank Andrew T. Wittenberg at NOAA for providing the climate data set. We also thank the anonymous reviewers for their helpful comments.

## References

1. Sauber, N., Theisel, H., Seidel, H.P.: Multifield-graphs: An approach to visualizing correlations in multifield scalar data. *IEEE Transactions on Visualization and Computer Graphics* 12, 917–924 (2006)
2. Qu, H., Chan, W.Y., Xu, A., Chung, K.L., Lau, K.H., Guo, P.: Visual analysis of the air pollution problem in Hong Kong. *IEEE Transactions on Visualization and Computer Graphics* 13, 1408–1415 (2007)
3. Glatter, M., Huang, J., Ahern, S., Daniel, J., Lu, A.: Visualizing temporal patterns in large multivariate data using textual pattern matching. *IEEE Transactions on Visualization and Computer Graphics* 14, 1467–1474 (2008)
4. Sukharev, J., Wang, C., Ma, K.-L., Wittenberg, A.T.: Correlation study of time-varying multivariate climate data sets. In: *Proceedings of IEEE VGTC Pacific Visualization Symposium*, pp. 161–168 (2009)
5. Ankerst, M., Berchtold, S., Keim, D.A.: Similarity clustering of dimensions for an enhanced visualization of multidimensional data. In: *Proceedings of IEEE Symposium on Information Visualization*, pp. 52–60 (1998)
6. Yang, J., Peng, W., Ward, M.O., Rundensteiner, E.A.: Interactive hierarchical dimension ordering, spacing and filtering for exploration of high dimensional datasets. In: *Proceedings of IEEE Symposium on Information Visualization*, pp. 105–112 (2003)
7. Izenman, A.J.: *Modern Multivariate Statistical Techniques: Regression, Classification, and Manifold Learning*, 1st edn. Springer Texts in Statistics. Springer, Heidelberg (2008)
8. Zimek, A.: Correlation Clustering. PhD thesis, Ludwig-Maximilians-Universität München (2008)
9. Pons, P., Latapy, M.: Computing communities in large networks using random walks. *Journal of Graph Algorithms and Applications* 10, 191–218 (2006)
10. Rousseeuw, P.J.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. *Journal of Computational and Applied Mathematics* 20, 53–65 (1987)

# Subversion Statistics Sifter

Christoph Müller, Guido Reina, Michael Burch, and Daniel Weiskopf

VISUS, Universität Stuttgart

**Abstract.** We present *Subversion Statistics Sifter*, a visualisation and statistics system for exploring the structure and evolution of data contained in Subversion repositories with respect to both developer activity and source code changes. We support a variety of visualisation techniques, including statistical graphics and graph plots. We exploit the inherent hierarchical structure of software archives to support users of the tool in navigation and orientation tasks and to allow them to obtain insight from the data on different levels of granularity such as directories, files, or even down to single lines of code. The tool is targeted towards large, tiled displays driven by graphics clusters; therefore, distant corresponding views are highlighted by a rubber-banding technique. The system is built on a .NET and WPF basis that exploits data binding and theming of common controls. Following this principle, the system can easily be extended by additional visualisation techniques. We illustrate the usefulness of *Subversion Statistics Sifter* by case studies of student lab and open source software projects.

## 1 Introduction

Version control systems like CVS [1] or Subversion (SVN) [2] are widely adopted nowadays to support the development process of successful software. In fact these configuration management systems are critical tools for managing medium to large-scale software development projects. Considering a large number of developers working for a long time on an extensive code base, the version control data base usually comprises a huge amount of data. While this data can form an invaluable source of information about the structure of the software system as well as its evolutionary process, it is often difficult to understand the vast amount of data in its textual form.

Software visualisation [3] tools tend to extract the raw data from software archives, pre-process it in a subsequent step, provide a variety of views combined by linking and brushing techniques and interactive features to manipulate and navigate the visually encoded data on screen. Many tools exploit the pre-processing step to mirror the data in a closed, locally stored data base, which may impede incremental inspection of software systems still under development.

In this paper, we present *Subversion Statistics Sifter* – a system for visually exploring Subversion repositories and their evolution that uses a time-efficient incremental data grabbing function when the code base is still under development. The already extracted evolutionary data can be visualised while the processing is still running and it can easily be extended by the latest committed versions of the software system.

As part of the software engineering courses, students at Universität Stuttgart have to participate in two large software development projects in groups of six to twelve members, which are intended to teach software engineering by means of a realistically-sized

project. A number of factors make grading these projects difficult, among them the long project duration of twelve months, the rather large number of students and the obfuscation of details of the work distribution from the advisors. Moreover it is known that the effectiveness and efficiency among students strongly vary. The documents stored in or data derived from version control repositories can provide objective hints about the individual development performance if the whole process is considered.

From the beginning, our visualisation system was intended to target single desktop computers – some of them having very high resolution displays attached – and large, tiled displays alike. The rationale is the experience that several advisors usually work together to grade the students. Additionally we found that a large screen real estate for inspecting several different views at the same time eases the grading task. For instance it is often desirable to keep a set of different views simultaneously visible to follow the provenance of findings or to compare several students.

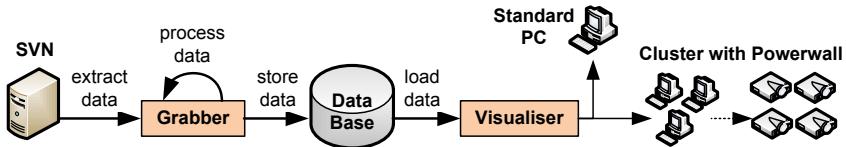
To show the benefits and usefulness of *Subversion Statistics Sifter*, we did not restrict its application to student lab projects, but we also analysed real world open source projects: a case study for the *Notepad++* software system is presented in Section 4

## 2 Related Work

Diehl [4] classifies software visualisation into three main fields: the visualisation of program structure, program behaviour and program evolution. The latter can be explored in many ways: source code is developed over a long period of time and stored by configuration management systems that also include information about developer activity. By applying suitable analysis tools evolutionary data or time series for all kinds of artefacts in this data base can be obtained. In this paper we also address additional factors that have to be observed when implementing a software visualisation tool: data acquisition and output as well as the display hardware itself.

*Subversion Statistics Sifter* belongs to the class of software evolution visualisation tools that focus on developer activity and additional statistical data. It primarily targets software managers who want to analyse the development process of software entities on different levels of granularity and with respect to developer activity. It is deliberately designed to be independent of the programming language and can therefore also be used with textual information that is not strictly structured like source code. In this respect our system is more generic than systems that rely on the in-depth analysis of the program structure, like *GEVOL* [5].

Software visualisation tools mainly extract their data from version archives. Since such archives do not only store information about the source code, but can also comprise the communication of developers via mailing lists, bug reports or trivially in log messages, many of the existing tools do not only extract knowledge from the source, but also include other information. *Seesoft* [6], for instance, is able to visualise large source codes (as thin pixel lines) and offers a treemap-like visualisation for displaying the system's hierarchical structure. However it is not able to show statistics over the system's evolution and changes between subsequent versions. Many tools have been inspired by *Seesoft*, e. g. *Augur* [7], which additionally provides a visual combination of software entities and the activities of developers.



**Fig. 1.** Architectural overview of *Subversion Statistics Sifter*: the Grabber retrieves the data from an SVN repository and stores it in a data base. The Visualiser then uses this data base as input.

Existing tools can also be differentiated by the main techniques they apply to the data sets. The *Evolution Matrix* [8] can be used to detect different phases during software evolution such as growth, stagnation or shrink phases by representation via two-dimensional boxes. The *Evolution Spectrograph* [9] adopts a quite similar approach: files are arranged on the vertical axis and time on the horizontal one. Depending on the distance to last commit time, saturation of the file glyphs fades to white. *Subversion Statistics Sifter* uses a similar colour-based technique for indicating file age. We also adopt the visualisation approach of *CVSScan* [10] and use line-oriented representations to express code changes. Here the abscissa visually encodes time, whereas the columns indicate the version of a file, and a number of metrics as well as the source code can be shown in separate linked displays. *CVSScan* is similar to *HistoryFlow* [11], which also shows code evolution on a horizontal time line separated into versions. *Subversion Statistics Sifter* can additionally show statistical data in coupled views, and the exploration of the data is supported by brushing and linking techniques.

All of the aforementioned systems lack the overview that a large display-ready application can provide. As *Subversion Statistics Sifter* is an application that runs on desktop and Powerwall displays, visualisation and interaction must be designed having large displays in mind without forsaking the advantages that keyboard interaction has for desktop applications. Pointing and clicking on the other hand is suitable for both desktop computers and wall-sized displays. There are approaches that allow for high input resolution, most prominently laser pointers either tracked by cameras in front [12] or behind the display wall [13]; even touch screens can provide a reasonable input precision [14]. Considering Fitt's law, interaction elements for large displays are then oftentimes laid out radially [15], a strategy we adopted in our tool when displaying the hierarchical structure of the system in an overview.

Our system is unique in its combination of visualisation features. Unlike any previous system that we are aware of, it offers the possibility to work cooperatively on large, tiled displays and it supports an incremental analysis of the version archive. Work on large displays is eased by several user interface features, like correspondence rubber bands or redundant and freely-placeable widgets.

### 3 System Description

Figure 1 shows *Subversion Statistics Sifter* and its main components: the Grabber and the Visualiser. The first is responsible for extracting the required data from the Subversion repository and storing it in a relational data base. The latter works solely on the

data base. It offers a visual representation of the repository either on a standard PC or on a cluster of PCs driving a tiled display. The separation of our system into two applications is mainly motivated by having a lightweight Grabber that can run in multiple instances independently from the user interface.

### 3.1 Grabber

The Grabber retrieves information from a Subversion repository revision by revision and stores it to the data base. This process can be performed partially (up to a certain revision) and also incrementally, e. g. for updating the data base to the latest revision, which allows in turn for arbitrary interruption and resumption.

Besides the file system structure, the Grabber retrieves all information that can be obtained from the commit log, including the author names, commit messages, etc. All information is stored as a directed acyclic graph in the data base, which minimises data redundancies. The schema is designed in a way not to rely on assumptions about the semantics of the graph whenever reasonable. These are rather defined by the data contained in the tables, e. g. by the edge annotations classifying an edge as folder structure edge. The Grabber and Visualiser applications can therefore extend the semantics of the graph by adding new types of nodes and edges without changing the data base schema.

### 3.2 Visualiser

The primary display of the Visualiser has been designed around a central tree visualisation (Fig. 2) that shows the directory structure in the background. Users can reach all further detail views from here. These are shown in separate lightweight windows, which can independently be adjusted to a specific revision or range of revisions using their own revision slider.

The user interface of the Visualiser is implemented using Windows Presentation Foundation (WPF). This allows practically any visual element to be interactive: for instance new detail views can be created by tearing off nodes from tree views. Likewise brushing is possible for any element representing an author or a contribution of a single author. Furthermore the extensive data binding and theming capabilities of WPF allow for designing easily extensible visualisation via data models representing e. g. metrics and different visual representations that bind to the model.

For the cluster version of the Visualiser, which drives tiled displays, we opted for using the synchronised execution pattern common for VR applications [16]. Each of the cluster nodes runs a separate instance of the application synchronised via a slim network protocol. This decouples rendering operations from the network and improves the rendering performance through partial view ports, because of decreased geometry load on each node. The latter is an important factor bearing in mind that managing extensive scene graphs is a time and memory-consuming task for WPF.

**Views.** The tree view visualises the directory hierarchy. Its design is closely related to a biological tree and inspired by the idea that a large, tiled display forms a window to the repository landscape. This metaphor is partially extended to a forest if *tags* or *branches* are present, as these folders are shown as additional tree stumps beside the repository root that can be unfolded to replace the central repository tree (cf. bottom of Fig. 2).



**Fig. 2.** Visualisation of *Subversion Statistics Sifter* itself on a Quad HD display ( $3,840 \times 2,160$  pixels). The main view is the tree showing the static structure of the repository at a given revision. Around this tree the user can group detail views as needed, like the sequence of trees on the left side showing that the blue coloured user takes ownership of more and more lines of code. *Subversion Statistics Sifter* specifically supports analysis on high-resolution and physically large displays, like Powerwalls, via its revision grouping feature. Revision grouping directs the user from one view to all other views that show the same revision using a glow effect that points towards these views.

Each branch of the tree represents the path to one of the folders represented by inner nodes and tree leaves. The content of each folder is visualised by small iconic detail views on each node of the tree.

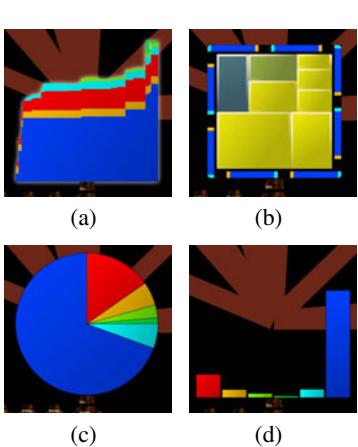
We distinguish three different classes of detail views, which can be opened around the central tree: *Repository views* show the repository folder structure, either as a whole or a specific part of it. *Folder views* display information about a single folder, which can be data like inferred statistics or the immediate content of the folder. Finally *file views* highlight data about a single file. *Subversion Statistics Sifter* implements at least one visual representation for each class and is designed to be easily extensible by further views or data models for these views.

For folders there are four different views that are applicable as separate detail views as well as visualisations for the tree nodes: the contingent progression (Fig. 3(a)) – an important summarising view for our application scenario – shows how the contingents of the authors changed per folder over time. A squarified tree map [17] displaying all immediate file children of a folder (Fig. 3(b)) is the standard folder view for nodes in the tree. The area of the map elements represents the size – in lines of code or bytes – of the files, while their colour from a blue-yellow colour table indicates the revision of the last change. Each file in the tree map provides additional information such as file names and sizes via a tool tip. The last folder views are pie and bar charts that show the contingents at a specific revision (Figs. 3(c) and 3(d)).

Files can be displayed in several different views, provided that they have a text MIME type. First, contingent progression and pie charts based on lines of code can be shown. In addition files support the SVN *blame* history view combining the information of all versions (Fig. 6). It vertically stacks coloured rectangles identifying the respective author while the versions of the file are arranged horizontally. Blocks that have not been changed according to the line matching algorithm of SVN blame are connected with lines between the versions. This allows for identifying stable code as well as probably error-prone parts changed frequently. In the context of grading student projects, this view is useful for identifying contributions in intermediate stages of the development. The revision number on top of each column can be clicked to navigate to the code view, which is the deepest level at which investigation can be conducted.

By selecting different authors lines in the code view can be coloured according to the authors that own them in the selected revision. Only a limited number of authors can be selected simultaneously for highlighting. While this number is configurable, it is reasonable to allow no more than eleven colours for reliable visual separation [18]. The author selection has an influence on all views and its visual characteristics depend on the view in question. For instance a stippled border that indicates each author's contribution by the length of the respective colour strips is put around tree maps (Fig. 3(b)).

**Interacting and Analysing with the Visualiser.** The Visualiser makes use of established techniques, such as direct manipulation, menu selection and iconic representations. Furthermore the characteristics of Powerwall interaction have to be considered, which implies that we mainly rely on pointing and clicking rather than keyboard input. The main part of the navigation is therefore based on multi-level radial context menus (Fig. 4) as well as direct manipulation: all settings can be changed and all views can be opened via the context menu, but we provide additional short cuts like tearing off a



**Fig. 3.** Detail views for folders: contingent progression (a), squarified tree maps (b), contingent pie (c) and bar charts (d)



**Fig. 4.** Small contributions in the final revision require investigation of previous versions to ensure that a student's work is not concealed by others' check-ins. Radial context menus (top left) facilitate the use of our system on large displays. Additionally origin highlighting (see arrow) allows for tracing the source of a detail view over the whole display.

file or folder out of its current view for opening a context menu that allows for creating any possible detail view at the current mouse position. These methods were adopted to reduce pointer travelling distances, which is relevant for direct pointing devices used in front of a wall-sized display as well as for indirect pointers on high-resolution displays. For the same reason the tool bars for switching the revision of the main view and for selecting authors float freely and can be dragged to or opened at any position on the screen, also several instances at the same time.

*Subversion Statistics Sifter* is designed to exploit screen real estate for allowing the user to form spatial clusters of views, e. g. for grouping different stages of analysis to help the understanding of distinct aspects and retracing the performed analysis. Advisors can also employ such grouping to communicate their approach and motivation to their colleagues. However, large distance between views might also cause the user to lose orientation, e. g. if there are many views showing many different revisions. To address such problems our system implements *origin glow* (Fig. 4 see arrow) and *revision grouping* (Fig. 2): when activating a detail view, the origin of its content blinks up shortly. Vice versa, all relevant detail views blink up shortly when selecting a file or folder. The optional revision grouping is a feature specifically designed for large displays. It links views set to the same revision number via a border and a rubber band directing to the location of all other views. The visual metaphor of the rubber band is inspired by [19]. Besides spatial clustering of views, revision groups provide a built-in clustering of views far away from each other.

If there are views the user does not require currently, but wants to preserve for future use, *Subversion Statistics Sifter* provides a view tray, which is a temporary storage for detail views. The user can minimise views into the tray and restore them from a list of thumbnails along with all parameters to their original position later on.

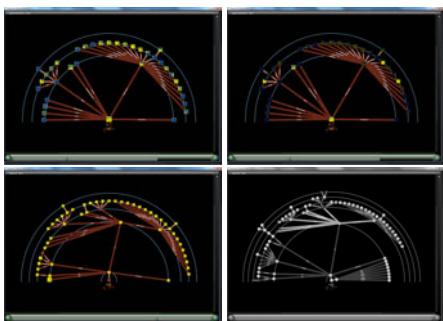
## 4 Case Studies

**Student Projects.** Our general approach for the analysis of student projects is a top-down retrograde workflow. We start from the last revision and verify the code ownership distribution against the final reports the students deliver. Discrepancies at this point would be indicators of severe issues; however it is even more important to relate to the evolution of the ownership over time. So we inspect the progression views and navigate the development process to search for anomalies that prove or disprove that static distribution and supply indications about relevant developer interactions and analyse those. When we have found and gauged the contribution of each student, we can factor the software development aspect into the respective grade.

The first project we analysed was developed by eleven students over 5,408 revisions, totalling 483,134 lines of code (including documentation) in the final revision. The document ownership distribution in the last revision reflected the reports quite closely and varied strongly overall, which however is rather common. In this particular project we knew of one student who made no significant contribution, which is of course directly reflected in the visualisation (see Fig. ④). To verify whether this student could have produced code that was replaced by his colleagues later on we switched to the contingent progression view and were able to spot a few other documents he co-authored over time. The number of files in question was small enough to allow for direct inspection: the blame view proved that on the one hand the contribution was very limited in the first place, and on the other hand most of the meaningful content was replaced over time by other students. From the log messages we learned that some commits had indeed led to non-compiling or otherwise faulty code that had to be fixed later on.

Although the inspection of the evolution of the repository mostly confirmed the impression from the last revision, we nevertheless found contributions that had been removed over time, supporting our assumption that a system must be able to present the whole process for this particular usage scenario. In this and several other projects we repeatedly found the situation that a single person takes ownership of whole files or folders. This can be easily observed in the contingent progression and is usually caused by automatic re-formatting using the IDE, updates in entirely generated files or by changing the line breaks. Even though the task of identifying the contributions of each student is author-centric, it requires different views, sometimes down to the code level, to be reliably solved. Most importantly we found that a global revision slider, which *Subversion Statistics Sifter* has, is not sufficient if there are thousands of revisions. In fact our system would require a static view of the whole temporal evolution to be sure about the findings.

We also analysed *Subversion Statistics Sifter* itself. Here we observed the situation that one author owns a disproportionately large part of the code. Investigating the history of the project, we found that this author is indeed a major contributor, but he did not own the majority of the code from the beginning. This is highlighted by the three repository tree snapshots at the left border of Fig. ②. Drilling down to the source code level, we found that the author in question did a significant re-engineering of the source code single-handedly, therefore concealing the previous contributions of other people. Most of the previously known visualisation systems do not support such a detailed



**Fig. 5.** Different snapshots of the *Notepad++* repository: revision 170, 171, 460 and the final revision 631



**Fig. 6.** Blame view of the file containing the entry point of *Notepad++*. Discontinuities in the connections between lines usually indicate interesting modifications (marked by circles).

inspection as they focus on larger scale and overview. However, judging the code quality is essential for grading of students, wherefore an integrated code view streamlines the workflow.

**Notepad++** is a widely known text editor for Microsoft Windows. Since it is an open source project hosted on *sourceforge.net*, we could readily process the repository and analyse it with *Subversion Statistics Sifter*.

Figure 5 shows various states of the folder hierarchy of *Notepad++*. It is obvious from the contingent pie chart of the latest revision that the development is mostly a one-man project with a second author adding some improvements and two others providing minor contributions. The structure of the project as a whole does not change dramatically over time. In fact already the initial check-in comprised a large software system.

The two upper screen shots of Fig. 5 highlight the first activities of the second, above-mentioned author in the project. From the author highlighting and the folder names in the second image we can infer that he started by adding a new tool bar and integrating it into the main program. The details of the main program file additionally show a huge amount of activity given that one would only expect the program entry point. Inspecting the file in the blame view (Fig. 6), we also gained more insight into the role of the dark green author. He mostly added improvements of existing functionality, like a new command line parsing routine (first check-in) or a more comprehensive error handling (second check-in). Obviously, big changes in more or less stable code, which become quite obvious in our blame view as white parts between the grey lines, are somehow suspicious: for instance the rightmost circled check-in in Fig. 6 adds a fix for a bug in the command line processing happening “for unknown reason” according to code comments. However not all of these visual patterns indicate problems: the third red-circled check-in just added a new command line help and removed some dead code, which causes the thick diagonal line in the blame view.

The tree maps of the *Notepad++* repository show another interesting anomaly: while the first several hundred revisions expose a somehow expected behaviour of some files being untouched for a long time (blue fields in Fig. 5, top left screen shot), the later

revisions have all files being changed recently (Fig. 5 bottom left screen shot). Furthermore one can find the anomaly of the revision slider being discontinued for very central parts like the whole “src” folder (Fig. 5 upper screen shots). Normally this only happens for single files that did not exist in all revisions. Further investigation revealed that in revision 459 the whole repository was erased and re-added in the following check-in. While *Subversion Statistics Sifter* helped to find such an anomaly, the reason for it remains unknown as the log message for revision 460 does not mention it.

## 5 Conclusions and Future Work

We presented *Subversion Statistics Sifter* – a system for visually analysing the software development process-related information from Subversion repositories. Multiple views with intuitive visualisation metaphors allow for easy exploration. Even for extremely large data an overview can effectively be grasped thanks to the support for large display and Powerwall environments powered by single or cluster computers. We addressed usability issues on Powerwall displays by customising standard widgets accordingly and by adding rubber-band correspondence highlighting. Large display support favours operation in collaborative scenarios, which is extremely useful for the grading of the large-scale student lab projects with several advisors. We employed the tool for the grading of the project participants who developed it. While we are aware that such a software system cannot be the only source for grading – as it cannot reveal cooperative work like pair programming – it is nevertheless a helpful building block for it.

Besides using *Subversion Statistics Sifter* for its intended purpose, we also applied it to gain insight for a third-party open source project. Although the system is not optimised for such an application scenario, we nevertheless were able to identify several anomalies. Furthermore we could gain some idea of the organisation of the project with its main author and aide.

The visual metaphors used in the presented system for the data work at a fine-grained level, but have a couple of drawbacks by design. Many of these problems, however, have no obvious solution, e. g. the aggregation of the code-level information into higher levels cannot be achieved easily without cluttering the display. The long-term structural changes of the repository layout were not the immediate subject of the project. In upcoming projects we want to address structural features as well as additional visual representations for the three above-mentioned classes of views. Furthermore we intend to include an extension interface to *Subversion Statistics Sifter* that allows for adding custom analysis steps to the Grabber and new visual representations to the Visualiser without requiring access to the source code of our system.

## Acknowledgements

The authors wish to thank Patrick Auwärter, Christoph Bergmann, Christian Dittrich, Stefan Grohe, Michael Kircher, Nico Rieck, and Sebastian Zitzelsberger for their implementation work on *Subversion Statistics Sifter*.

## References

1. Bar, M., Fogel, K.: Open Source Development with CVS, 3rd edn. Paraglyph Press, Scottsdale (2003)
2. Collins-Sussman, B., Fitzpatrick, B.W., Pilato, C.M.: Version Control with Subversion, 2nd edn. O'Reilly Media, Sebastopol (2008)
3. Stasko, J.T., Domingue, J.B., Brown, M.H., Price, B.A., Foley, J.: Software Visualization. MIT Press, Cambridge (1998)
4. Diehl, S.: Software Visualization: Visualizing the Structure, Behaviour, and Evolution of Software. Springer, Berlin (2007)
5. Collberg, C., Kobourov, S., Nagra, J., Pitts, J., Wampler, K.: A system for graph-based visualization of the evolution of software. In: Proceedings of ACM SoftVis 2003, pp. 77–86 (2003)
6. Eick, S.G., Steffen, J.L., Sumner, J.E.E.: Seesoft — a tool for visualizing line oriented software statistics. *IEEE Transactions on Software Engineering* 18, 957–968 (1992)
7. Froehlich, J., Dourish, P.: Unifying artifacts and activities in a visual tool for distributed software development teams. In: Proceedings of ICSE 2004, pp. 387–396 (2004)
8. Lanza, M.: The Evolution Matrix: Recovering software evolution using software visualization techniques. In: Proceedings of IWPSE 2001, pp. 37–42 (2001)
9. Wu, J., Spitzer, C.W., Hassan, A.E., Holt, R.C.: Evolution Spectrographs: Visualizing punctuated change in software evolution. In: Proceedings of IWPSE 2004, pp. 57–66 (2004)
10. Voinea, L., Telea, A., van Wijk, J.J.: CVSscan: Visualization of code evolution. In: Proceedings of ACM Softvis 2005, pp. 47–56 (2005)
11. Viegas, F.B., Wattenberg, M., Dave, K.: Studying cooperation and conflict between authors with history flow visualizations. In: Proceedings of CHI 2004, pp. 575–582 (2004)
12. Kirstein, C., Müller, H.: Interaction with a projection screen using a camera-tracked laser pointer. In: Proceedings of Multimedia Modeling 1998, pp. 191–192 (1998)
13. Chen, X., Davis, J.: Lumipoint: Multi-user laser-based interaction on large tiled displays. Technical report, Stanford University (2001)
14. Sears, A., Shneiderman, B.: High precision touchscreens: Design strategies and comparisons with a mouse. *International Journal of Man-Machine Studies* 34, 593–613 (1991)
15. Guimbretière, F., Winograd, T.: Flowmenu: combining command, text, and data entry. In: Proceedings of ACM UIST 2000, pp. 213–216 (2000)
16. Schaeffer, B., Goudeseune, C.: Syzygy: Native PC cluster VR. In: Proceedings of IEEE VR 2003, pp. 15–22 (2003)
17. Bruls, M., Huizing, K., van Wijk, J.J.: Squarified treemaps. In: Proceedings of EG/IEEE TCVG Symposium on Visualization, pp. 33–42 (1999)
18. Ware, C.: Visual Thinking for Design. Morgan Kaufman, Burlington (2008)
19. Baudisch, P., Cutrell, E., Robbins, D., Czerwinski, M., Tandler, P., Bederson, B., Zierlinger, A.: Drag-and-pop and drag-and-pick: Techniques for accessing remote screen content on touch- and pen-operated systems. In: Proceedings of Interact 2003, pp. 57–64 (2003)

# A Lossy/Lossless Coding Algorithm Using Histogram

Sunil Bhooshan<sup>1</sup> and Shipra Sharma<sup>2</sup>

<sup>1</sup> Department of ECE, Jaypee University of Information Technology, Waknaghat  
Tel.: +91-1792-239205  
Fax: +91-1792-245362  
[sunil.bhooshan@juit.ac.in](mailto:sunil.bhooshan@juit.ac.in)

<sup>2</sup> Department of CSE and IT, Jaypee University of Information Technology,  
Waknaghat  
Tel.: +91-1792-239338  
Fax: +91-1792-245362  
[shipra.sharma@juit.ac.in](mailto:shipra.sharma@juit.ac.in)

**Abstract.** In this paper we propose a method of compression which is both lossy and lossless. The decision regarding what amount of image will be compressed in lossy manner, and what in lossless, depends on the information obtained by the histogram of the image. Another important decision parameter is bit rate of the image. Together these two parameters enable us to divide the image in two parts for different types of compression. The results show that the difference between original and decompressed images is visually negligible. The PSNR and SSIM are comparable to JPEG. ....

**Keywords:** Histogram, Lossy Compression, Lossless Compression, bit rate.

## 1 Introduction

Compression is one of the oldest techniques to be applied on digital images. One of the main reasons for this is the large consumption of memory and bandwidth by these images. This has resulted in various compression techniques over the years. These techniques vary in the time they take, the type of images they can be applied and the loss of data the application can handle. The three main categories of compression are - lossy, lossless and hybrid of these two.

Huffman compression [1] is one of the most qualitative lossless compression existing. But, due to the way the codes are generated, it becomes very slow as the size of the image increases with varying pixel intensities. Two methods to overcome this disadvantage are as follows. Either dividing the image in small blocks like in [2] and standard methods like JPEG. Another way is to decrease the number of pixels having different intensities. In other words, bringing uniformity in the intensities of pixels. The method proposed in this paper lies in the later category.

This method is computationally less expensive than the one proposed in [2] or JPEG. One of the reasons for this is that Huffman coding is done in repeated manner in the existing techniques, although on smaller blocks, whereas in the proposed method it is done at one go on the whole image. As the maximum number of pixels are of zero intensity it is faster. Also, in [2] different bit rates are not considered, whereas in the proposed algorithm bit rate is the major factor in deciding how much image has to be compressed in lossy manner and lossless compression. In [3], [4] the authors focus on modifying Huffman with respect to how it is done in JPEG. In [5] the authors present a method of generation of Huffman coding tables. Other methods present in literature which are hybrid of lossless and lossy techniques are like [6], [7], [8], [9], [10], [11], [12] and [13]. Our method is simple and different from these existing ones.

We propose a lossy version of Huffman coding in this paper. The image as a whole is divided in two parts depending on its histogram information and bit rate. Lossless compression is performed on the one with top intensities and lossy on the remaining one. This considerably increases the compression ratio provided by Huffman encoding while modifying it for different bit rates.

The outlay of the paper is in four major sections. Section 2 deals with detailed description of the lossy and lossless compression algorithm. Section 3 discusses the results and comparison while Section 4 draws a few conclusion regarding the method.

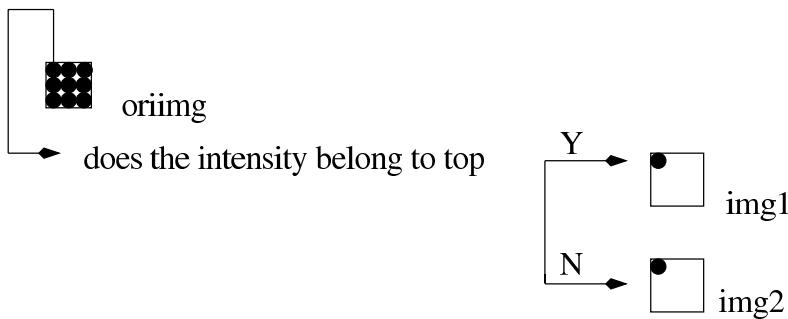
## 2 The Method

Acquire a  $m \times n$ ,  $b$  bit, grayscale image. We will call it  $oriImg$  from now on.

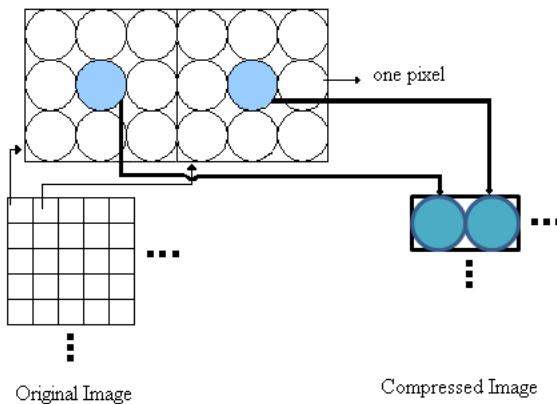
### 2.1 Compression

Perform the below mentioned steps on it:

- Step 1** Calculate histogram of  $oriImg$ . Histogram produces intensities v/s no. of pixels data. This helps us in determining how many pixels belong to a particular intensity.
- Step 2** We find the intensity with maximum number of pixels. Store it as  $I1$ . Similarly find the intensity with second highest number of pixels. Store it as  $I2$ . This is repeated  $b$  times. These calculated intensities  $I1, I2, I3, \dots, Ib$  are grouped together in a set say,  $topInt$ .
- Step 3** The  $oriImg$  is divided in two images. One consisting only of intensities in  $topInt$  set. Other consisting of all intensities except the one in  $topInt$  set. We name the former as  $img1$  and the later as  $img2$ . The same is depicted in Figure 11.
- Step 4** Apply Huffman encoding on  $img1$ . It results in a one dimensional code vector, say  $v1$ . High frequency colors are represented with high accuracy as they contain the maximum information in the image.

**Fig. 1.** Deciding about pixel

**Step 5** Compress  $img2$  in a lossy manner [14] as follows. Starting from the beginning consider a sub-block of size  $3 \times 3$  in  $img2$ . Acquire the intensity value of the center pixel as depicted in Figure 2. Perform the same operation on the next sub-block. The image obtained from all the acquired center pixels is the compressed image. Name it as  $imgCom$ .

**Fig. 2.** Lossy Compression [14]

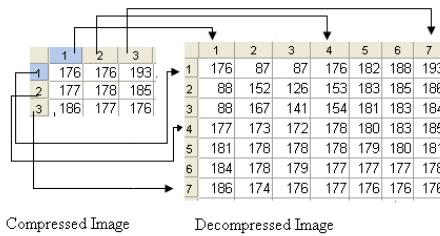
Hence,  $v1$  and  $imgCom$  are the compressed data we have in place of  $oriImg$ . These two data are transmitted to the user. As we can conclude from the above compression that lossy method uses the spatial redundancy present in images whereas lossless compression makes use of frequency information.

## 2.2 Decompression

Acquire  $v1$  and  $imgCom$ . Perform the following steps on them:

- Step 1**  $v1$  is decoded by Huffman decoding algorithm. We obtain the Huffman decoded image, say  $decom1$ .
- Step 2** Apply adaptive interpolation [14] on  $imgCom$  as follows

- Starting with the first pixel in  $imgCom$  acquire three pixels in horizontal direction.
- Derive their polynomial equation.
- Now, interpolate four pixels between them using the polynomial equation derived in the above step. Figure 3
- Repeat this until the whole image is interpolated both horizontally and vertically. Once this is done we obtain decompressed image with some loss of data.
- This image is referred to as  $decom2$ .



**Fig. 3.** Lossy Decompression [14]

- Step 3** We merge  $decom1$  and  $decom2$  transparently.

- $decom1$  is considered as background image and  $decom2$  as foreground.
- Calculate difference between  $decom1$  and  $decom2$  and name it  $diff$ .
- $param$  is a parameter defined, which accounts for the amount of transparency in the images. The values of  $param$  can be varied depending on the image. In other words,  $param$  is used to vary the effect of foreground and background image.
- Apply Equation 1 to obtain the final merged image.

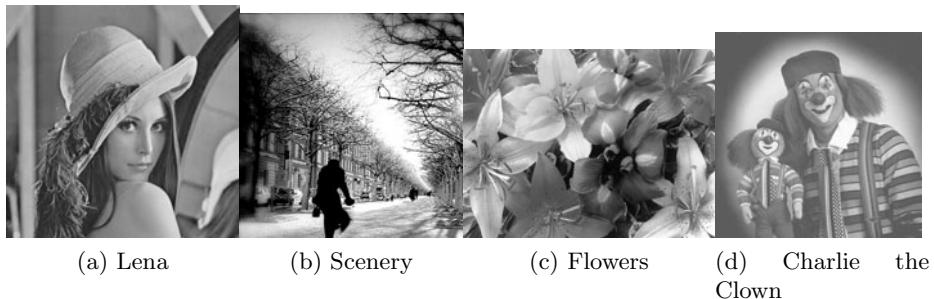
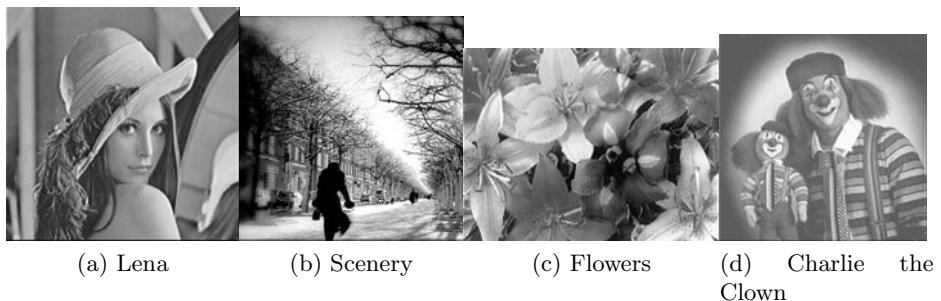
$$mergImg = decom1 + param * diff \quad (1)$$

$mergImg$  is the final decompressed image.

### 3 Experimental Results

#### 3.1 Our Results

We implemented the above algorithm on various set of images. Figure 4 shows few original images.

**Fig. 4.** Original Images**Fig. 5.** Decompressed Images**Table 1.** Compression Ratios

S.No.	Image Name	Size of Original File	Size of Compressed File (Lossy and Lossless)	Compression %
1.	Lena	257KB	29.6KB & 26.2KB	78.3%
2.	Man on Cycle	626KB	70.6KB & 106.6KB	71.6%
3.	Flowers	301KB	34.8KB & 45.5KB	73.3%
4.	Clown	1.1MB	125.3KB & 87.2KB	80.68%

They were compressed using the compression method explained above. After decompressing them as described the reconstructed images are as shown in Figure 5.

As can be seen visually there is negligible difference between the two images. Table shows the amount of compression obtained for the various images depicted in Figure 4.

### 3.2 Comparison Based on Various Parameters

We calculate PSNR and SSIM of the images depicted in Table 2 and compare the result with JPEG. The results show that our method is comparable to JPEG

**Table 2.** Comparative Study

S.No.	Image Name	PSNR(Our Method)	PSNR(JPEG)	SSIM(Our Method)	SSIM(JPEG)
1.	Lena	35.95	35.01	0.8976	0.9001
2.	Man on cycle	36.201	36.149	0.8776	0.8812
3.	Flowers	39.121	38.172	0.9112	0.9110
4.	Clown	40.011	39.557	0.9211	0.9313

in terms of both PSNR and SSIM. That means that our method preserved the structure of the image and also maintains a comfortable PSNR.

## 4 Conclusion

We have presented a simple and fast method to compress image by modifying the existing Huffman technique. The method makes the decision of what to compress in lossless manner and what in lossy based on the highest existing intensities in the image. Such intensities are given more importance than those occurring with less frequency. The number of intensities selected depends on the bit rate of the image. Therefore, images with different bits/pixel can be compressed with the proposed algorithm. This takes care of different bit rates, hence avoiding blocking artifacts, to a large extent, in low bit rates. The compression obtained is almost comparable to JPEG and is considerably more than Huffman encoding. It works with different type of images as has been depicted in results. The quality of decompressed images are comparable to existing standards.

## References

1. Huffman, D.: A method for the construction of minimum-redundancy codes. *IRE* 40, 1011–1098 (1952)
2. Bhooshan, S., Sharma, S.: An efficient and selective image compression scheme using huffman and adaptive interpolation. In: *Image and Vision Computing*, New Zealand (2009)
3. Lakhani, G.: A modification to the huffman coding of jpeg’s baseline compression algorithm. In: *Proceedings on the conference of Data Compression*, p. 557. IEEE Computer Society, Los Alamitos (2000)
4. Lakhani, G.: Optimal huffman coding of dct blocks. *IEEE Transactions On Circuits And Systems For Video Technology* 14, 522–527 (2004)
5. Battiatto, S., Bosco, C., Bruna, A., Di Blasi, G., Gallo, G.: Statistical modeling of huffman tables coding. In: Roli, F., Vitulano, S. (eds.) *ICIP 2005. LNCS*, vol. 3617, pp. 711–718. Springer, Heidelberg (2005)
6. Somchart, C., Masahiro, I., Somchai, J.: A new unified lossless/lossy image compression based on a new integer dct. *IEICE Trans. Inf. Syst.* E88-D, 1598–1606 (2005)
7. Sheng, F., Bilgin, A., Sementilli, P.J., Marcellin, M.W.: Lossy and lossless image compression using reversible integer wavelet transforms. In: *ICIP*, vol. 3, pp. 876–880 (1998)

8. Reichel, J., Menegaz, G., Nadenau, M.J., Kunt, M.: Integer wavelet transform for embedded lossy to lossless image compression. *IEEE Transactions on Image Processing* 10, 383–392 (2001)
9. Wang, L., Wu, J., Jiao, L., Zhang, L., Guangming: Lossy to lossless image compression based on reversible integer dct. In: ICIP (2008)
10. Bassiouni, M.A., Tzannes, A.P., Tzannes, M., Tzannes, N.: Image compression using integrated lossless/lossy methods. In: ICASSP, vol. 4, pp. 2817–2820 (1991)
11. Subhash Chandra, N., Bala Raju, M., Satyanarayana, B., Raja Vikram, B., Mahabub Basha, S., Govardhan, A.: Lossy hybrid binary merge coding for image data compression. *Journal of Engineering and Applied Sciences* 4, 141–144 (2009)
12. Philips, W.: The lossless dct for combined lossy/lossless image coding. In: ICIP, vol. 3, pp. 871–875 (1998)
13. Xin Chen, J.F.F., Kwong, S.: Lossy and lossless compression for color-quantized images. In: ICIP, vol. 1, pp. 870–873 (2001)
14. Bhooshan, S., Sharma, S.: Image compression and decompression using adaptive interpolation. In: The WSEAS International Conference on Signal Processing, Robotics and Automation, University of Cambridge, Cambridge, WSEAS (2009)

# Stereo Matching in Mean Shift Attractor Space

Michal Krumnikl

VŠB - Technical University of Ostrava, FEECS  
Department of Computer Science, 17. listopadu 15/2172,  
70833 Ostrava-Poruba, Czech Republic  
[Michal.Krumnikl@vsb.cz](mailto:Michal.Krumnikl@vsb.cz)

**Abstract.** In this paper, we present a novel method for improving the speed and accuracy of the initial disparity estimation of the stereo matching algorithms. These algorithms are widely investigated, but fast and precise estimation of a disparity map still remains a challenging problem. Recent top ranking stereo matching algorithms usually utilize a window-based approach and mean shift based clustering. We propose an algorithm inspired by a top-down approach exploiting these two steps.

By using the mean shift algorithm, we transform the input images into the attractor space and then perform the matching on the attractor sets. In contrast to the state-of-the-art algorithms, where matching is done on the basis of pixel intensities, grouped according to the results of mean shift algorithm, we perform the matching between the attractor sets of both input images. In this way we are able to acquire fast disparity estimates for whole segments.

## 1 Introduction

Despite the fact that the stereo vision has been widely investigated in the last few decades, fast and precise estimation of the spatial depth still remains a challenging problem. A lot of computational approaches have been examined, but none of them seems to be fully satisfactory.

The basic classification system of the stereo matching algorithms divides the methods into two groups based on the type of output disparity map – the dense stereo matching and sparse stereo matching [1]. The most demanded are the dense disparity maps, in which almost all pixels have disparity values. Sparse disparity maps provide disparity values for only a limited number of pixels. However, these estimation methods are likely to be more reliable.

In recent years, we have noticed a large interest in algorithms producing the dense disparity maps, as these can be used for creating realistic 3D scenes and view synthesis for virtual reality. For more details we refer the reader to a comprehensive discussion regarding the dense two frame stereo matching algorithms by Scharstein and Szeliski [2]. These authors also provide a testbed and online evaluation of the two-frame stereo correspondence algorithms.

In this paper, we propose a new method for improving the initial steps of these algorithms. We will reuse the results of the mean shift algorithm to accelerate

the initial matching by taking advantage of the transformation of input images into the mean shift attractor space. We will show that it is possible to perform the matching on the attractor domain, instead of the original pixel domain. In this way, we are able to acquire faster matching than standardly used SAD-based algorithms and furthermore, decrease the error rate on homogeneous textured areas.

The remainder of the paper is organized as follows. In the Section 2, we present the related works. The Section 3 briefly describes the mean shift algorithm. In the Section 4 we will describe our algorithm and its application. The experimental results of our approach will be provided in the Section 5, followed by a short summary and conclusions.

## 2 Related Works

In this section, we will concentrate only on those stereo matching algorithms, which utilize the mean shift algorithm. The mean shift is either used for image segmentation or disparity space filtering. Because our algorithm is closely related to the successive steps of the stereo matching process, we will examine some of these algorithms in closer detail.

Klaus et al. proposed a stereo matching algorithm that utilizes disparity planes instead of using independent disparity value of each pixel [3]. The stereo matching process described by Klaus goes as follows. In the first step, a decomposition of the image is done using a mean-shift analysis. Local matching algorithm with the fixed size window ( $3 \times 3$  pixels) is applied on all pixels. Absolute intensity differences (SAD) combined with a gradient based measure are used as a dissimilarity measure. In the next step, the disparity planes are estimated from all pixels of each retrieved segment. The plane parameters are estimated by a decomposition method. Having the first estimation of plane parameters, a Belief Propagation (BP) is used to approximate the final optimal solution, which is formulated as an energy minimization problem.

Wang et al. utilized an inter-regional cooperative strategy for the final optimization stage [4]. They preserved the mean-shift algorithm in the first stage and slightly adjusted the stereo matching algorithm using the adaptive correlation window instead of SAD. A voting based plane fitting algorithm is used. Plane fitting algorithm, described by the authors, exploits additional information involved in a weight reflecting the matching reliability of each pixel.

Bleyer and Gelautz adopted similar steps but used a layered representation with different cost function [5]. Alternative planar model was used for the processing. Tombari et al. used symmetric aggregation strategy by applying a segmentation on both reference and target images [6]. The concept of color proximity is expanded more generally.

Hosni et al. have proposed the usage of geodesic distance as a support of segmentation process [7]. Yang et al. combined together the color segmentation, color-weighted correlation, hierarchical BP, left/right checking, plane fitting and

depth enhancement [8]. Yang refers to a novel technique used for pixel classification. The proposed optimization model takes into account current understanding of which pixels are stable, unstable (due to poor texture) or occluded. The discrimination of stable and unstable pixels is connected with correlation confidence, while the occluded regions are detected by mutual consistency check.

### 3 Mean Shift Segmentation Algorithm

Mean shift is a non-parametric feature-space analysis technique, founded on a density gradient estimation using a generalized kernel approach. It is based on an observation that the value of a density function can be estimated using the sample observations falling into a small area around the point. The feature space can be considered as the empirical probability density function of the represented parameter [9]. The algorithm seeks for the local maxima of the probability density function (given by the samples) which correspond to the dense regions of the sample space. Once the local maxima are located, segments can be recognized as the clusters associated with it.

In the majority of mean shift implementation the kernel density estimator is defined as follows: Assume you have  $n$  data samples  $x_1, \dots, x_n$  in the  $d$ -dimensional space  $R^d$ . The kernel density estimator is defined as function

$$\tilde{f} = \frac{1}{nh^d} \sum_{i=0}^n K\left(\frac{\mathbf{x} - \mathbf{x}_i}{h}\right), \quad (1)$$

where  $h > 0$  is the single bandwidth parameter and  $K$  is radially symmetric kernel satisfying

$$K(\mathbf{x}) = c_k k(\|\mathbf{x}\|^2), \quad (2)$$

with normalization constant  $c_k$  and function  $k(x)$ ,  $x > 0$  called the kernel profile. For the profile function  $k(x)$ , usually the Epanechnikov kernel is used (other types of kernel functions as uniform, triangular or Gaussian kernel can be used as well). Epanechnikov kernel is defined as

$$k(x) = \begin{cases} 1 - x & 0 \leq x \leq 1 \\ 0 & x > 1 \end{cases} \quad (3)$$

The density maxima is found when the gradient of density estimator  $\nabla \tilde{f}$  is equal to zero. The density gradient estimator can be rewritten as follows:

$$\nabla \tilde{f} = \frac{2c_k}{nh^{d+2}} \left( \sum_{i=0}^n g_i \right) \left( \frac{\sum_{i=0}^n \mathbf{x}_i g_i}{\sum_{i=0}^n g_i} - \mathbf{x} \right), \quad (4)$$

where  $g(x) = k'(x)$  and  $g_i = g\left(\left\|\frac{(\mathbf{x}-\mathbf{x}_i)}{h}\right\|^2\right)$ . The second term in brackets is the mean shift vector ( $m(x)$ ). Mean shift vector points to the region of maximum point density. For each sample we compute successive locations, given by the shifts of the mean shift vector. Thus we can define a path to a convergence

point, referred as an attractor. For the segmentation purposes, we have to group all attractors closer than given threshold in the spatial and range domain. In the next section we will describe the process of exploiting the known attractors of both images for accelerating the initial stereo matching.

Mean shift procedure was firstly described in 1975 by Fukunaga and Hostetler [10] and reintroduced in 1995 by Cheng [11]. Readers who are interested in detailed description of mean shift algorithm and its applications in low-level vision tasks and image segmentation can also read a comprehensive study written by Comaniciu and Meer entitled "Mean Shift: A Robust Approach Toward Feature Space Analysis" [9].

## 4 Matching in the Attractor Space

The basic idea underlying this approach is quite simple. As we have mentioned before, some of the stereo matching algorithms apply a mean shift segmentation on the pair of input images. In our approach, we will reuse certain partial results from the segmentation in the matching process. Because the mean shift is already done, it is achieved at almost no additional computation cost. The aim is to narrow down the search area of the SAD-based algorithm which follows the segmentation.

Our algorithm can be divided into three steps: (i) Attractor clustering; grouping the convergence points to represent more coherent objects. This step is necessary because the mean shift tends to over segment the input image. (ii) Attractor matching; establishing the matching between the attractor clusters. The goal is to match as many pairs of attractor clusters as possible. We are matching the segments of the same object as seen on the left and right image. (iii) Calculating the attractor disparity and utilizing the value in the fixed window-based algorithm to improve the performance. Having the correct matches we can calculate the segment's disparity. This disparity is used as the estimation of the window center of the SAD-based algorithm. The computation of SAD scores was optimized according to [12]. We have used windows of a fixed size.

In the next paragraphs, we will describe further utilization of the mean shift clustering and its attractor domain in more detail.

### 4.1 Attractor Clustering

Let  $x_i^L, x_i^R, i = 1, \dots, n$ , be the pixels of input images,  $z_i^L, z_i^R$  are the mean shift convergence points of the left and right image, calculated as described in the Section [3].

We create attractor clusters for each image  $\{A_k^L\}_{k=1\dots p}$  and  $\{A_l^R\}_{l=1\dots q}$  by grouping those convergence points  $z_i^L (z_i^R)$ , respectively, which are closer in spatial domain than threshold  $d_A$ . The cluster can be seen as a group of points in the attraction basin, a region where convergence paths lead. A simple agglomerative clustering, starting with each attractor in a separate cluster and merging iteratively those clusters that have distance less than  $d_A$  is not effective. A more

convenient method is to map the convergence points into the fixed grid and connect them using component labeling algorithm, e.g. flood fill. We recommend using the second method as it is faster, and clusters can be easily accessed since the grid is closely related to the original image grid.

For each attractor cluster we calculate its centroid ( $C = \{C_x, C_y, C_c, C_w\}$ , representing the  $x, y$ -coordinates, color and number of convergence points. This is an auxiliary structure that is used to improve the performance and will be used in the next step.

The primary motivation of this step is to provide clustering, which will separate the image into the coherent objects. It is assumed that the same object will be decomposed into similar group of segments on both images, despite the perspective projection. The idea is that we do not have to do pixel-to-pixel matching of the objects, as far as we are able to disassemble them into the matching segments. In the next step, we will describe the matching process itself.

## 4.2 Attractor Matching

The attractor matching is done between the sets of  $A_k^L$  and  $A_l^R$ . We are looking for the most acceptable attractor cluster of the right image, to match it with the attractor cluster of the left image. The search range is limited to the area bounded by  $[d_{min}, d_{max}]$ , the minimum and maximum allowed disparity in the horizontal direction and  $[min(z_{y_i}^L), max(z_{y_i}^L)]$ ,  $z_i^L \in A_m^L$ , minimum and maximum  $y$ -coordinates of samples connected with the given cluster in the vertical direction. We are looking for the matches of clusters sharing the similar color and cardinality.

A matching factor is computed as the weighted sum of squared differences of average attractor color and size (the same results can be obtained using already calculated values  $C_c, C_w$  of each attractor). The cost of matching is defined as follows:

$$\begin{aligned} M_c(k, l) &= w_1 \left( \frac{\sum_{z_c^L \in A_k^L} z_c^L}{|A_k^L|} - \frac{\sum_{z_c^R \in A_l^R} z_c^R}{|A_l^R|} \right)^2 + w_2 (|A_k^L| - |A_l^R|)^2 \\ &= w_1 (C_c(k) - C_c(l))^2 + w_2 (C_w(k) - C_w(l))^2, \end{aligned} \quad (5)$$

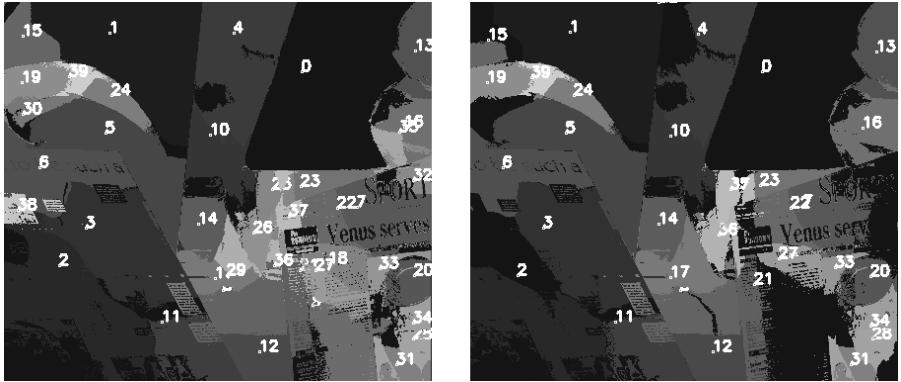
where the parameter  $w_1$  controls the strength of the color constraint. We suppose that the matching attractors will have almost the same average color. The parameter  $w_2$  controls the strength of the cardinality constraint, assuming the number of convergence points should be approximately the same.

Despite the fact that these constraints induce certain matching errors, they were chosen as the compromise that will result in an acceptable calculation speed. We strongly suggest sorting the attractor clusters according to its cardinality and starting the matching with the largest sets.

The best match between the attractor clusters is the one that minimizes the matching cost:

$$M_{best}(k) = \arg \min_l M_c(k, l), \quad (6)$$

where the possible matches are located in the searched area as described before. Figure 1 shows an example of the matching produced by our approach. Matching segments are labeled and colored for better identification. In practice, we match only large segments, since the small ones are ambiguous for matching.



**Fig. 1.** Results of attractor matching. Visualisation of matched image segments (Venus pair) as found by our algorithm. Matching segments are labeled according to its size and colored for better identification.

### 4.3 Improving Local Matching

After the matching, we calculate the  $y$ -coordinate difference of the attractor centers given by

$$d_i = C_y(M_{best}(k)) - C_y(k). \quad (7)$$

This value will be used to narrow down the number of possible matches of the SAD-based matching. We assume that the matching window will be found in the interval  $\langle d_i - S_c, d_i + S_c \rangle$ , where  $S_c$  is the parameter defining the width of search area. This parameter affects allowed point deviation from the supposed segment plane. In this way we have narrowed down the number of possible matches from  $\langle d_{min}, d_{max} \rangle$  to  $\langle d_i - S_c, d_i + S_c \rangle$ , in most cases resulting in removal of the highly improbable positions. The remaining successive steps, by which the final disparity is calculated and optimized, remain the same as in original algorithms (e.g. [34]).

#### 4.4 Algorithm Summary

This paper presents a modification of the first stage of the current stereo matching algorithms based on the mean shift clustering. This is one of the possible utilization of the attractor matching approach.

We can summarize our approach in the paper as follows: (i) perform the mean shift clustering on both input images, (ii) group the points of convergence into the attractor clusters, (iii) match the attractor clusters between the images, (iv) obtain the difference of matched cluster positions, (v) utilize this value to narrow down the search area of SAD-based algorithm.

## 5 Results

The experimental results of our approach are shown in Table II. It is the application of the stereo matching acceleration based on the attractor matching. All the tests were performed on the datasets provided by [2][13]. For the performance evaluation we followed the methodology proposed by Scharstein and Szeliski [2].

The algorithm was run with constant parameters on all four image pairs. The error rate (Err) is calculated as the percentage ratio of bad matching (where the absolute disparity error is greater than 1 pixel) of the original non-optimized initial matching and our approach. We provided only relative error rate, since the absolute values might be misleading, because they represent the initial matching, which is further improved by the optimization steps (they are not the subject of our paper).

In order to evaluate the speed of our algorithm, we have compared the number of SAD calculations. The percentage speed-up (Sp) vary across the images and is strongly influenced by the parameter  $S_c$ . This parameter controls the number

**Table 1.** Comparing the results of the original local matching algorithm with the fixed size window ( $W_s$ ) and improved version with reduced number of potential matches ( $S_c$ ). Error rate (Err) is the difference between the original and new one (the lower the better) and relative speedup (Sp).

Image pair	$W_s$	$S_c = 3$		$S_c = 7$		$S_c = 11$	
		Err [%]	Sp [%]	Err [%]	Sp [%]	Err [%]	Sp [%]
<i>Tsukuba</i>	3	<b>-1.61</b>	<b>+16.00</b>	+0.96	+11.00	+3.22	+7.00
	5	+4.11	+16.00	+4.57	+11.00	+5.94	+7.00
	7	+9.66	+17.00	+9.09	+11.00	+9.09	+6.80
	9	+12.82	+17.00	+10.26	+12.00	+10.90	+7.00
<i>Venus</i>	3	-27.55	+34.00	-6.26	+26.00	-4.29	+19.00
	5	<b>-28.00</b>	<b>+35.00</b>	-4.71	+26.00	-4.47	+19.00
	7	-26.33	+35.00	-2.07	+27.00	-4.14	+19.00
	9	-22.22	+35.00	+1.79	+27.00	+0.00	+19.00
<i>Teddy</i>	3	+1.08	+19.00	-1.22	+18.00	<b>-1.49</b>	<b>+17.00</b>
	5	+3.87	+19.00	+0.62	+18.00	-0.31	+17.00
	7	+5.53	+20.00	+1.68	+18.00	+0.34	+17.00
	9	+6.84	+20.00	+2.63	+18.00	+0.88	+17.00
<i>Cones</i>	3	+1.57	+8.00	-0.52	+7.00	-1.18	+7.00
	5	+0.00	+8.00	-2.72	+7.00	<b>-3.72</b>	<b>+7.00</b>
	7	+4.29	+8.00	+1.11	+7.00	-0.32	+7.00
	9	+4.92	+8.00	+1.48	+7.00	-0.16	+7.00

of potential matches searched around the disparity estimation based on matched attractor clusters. The total speed-up depends also on the hit rate of the attractor matching.

Our approach is primarily meant for urban and manmade sceneries with large textureless regions. An analysis of the results verifies that. Our algorithm is most successful on images with large areas of homogeneous textures such as Venus image pair. The speed improvement was over 30% reducing the error rate by more than 25%. The acceleration was not so significant on the other images and the error rate remained at the same level. The best results for each image were highlighted in the table.

## 6 Conclusions

In this paper, we have proposed an algorithm improving the speed of the initial disparity estimation of the stereo matching algorithms. Moreover, we have experimentally shown that our algorithm produces fewer errors when processing images with large homogeneous surfaces. The purpose of this paper was not to provide the complete and reliable stereo matching algorithm, but to improving the speed of existing one. Our main contribution lies in enhancing the effectiveness of the current stereo matching algorithms based on the mean shift clustering.

Although our optimization approach does not guarantee the significant improvements for all possible inputs, we have achieved promising results on images with large areas of homogeneous textures.

We see great potential in stereo matching in the mean shift attractor space. Additional studies are needed to fully exploit this approach. The number of missing matches, especially between the small segments, suggests that the limits of our approach have not been reached.

## Acknowledgement

This work was partially supported by the grant FR-TI1/262 of the Ministry of Industry and Trade of the Czech Republic.

## References

1. Cyganek, B.: *An Introduction to 3D Computer Vision Techniques and Algorithms*. John Wiley & Sons, Chichester (2007)
2. Scharstein, D., Szeliski, R.: A taxonomy and evaluation of dense two-frame stereo correspondence algorithms. *International Journal of Computer Vision* 47, 7–42 (2002)
3. Klaus, A., Sormann, M., Karner, K.F.: Segment-based stereo matching using belief propagation and a self-adapting dissimilarity measure. In: ICPR, vol. (3), pp. 15–18. IEEE Computer Society, Los Alamitos (2006)
4. Wang, Z.F., Zheng, Z.G.: A region based stereo matching algorithm using cooperative optimization. In: CVPR. IEEE Computer Society, Los Alamitos (2008)

5. Bleyer, M., Gelautz, M.: A layered stereo algorithm using image segmentation and global visibility constraints. In: ICIP, pp. 2997–3000 (2004)
6. Tombari, F., Mattoccia, S., di Stefano, L.: Segmentation-based adaptive support for accurate stereo correspondence. In: Mery, D., Rueda, L. (eds.) PSIVT 2007. LNCS, vol. 4872, pp. 427–438. Springer, Heidelberg (2007)
7. Hosni, A., Bleyer, M., Gelautz, M., Rhemann, C.: Local stereo matching using geodesic support weights. In: Submitted to IEEE International Conference on Image Processing, ICIP (2009)
8. Yang, Q., Engels, C., Akbarzadeh, A.: Near real-time stereo for weakly-textured scenes. In: British Machine Vision Conference, BMVC (2008)
9. Comaniciu, D., Meer, P., Member, S.: Mean shift: A robust approach toward feature space analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 24, 603–619 (2002)
10. Fukunaga, K., Hostetler, L.: The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Transactions on Information Theory* 21, 32–40 (1975)
11. Cheng, Y.: Mean shift, mode seeking, and clustering. *IEEE Trans. Pattern Anal. Mach. Intell.* 17, 790–799 (1995)
12. Stefano, L.D., Marchionni, M., Mattoccia, S., Neri, G.: A fast area-based stereo matching algorithm. *Image and Vision Computing* 22, 983–1005 (2002)
13. Scharstein, D., Szeliski, R.: High-accuracy stereo depth maps using structured light. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR 2003), Madison, WI, USA, pp. 195–202 (2003)

# Undecimated Wavelet Transform-Based Image Interpolation

Numan Unaldi<sup>1</sup> and Vijayan K. Asari<sup>2</sup>

<sup>1</sup> Dept. of EE, Turkish Air Force Academy, Istanbul, 34149, Turkiye  
nunaldi@hho.edu.tr

<sup>2</sup> Dept. of ECE, University of Dayton, OH, 45469, US  
Vijayan.Asari@notes.udayton.edu

**Abstract.** In this paper, a new wavelet-based image interpolation algorithm is developed for magnifying the image details so that the visibility of tiny features in a digital image is improved. The algorithm takes the LR image as the low-pass filtered subband of an unknown wavelet transformed high resolution image. Then an initial HR image of size twice the LR image is estimated using zero padding of the details. The HR image is transformed via UWT resulting in four subbands, three of which are related with the high frequency components of the image. In the UWT domain, the LL subband is replaced with the initially estimated HR image and applying the inverse UWT, the final HR image is determined. Experiments conducted with both gray level and color images show the superiority of the proposed algorithm over the state-of-the-art interpolation methods.

## 1 Introduction

In many image processing applications, magnifying the details in an image is often required, especially when the resolution is limited. Digital satellite, aerial images, and medical imaging along with distant object/face recognition are examples of such applications. It is necessary to make the magnification (interpolation) without blurring for the magnified details to be useful for object/face recognition.

Conventional techniques such as spline interpolation, nearest neighbor interpolation, bilinear interpolation, cubic convolution, b-spline, and tapered sinc [1]-[4] utilize the coherence of adjacent points. Although these techniques have advantages in simplicity and fast implementation, the result of these interpolations may degrade fine details in an image. Linear interpolation tries to fit a straight line between two points. This technique leads to blurred image. Pixel replication copies neighboring pixel to the empty location. This technique tends to produce blocky images. Approaches like spline and sinc interpolation are proposed to reduce these two extremities. Spline interpolation is inherently a smoothing operation, while sinc produces ripples (the Gibbs phenomenon) in the output image.

The Human Visual System is highly sensitive to edges, which play a key role in the perception of an image as high quality. Therefore, a good interpolation algorithm possesses a requirement to correctly reconstruct the original scene edges or at least maintain the sharp edges in the scene. Recently, nonlinear interpolation techniques

[5]-[11] have been developed to fulfill this requirement. Wavelet-based interpolation techniques [12]-[20] have been widely used for performing image interpolation for more than a decade. The input image is usually treated as the low-pass filtered subbands of an unknown wavelet-transformed high-resolution image, and then the unknown high-resolution image is produced by estimating the wavelet coefficients of the high-pass filtered subbands.

In this paper, a new wavelet-based image interpolation algorithm is introduced for magnifying the image details so that the visibility is improved. The image in hand is considered a low resolution (LR) image, and is treated as the approximation part (i.e. low-pass filtered subband) of an unknown wavelet transformed high resolution (HR) image. The detail coefficients (i.e. the missing high frequency content) are estimated using the undecimated wavelet transform (UWT) to obtain the unknown HR image. The proposed technique suggests a simple but efficient estimation for the high pass filtered subbands. Experiments conducted with both gray level and color images show the superiority of the proposed algorithm over the state-of-the-art interpolation methods.

## 2 Related Work

The simplest approach in wavelet interpolation is padding of the unknown high-pass filtered (detail) subbands with zeros and then taking the inverse wavelet transform [12]. Chang et al. [13] extrapolate the features in textured regions by examining the evolution of wavelet transform extrema and important singularities. Large magnitude coefficients are selected, since modeling for other coefficients is not easy. A least squares error criterion based approach is adopted to determine the corresponding extrema at the finest scale. Carey et al. [14] based on the same approach in [13], use the Lipschitz property, which states the wavelet coefficients corresponding to large singularities decay exponentially over scale [15]. At each index, an exponential fit over scale was used for wavelet coefficients to predict the detail coefficients at the finer scale. In both methods, only coefficients with large magnitude are used indicating moderate details cannot be treated this way. Moreover since the wavelet coefficients are formed by contributions from more than one coefficient in a neighborhood determined by the support of the filters used in the analysis, edge reforming based on extrema evolution that takes account of only significant magnitudes, affect the quality of edge reconstruction. Finally, signs of estimated coefficients are replicated directly from ‘parent’ coefficients without any attempt made to estimate the actual signs, implying that signs of the coefficients estimated using extrema evolution techniques cannot be relied upon.

Crouse et al. [16] propose the use of the Hidden Markov Model (HMM) for predicting wavelet coefficients over scales. In the training phase, HMM is trained using an image database. They predict the exact coefficient from the observed coefficient of a noisy image, for denoising application. The principle used here is that the coarser scale coefficients are less affected by noise, while the detail coefficients contain most of the noise. The same idea is extended to image zooming by Kinebuchi and Woo [17]. Greenspan et al. [18] and Burt et al. [19] both utilize the inter-scale dependency

(mainly related to edges) to extrapolate lost high-frequency components and use zero-crossings of the second derivative of smoothed images to locate edges. Based on the ideal step edge model, they estimate the interscale relations of edges in order to estimate edges at finer scales.

The decimated WT (DWT) is not shift-invariant and, as a result, suppression of wavelet coefficients, such as quantization of coefficients during the compression process or non-exact estimation of high-frequency subband coefficients, introduces cyclostationarity into the image which manifests itself as ringing in the neighborhood of discontinuities [12]. Temizel and Vlachos [12] propose a method in which initially estimated HR image via zero padding is shifted and wavelet transformed several times with different amount of shifts each time, followed by the inverse wavelet transform and the shift back to their original location. The results from each shift-transform cycle is then averaged in order to remove the ringing artifacts appeared before cycle-spinning process. Celik and Kusetogullari [20] present an interpolation technique using dual-tree complex wavelet transform (DT-CWT) [21] which exhibits approximate shift invariant property and improved directional resolution when compared that of the DWT. Although standard DWT gives good results in image compression, it is not optimal for other applications such as filtering, deconvolution, detection, or more generally, analysis of data [22]. The main reason for this is the lack of shift-invariance property in the DWT, which leads to many artifacts when an image is reconstructed after modification of its wavelet coefficients. Therefore an undecimated wavelet transform (UWT) in which the decimation step of the standard DWT is eliminated [23,24] can be used alternatively for other applications such as denoising [25].

### 3 Undecimated Wavelet Transform Using the “à Trou” Algorithm

The undecimated UWT of a 1D signal  $c_0$ , W using the filter bank  $(h, g)$  is a set  $W = \{w_1, \dots, w_J, c_J\}$  where  $w_j$  are the wavelet coefficients at scale  $j$  and  $c_J$  are the coefficients at the coarsest resolution. The “à trous” (meaning ‘with holes’ in French) algorithm [22]-[24] can be applied in order to obtain wavelet coefficients at one resolution from another using the following equations:

$$\begin{aligned} c_{j+1}[l] &= \left( \bar{h}^{(j)} * c_j \right)[l] = \sum_k h[k] c_j[l + 2^j k] \\ w_{j+1}[l] &= \left( \bar{g}^{(j)} * c_j \right)[l] = \sum_k g[k] c_j[l + 2^j k] \end{aligned} \quad (1)$$

where ‘\*’ is the convolution operator and  $\bar{h}[n] = h[-n], n \in \mathbb{Z}$  is the time-reversed of the discrete-time filter with an impulse response  $h[n]$  and  $h^{(j)}[l] = h[l]$  if  $l/2^j$  is an integer and 0 otherwise. For example when  $j=1$ ,

$$h^{(1)} = (\dots, h[-2], 0, h[-1], 0, h[0], 0, h[1], 0, h[2], \dots)$$

The reconstruction of the signal  $c_j$  is realized via:

$$c_j[l] = \frac{1}{2} \left[ (\tilde{h}^{(j)} * c_{j+1})[l] + (\tilde{g}^{(j)} * w_{j+1})[l] \right] \quad (2)$$

where  $\tilde{h}$  and  $\tilde{g}$  are the filters corresponding to analysis filter pairs  $h$  and  $g$ , respectively. The only exact reconstruction condition [22] for the filter bank  $(h, g, \tilde{h}, \tilde{g})$  is given by,

$$H(z^{-1})\tilde{H}(z) + G(z^{-1})\tilde{G}(z) = 1 \quad (3)$$

where  $H(z)$  is the z-transform of a filter  $h$  and so on. This condition determines how one should design the synthesis type filter bank given the analysis filters providing a higher degree of freedom when compared the DWT. Extension of the à trous algorithm to 2D is straightforward and can be realized by the convolution of  $c$  with the separable filter  $hg$  (i.e. convolution first along the columns by  $h$  and then convolution along the rows by  $g$ ). At each scale, three wavelet images,  $w^h, w^v, w^d$  each of which has the same size as the original image, representing edges along horizontal, vertical and diagonal directions.

In [22], it is shown that using non-bi-orthogonal filter banks, one can build the UWT. One example to this is:

$$\begin{aligned} h^{1D}[k] &= [1, 4, 6, 4, 1]/16, k = [-2, \dots, 2] \\ h[k, l] &= h^{1D}[k]h^{1D}[l] \\ g[k, l] &= \delta[k, l] - h[k, l] \end{aligned} \quad (4)$$

where  $\delta$  is defined as  $\delta[0,0] = 1$  and  $\delta[k, l] = 0$  otherwise. This filter bank is one widely used in analyzing the astronomical data. Following the exact reconstruction condition, it can be shown that for the above analysis filter bank  $\tilde{h} = \tilde{g} = \delta$  can be taken as synthesis filters yielding perfect reconstruction. Then just by co-additions of all scales perfectly reconstruct the original image:

$$c_0[k, l] = c_j[k, l] + \sum_{j=1}^J \sum_{n=1}^3 w_j^n[k, l] \quad (5)$$

where  $n$  stands for the three orientations at each scale. As previously stated, the non-subsampled nature of the decomposition allows one to reconstruct the original image from its wavelet transform in many ways. For a given filter bank  $(h, g)$ , any filter bank  $(\tilde{h}, \tilde{g})$  which satisfies the reconstruction condition given in (3), can be used in the reconstruction to obtain the original image. For example, for the analysis filters given in (4),  $\tilde{h} = h$  and  $\tilde{g} = \delta + h$  also constitute the prototype for the synthesis filter bank. For  $h = [1 4 6 4 1]/16$  corresponding reconstruction filter  $\tilde{g} = [1 4 22 4 1]/16$  is positive implying it is no longer related to a wavelet function.

## 4 The Proposed Algorithm

The first step in the implementation is wavelet domain zero padding (WZP) for an initial estimation of HR image. In WZP the unknown HR image is estimated by zero padding the high-frequency subbands and taking the inverse wavelet transform using the standard DWT. If the LR image in hand is denoted with  $s$  of size  $mxn$ , then the the initial estimate of the HR image is:

$$\hat{x} = \text{IDWT} \begin{bmatrix} s & \mathbf{0}_{mxn} \\ \mathbf{0}_{mxn} & \mathbf{0}_{mxn} \end{bmatrix} \quad (6)$$

where  $\mathbf{0}_{mxn}$  is the zero matrix of size  $mxn$  and IDWT is the inverse decimated wavelet transform.

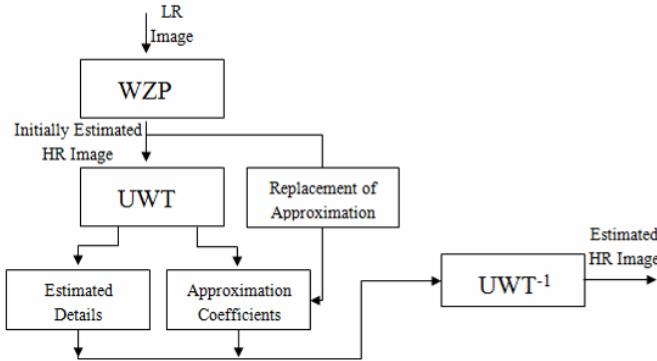
Although the results of the WZP interpolation introduce some blur into the reconstructed image due to low pass nature of the filters used in the inverse transform and lack of the high frequency components, it achieves higher *PSNR* values over bicubic interpolation and even over the sophisticated method of Carey et al. [14,26]. By successful estimation of the high-frequency subband coefficients, it is possible to improve the visual quality of the reconstructed images as well as the *PSNR* values indicating how close the produced images are to the actual HR images.

The decimated wavelet transform is not shift-invariant and, as a result, non-exact estimation of high-frequency subband coefficients introduces cyclostationarity into the image which manifests itself as ringing in the neighborhood of discontinuities [12]. Celik and Kusetogullari [20] present an interpolation technique using the dual-tree complex wavelet transform (DT-CWT) [21] that exhibits approximate shift invariant property and improved directional resolution when compared that of the DWT. In this implementation, the UWT is employed to the WZP-produced HR images in order to provide a good estimation for the detail coefficients in the second step of the proposed technique. The filter bank given in (4) is used and one-level decomposition is applied to the WZP processed image to produce the estimated coefficients.

In the third step, the approximation coefficients obtained from the UWT decomposition are replaced with the WZP processed image in a different way to those techniques appear in literature in which the LR image in hand is taken as the low-frequency subband of the DWT. Finally taking the inverse UWT, the estimated final HR image is produced. The scheme of the proposed algorithm is illustrated in Fig. 1. where  $\text{UWT}^{-1}$  indicates the inverse undecimated wavelet transform.

## 5 Experimental Results and Discussion

In this paper, two different image quality metrics between the original HR images and the reconstructed images from the simulated LR images are utilized to provide objective performance comparisons: (1) Peak Signal-to-Noise Ratio (*PSNR*) (2) Universal Quality Index (QI) proposed by Wang and Bovik [27]. QI indicates the similarity between the reference image and the processed one. It is defined as:



**Fig. 1.** The proposed interpolation algorithm

$$Q = \frac{4\sigma_{xy}\bar{x}\bar{y}}{(\sigma_x^2 + \sigma_y^2)(\bar{x}^2 + \bar{y}^2)} \quad (7)$$

where  $\bar{x}$  and  $\bar{y}$  are the mean values while  $\sigma_x^2$  and  $\sigma_y^2$  indicate variances of the reference and processed images, respectively and  $\sigma_{xy}$  is the covariance between the two images. The range of the Q index is [-1,1] in which 1 is the best value calculated if two images are identical. The quality index constitutes a measure for determining the distortion as a combination of three different factors, i.e. loss of correlation, luminance and contrast distortions. In addition to the objective quality assessment, the results from the proposed method, along with the other state-of-the-art interpolation techniques are illustrated to show the degree of the visual quality of the proposed method.

Six test images of size 512x512 which are illustrated in Fig. 2. are used in the experiments to show the capability of the proposed algorithm. These images are chosen for comparison since they are widely used for this purpose in literature because they provide both high and low frequency content simultaneously. The LR images are simulated from the images shown in Fig. 2. by low-pass filtering with a 3x3 average filter as a point spread function (PSF) of the imaging system and down sampling by 2 along each dimension. The final HR images are reconstructed again using a magnification factor of 2 as commonly seen in literature [12]-[20].

The power of an interpolation algorithms can then be determined based on how much the estimated HR image resembles the actual HR image visually by simply showing the interpolation results from the proposed method and statistically using the two metrics introduced in the previous section. Comparisons between the proposed and other techniques are provided, that are two well-known conventional techniques, bilinear and bicubic interpolation, two wavelet domain techniques [12,20] and two state-of-the-art spatial domain edge-based interpolation techniques [6,11] along with the WZP interpolation.

As stated previously, WZP is realized with the CDF 9/7 filters. In the second step of the algorithm, i.e. the estimation of the detail coefficients employs the filter bank of (4), and the synthesis filters  $\tilde{h} = h$  and  $\tilde{g} = \delta + h$  for the inverse UWT at the final reconstruction step.



**Fig. 2.** Test images: Left-to-right: Peppers, Bridge, Elaine, Barbara, Boat; and Lena

In Fig. 3. an example of the interpolation result is shown. In the figure the HR and simulated LR Barbara images, the result from the proposed algorithm applied to the LR image along with the initial WZP estimated HR image are given to illustrate the impact of the algorithm. Notice how the UWT based detail estimation improves the visual quality of the initially estimated HR image. Local contrast and sharpness of the WZP-processed images are improved after introducing the estimated details while most of distortions seen in the LR images are removed.



**Fig. 3.** The result of the interpolation algorithm. Left to right: Original image, simulated LR image, WZP result, Final reconstructed image.

The *PSNR* values resulting from the various interpolation methods are given in Table 1, while the Quality Index comparison is illustrated in Table 2. From both tables it can be inferred that the proposed algorithm outperforms other methods.

**Table 1.** The PSNR(dB) results for various interpolation methods. The LR images are simulated from corresponding HR images using an average filter of size 3x3 and downsampling by 2 in both dimensions.

Test Images	METHODS						
	Bilinear	Bicubic	NEDI [6]	EGI [11]	CS [12]	Method in [20]	Proposed
Peppers	29.83	30.29	32.24	32.54	33.02	33.75	<b>33.84</b>
Bridge	24.64	25.07	25.66	25.99	26.58	26.60	<b>27.46</b>
Elaine	30.71	31.04	32.23	32.43	32.73	33.00	<b>33.08</b>
Barbara	23.92	24.12	24.45	24.61	24.81	25.13	<b>25.39</b>
Boat	26.86	27.34	28.50	28.73	29.27	29.80	<b>30.19</b>
Lena	29.70	30.22	32.38	32.59	33.29	33.93	<b>34.39</b>

**Table 2.** The Quality Index[27] results for the same experiment explained within the caption of Table 1

Test Images	METHODS						
	Bilinear	Bicubic	NEDI [6]	EGI [11]	CS [12]	Method in [20]	Proposed
Peppers	0.6532	0.6766	0.6862	0.6980	0.7187	0.73	<b>0.7449</b>
Bridge	0.6637	0.7080	0.7237	0.7455	0.7825	0.79	<b>0.8439</b>
Elaine	0.6138	0.6356	0.6418	0.6529	0.6727	0.69	<b>0.7078</b>
Barbara	0.6043	0.6335	0.6600	0.6563	0.6811	0.71	<b>0.7364</b>
Boat	0.6072	0.6429	0.6609	0.6779	0.7042	0.73	<b>0.7521</b>
Lena	0.6793	0.7087	0.7290	0.7419	0.7661	0.79	<b>0.8071</b>

In Fig. 4. the original Bridge image is shown as well as the interpolation results for both images respectively to illustrate the degree of the visual quality of the proposed interpolations. It can be seen from the figures that among all the interpolation results, the closest to the original one in terms of local contrast and illumination are the the result of the proposed algorithm. Bilinear and bicubic result suffer from blurring. While methods of [6] and [11] both uses edge-based interpolation techniques they provide good interpolations especially along strong edges producing thin edges in the processed images, however they cannot provide sufficient sharpness for the entire texture of the image scene [11]. The images produced by the Cycle Spinning method [12] almost have the same appearance with WZP processed images except for a mild improvement around the edges. The resulting sharpness of the image is not as good as the one produced by the proposed algorithm.



**Fig. 4.** Interpolation results of the image *Bridge*. Left-to-right: Top row: Original, Bilinear, Bicubic, NEDI [6]; Bottom row: EGI [11], Cycle Spinning [12], Proposed.

In Fig. 5. the residual images, i.e. the difference image between the reconstructed image and the ground truth image are shown. Clearly, the proposed algorithm produces the best result when compared to the benchmark methods.

The proposed algorithm can also be applied for color image interpolation. The interpolations are carried out in each color channel separately. Experiments [28] conducted with the color images show that the highest PSNR and QI values are produced again by the proposed method when compared to the benchmark algorithms.



**Fig. 5.** Comparison of the residual images between the original *Elaine* image and images reconstructed from LR *Elaine* image. Left-to-right: Top row: Original, Bilinear, Bicubic, NEDI [6]; Bottom row: EGI [11], Cycle Spinning [22], Proposed.

## 6 Conclusion

In this paper, a new wavelet-based image interpolation algorithm for the improvement of the visibility of tiny features in an image is proposed. The UWT is employed to estimate the lost high frequency content in the LR images. Experiments conducted with both gray level and color images show improved PSNR and QI values for the proposed algorithm compared to both conventional and to the recent interpolation methods.

## References

1. Castleman, K.R.: *Digital Image Processing*. Prentice-Hall International, Inc., Englewood Cliffs (1996)
2. Unser, M., Aldroubi, A., Eden, M.: Enlargement or Reduction of Digital Images with Minimum Loss of Information. *IEEE Trans. on Image Processing* 4, 247–258 (1995)
3. Keys, R.G.: Cubic convolution interpolation for digital image processing. *IEEE Trans. on Acoustics, Speech, Signal Processing ASSP-29*, 1153–1160 (1981)
4. Fomel, S.: Three-dimensional seismic data regularization. Ph.D. dissert., Stanford University (2000)
5. Jensen, K., Anastassiou, D.: Subpixel edge localization and the interpolation of still images. *IEEE Trans. on Image Processing* 4(3), 285–295 (1995)
6. Li, X., Orchard, M.T.: New edge-directed interpolation. *IEEE Trans. on Image Processing* 10(10), 1521–1527 (2001)

7. Carrato, S., Tenze, L.: A high quality image interpolator. *IEEE Signal Processing Letters* 7(6), 132–135 (2000)
8. Takahashi, Y., Taguchi, A.: An enlargement method of digital images with the prediction of high-frequency components. In: Proc. Intl. Conf. Acoustics, Speech, Signal Processing, vol. 4, pp. 3700–3703 (2002)
9. Muresan, D.D.: Fast edge directed polynomial interpolation. In: Proc. Intl. Conf. Image Processing, vol. 2, pp. 990–993 (2005)
10. Malgouyres, F., Guichard, F.: Edge direction preserving image zooming: A mathematical and numerical analysis. *SIAM J. Numer. Anal.* 39, 1–37 (2001)
11. Zhang, L., Wu, X.: An edge-guided image interpolation algorithm via directional filtering and data fusion. *IEEE Trans. on Image Processing* 15(8), 2226–2238 (2006)
12. Temizel, A., Vlachos, T.: Wavelet domain image resolution enhancement using cycle-spinning. *IEE Electronics Letters* 41(3), 119–121 (2005)
13. Chang, G.G., Cvetkovic, Z., Vetterli, M.: Resolution enhancement of image using wavelet transform extrema interpolation. In: Proc. ICASSP, vol. 4, pp. 2379–2383 (1995)
14. Carey, W.K., Chuang, D.B., Hemami, S.S.: Regularity-preserving image interpolation. *IEEE Trans. on Image Processing* 8(9), 1293–1297 (1999)
15. Mallat, S.G., Zhong, S.: Characterization of signals from multiscale edges. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 14(7), 710–732 (1992)
16. Crouse, M.S., Nowak, R.D., Baraniuk, R.G.: Wavelet-based statistical signal processing using hidden markov models. *IEEE Trans. on Signal Processing* 46, 886–902 (1998)
17. Kinebuchi, K., Muresan, D., Parks, T.: Image interpolation using wavelet-based hidden markov trees. In: IEEE Int. Conf. Acoustics, Speech, Signal Processing, pp. 7–11 (2001)
18. Greenspan, H., Anderson, C., Akber, S.: Image enhancement by nonlinear extrapolation in frequency space. *IEEE Trans. on Image Processing* 9, 1035–1047 (2000)
19. Burt, P., Kolczbyski, R.: Enhanced image capture through fusion. In: Proc. Intl. Conf. on Computer Vision, Germany, pp. 173–182 (1993)
20. Celik, T., Kusetogullari, H.: Self-sampled image resolution enhancement using dual-tree complex wavelet transform. In: Proc. EUSIPCO 2009 Glasgow, Scotland (August 2009)
21. Kingsbury, N.G.: Complex wavelets for shift invariant analysis and filtering of signals. *Appl. and Comp. Harmon. Anal.* 10, 234–253 (2001)
22. Starck, J.L., Fadili, J., Murtagh, F.: The undecimated wavelet decomposition and its reconstruction. *IEEE Trans. on Image Processing* 16(2), 297–309 (2007)
23. Dutilleux, P.: An implementation of the “algorithme à trous” to compute the wavelet transform. In: Proc. Wavelets: Time-Frequency Methods and Phase-Space, pp. 298–304 (1989)
24. Shensa, M.J.: Discrete wavelet transforms: Wedding the à trous and Mallat algorithms. *IEEE Trans. on Signal Processing* 40(10), 2464–2482 (1992)
25. Starck, J.-L., Elad, M., Donoho, D.L.: Redundant multiscale transforms and their application for morphological component analysis. *Adv. Imaging Electron. Phys.* 132, 287–348 (2004)
26. Li, X.: Image resolution enhancement via data-driven parametric models in the wavelet space. *EURASIP Journal on Image and Video Processing* 2007, Article ID 41516 (2007)
27. Wang, Z., Bovik, A.C.: A universal image quality index. *IEEE Signal Processing Letters* 9(3), 81–84 (2002)
28. Unaldi, N.: Wavelet-based enhancement technique for visibility improvement of digital images. Ph.D. dissertation, Old Dominion University, Norfolk, VA, USA (2010)

# The Influence of Multimodal 3D Visualizations on Learning Acquisition

Phuong T. Do<sup>1,\*</sup>, John R. Moreland<sup>2</sup>, and Dennis P. Korchek<sup>3</sup>

<sup>1</sup> Department of Behavioral Sciences

<sup>2</sup> The Center for Innovation through Visualization and Simulation (CIVS)

<sup>3</sup> Department of Construction Science and Organizational Leadership

Purdue University Calumet, Hammond, IN 46323-2094, USA

**Abstract.** The present research addressed a critical barrier constantly facing developers and instructors involved in interactive web-based teaching. Participants were assigned to different multimodal training conditions (visual-, auditory-, audiovisual-modality, or no training) to learn and free recall a list of 14 terms associated with construction of a wood-frame house. The audiovisual- and visual-modality training conditions displayed comparable accuracy rates, while the auditory-modality training condition revealed lower accuracy, and the no-training condition exhibited little or no learning acquisition. The process of simultaneously exposing learners to interactive dynamic visualizations and prompting them to attend to information through the pragmatic use of audio cues reduced memory load, and in turn facilitated memory recall. Findings provided constructive feedback on the efficacy and usability of three-dimensional (*3D*) dynamic visualizations in web-based distance education, and implementations for future development of human-computer user interfaces to optimize engineering design effectiveness.

**Keywords:** *3D*, multimodal visualizations, learning and memory, virtual reality, e-learning.

## 1 Introduction

The categorization field has recently embraced the possibility that learning and transfer are mediated by an integration of information from multiple sensory modalities [1, 2]. The primary findings on multimodal perceptual effect focus on the combination of narration and two-dimensional animations [3]. Even so, these findings are drastically fewer than other instructional effects (e.g., split-attention, redundancy, and imagination effects) [4, 5]. Despite the research in elucidating the complex perceptual processes of modality effects on multimedia learning, the question of whether knowledge derived from stereoscopic *3D* dynamic visualization is mediated by an integration of information from multiple sensory modalities has not been fully explored.

---

\* Correspondence concerning this manuscript should be addressed to Phuong T. Do, Department of Behavioral Sciences, Purdue University Calumet, 2200 169th Street, Hammond, IN 46323-2094. Email: pdo@calumet.purdue.edu. Office: 219/989-2576. Fax: 219/989-2008.

The theoretical question of how to effectively and efficiently integrate schemata of existing concepts into our design of instructional visualization to enhance learning performance is proposed here. Schemata, representational structures in memory, originated from the work of Bartlett [6] in which he suggested that knowledge could be categorized into meaningful instances. The concept of a schema since then has been incorporated into the designs of artificial intelligence and robotic machinery. For instance, schemata play a central role in the data structures of computer software [7] and the general structure for biomedical informatics [8].

Three-dimensional (3D) dynamic information visualization is a relatively new field whose interdisciplinary nature incorporates computer and behavioral sciences [9]. The extent to which human cognitive abilities are being modulated by computer simulations and interactive interfaces is a fundamental component of our learning adaptability to recent advances in information technology [10-12].

The new movement towards web-based and multimedia learning has recently placed great emphases on interactive, dynamic visualizations [13, 14]. Dynamic visualizations have been shown to facilitate learning and teaching effectiveness [15, 16]. Interactive 3D virtual learning environments not only display a stimulating and informative framework for distance learners, but provide educators a communicative tool that can be utilized to foster students' experiential learning and constructive collaboration [17]. One of the many advantages of information visualization is that the web-based distance learners are able to examine their conceptual understanding of abstract concepts without viable repercussions. The process of engaging distance learners with interactive 3D virtual models exposes each individual to experiential learning (*ExL*) experiences by default, which often leads to a better conceptual understanding of the content [18]. Interactive, dynamic visualizations not only serve as a bridge between *ExL* and distance education but also provide instant feedback on teaching and learning effectiveness at no cost. The client-server communication on the Internet offers ample opportunities for educators to create virtual classrooms [19]. Thus, it is economical for both instructors and students to use dynamic visualizations as a medium to gain experiential learning in distance education.

Despite the potential benefits of embedding stereoscopic 3D content into our e-learning instructional curriculum, the reciprocal relationship between 3D human-computer interfaces and distance education is unclear. The current research explores the pragmatic application of 3D information visualization in traditional classroom environments. Undergraduate students completed a memory test determining their cognitive ability to free recall the definitions of 14 studied terms associated with the construction of a wood-frame house after going through a multimodal visualization learning phase. Two research questions are proposed: (1) which types of dynamic visualizations enable distant learners to gain experiential learning experience without inducing cognitive overload, and (2) how can psychological and technological disciplines be combined to develop effective research practices in new development of 3D interactive, dynamic visualizations? The theoretical goal of this work is to suggest how the present findings provide implications for theories of sensory integration, and perhaps propose a theoretical framework for testing dynamic visualization with recent technologies such as haptic user interfaces and stereoscopic 3D computer monitors.

## 2 Method

### 2.1 Participants

Eighty ( $N = 80$ ) Purdue University Calumet undergraduate students (29 males and 51 females) were randomly recruited to participate in the current study. Student participation was voluntary, and participation or nonparticipation did not affect their grades. All of the participants ranged in age from 18 to 25 years and had normal or corrected-to-normal visual acuity, and were naïve to the purpose of the experiment. A total of twenty participants were randomly assigned to one of the four learning-testing conditions. Probability sampling strategies and random selection were applied during the participant recruitment process to eliminate the potential for selection bias. Of the 72 participants who provided demographic information, 60% of the students were identified as Caucasians, 15% as African Americans, 4% as Asians, and 21% as other ethnic minorities.

### 2.2 Materials, Stimuli, and Apparatus

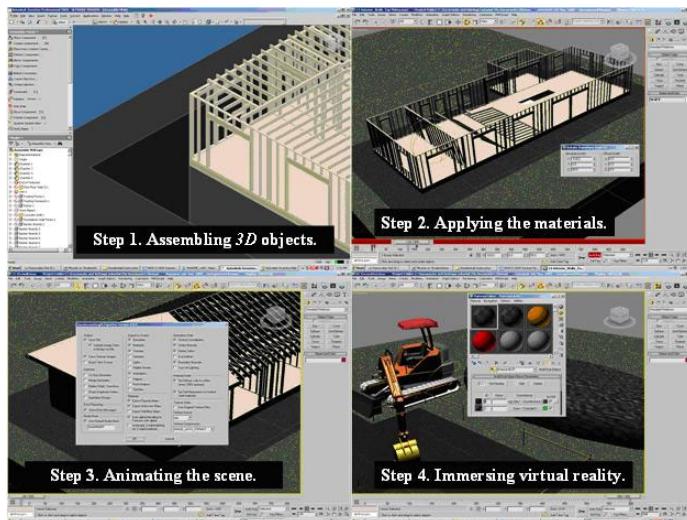
The learning stimuli consisted of a vocabulary list of 14 technical terms associated with the construction of a wood-frame house: beams, ceiling joist, columns, floor joist, floor sheathing, footing, formwork, foundation wall, header, mud sill, reinforcing, ribbon joist, top/bottom plate, and wall studs. These terms were presented in different contexts based on the training condition.

For the audiovisual and visual condition, a virtual walkthrough was used to assist participants in their comprehension of the wood-frame house construction and design while allowing them to interact with objects in 3D. The walkthrough was presented in an immersive virtual environment and was developed using a mixture of commercial, OpenSource, and custom-built software. The wood-frame house was constructed and animated using Autodesk 3dsMax (Figure 1).

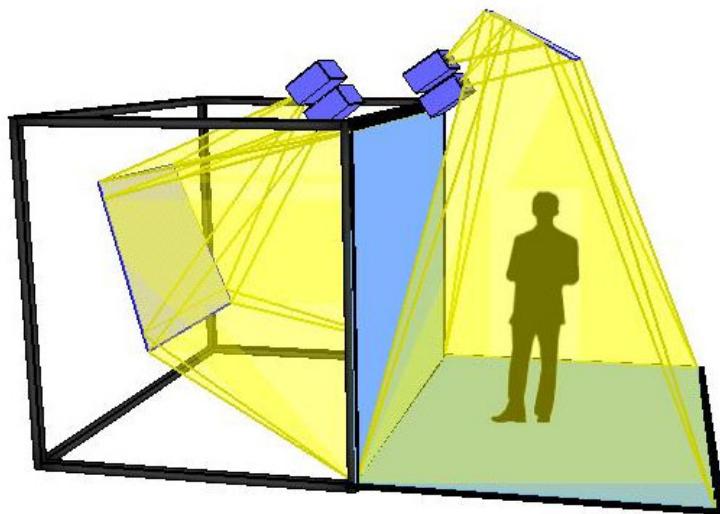
This artifact was then exported as a 3D mesh model that was loaded into a custom-built application that utilized the OpenSource VRJuggler & OpenSceneGraph libraries. Once loaded into the immersive environment, the different phases of construction were shown as a series of interactive stereoscopic 3D animations.

The immersive virtual environment was displayed using a virtual reality system that allowed participants to see and interact with data and objects in a three-dimensional space. The display consists of two large screens: a 7'6" x 10' rear projected wall, and a floor screen of the same size that participants stand on (Figure 2). Combined with 3D glasses, this configuration allows users to stand inside the visualization, with imagery coming out of both the screen in front of them and the floor beneath them.

The system uses four projectors (two per screen) to provide the 3D imagery (called Stereoscopic 3D) and creates an illusion of depth. Filters built into the projectors correspond with the special lenses in the 3D glasses so that each eye only sees an image from one projector in a stereoscopic pair. The 3D images are sent to the projectors from a high-end PC with four video outputs. While the system is also capable of tracking a user's head and hand movements within the space, this



**Fig. 1.** A brief demonstration of the 3D, dynamic visualization process of designing the immersive virtual reality environment of a wood-frame construction



**Fig. 2.** An illustration of the immersive virtual reality system used in the study, by VisBox Inc

functionality was not used in the present study. Following the learning phase, participants were given a written test asking them to match each term with its corresponding definition.

### 2.3 Research Design and Procedure

A cohort of five participants were randomly assigned to one of the four counterbalanced conditions. The four cohorts of participants for each condition were

counterbalanced and randomly rotated across the four training conditions (e.g., ABCD, BCDA, CDAB, DABC). A Greco-Latin square was used to assign participants to the four training conditions, with O1, O2, O3, O4; and Visual, Auditory, Audiovisual, No Learning representing the order of learning and training conditions, respectively. A total of five subjects in each condition were randomly assigned to each row of the resulting square. As a consequence, each participant was tested equally often by each training condition and learning order (Table 1).

**Table 1.** A Greco-Latin square representation

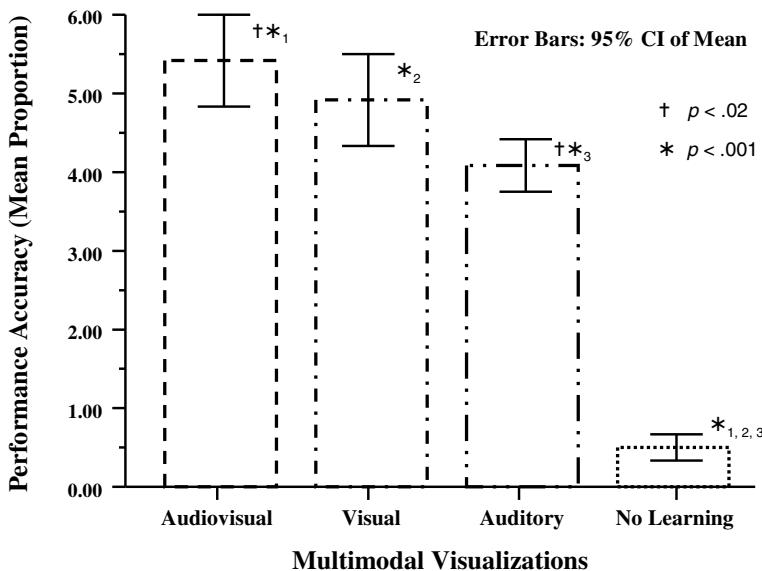
Participants	Training Conditions			
	Visual	Auditory	Audiovisual	No Training
01-05	O1	O2	O3	O4
06-10	O2	O3	O4	O1
11-15	O3	O4	O1	O2
16-20	O4	O1	O2	O3

Prior to the experiment, standard learning instructions were read to inform participants that all 14 technical terms were associated with wood-frame constructions. Participants were then presented a series of 14 terms and their definitions in the different contexts already described over a duration of 6 minutes to help them familiarize with the technical usage. Based on our pilot data and prior evidence that multimodal visualization had been shown to display a much faster processing time than visual or auditory training alone [20], the present experiment equated learning time intervals for each respective training condition. For the visual training condition, each learning word was carefully presented in concurrence with the demonstration of the virtual construction of a single-family wood framed house. For the auditory training condition, the recorded narrative was the primary learning source. For the audiovisual training condition, participants were visually exposed to the virtual depiction of the construction and they simultaneously heard the narrative as the wood framed house was being built in virtual reality. Thus, information was presented simultaneously in bimodal situations. It should be noted that the visual-auditory training modality condition had exposure of neither visual nor auditory response modality dominance. An imposition of a memory demand on vision or audio might bias learning judgments, due to the possibility of modality differences in memory [21]. Participants in the present study were instructed to rely on neither modality, but to use their cognitive ability to recalibrate multimodal information during learning. Following the learning phase, participants were then given standard

testing instructions for the transfer test, which occurred immediately after learning. The amount of time it took to complete the research activity ranged from 20 to 30 minutes.

### 3 Results

Data revealed a significant effect of multimodal visualizations on learning performance,  $F(3, 76) = 54.27, p < .001$  (Figure 3). The audiovisual condition displayed the highest mean proportion ( $M = 5.43, SD = 2.39, \sigma_M = .54$ ) as compared to the visual condition ( $M = 4.89, SD = 2.56, \sigma_M = .57$ ), the auditory condition ( $M = 4.11, SD = 1.21, \sigma_M = .27$ ), and the no learning condition ( $M = .54, SD = .55, \sigma_M = .12$ ). Such a difference is relatively large,  $\eta^2 = .68$ . It should be noted that the assumption of homogeneity of variance was violated. Hence, the Brown-Forsythe and the Welch  $F$ -ratios are reported,  $F(3, 48.06) = 54.27, p < .001$  and  $F(3, 36.34) = 153.38, p < .001$ , respectively.



**Fig. 3.** Learning performance accuracy as a function of multimodal visualizations

Table 2 shows the Bonferroni post hoc analysis. When participants were exposed to the auditory or visual training condition, they displayed relatively similar learning performance accuracy ( $p = .40$ ). The visual and audiovisual training conditions revealed similar performance accuracy such that the probability of obtaining any differences between these two group means was not probable ( $p = 1.00$ ). Participants performed equally well whether or not they learned the construction concepts visually or in both perceptual modalities (i.e., visual and auditory). However, participants showed different rates of learning acquisition when they were assigned to the auditory

**Table 2.** Bonferroni post hoc analysis of multimodal visualization performance accuracy

Bonferroni Analysis	Mean Difference	Significance*
Auditory vs. Visual	1.10	.40 – NS
Auditory vs. Audiovisual	1.85	.02 – SIG
Visual vs. Audiovisual	.75	1.00 – NS
No Learning vs. Auditory	5.00	.001 – SIG
No Learning vs. Visual	6.10	.001 – SIG
No Learning vs. Audiovisual	6.85	.001 – SIG

\*The mean difference is significant at the .05 alpha level.

or audiovisual training condition ( $p < .02^+$ ). It is interesting to note that the mean difference of performance accuracy between the no learning condition and the three multimodal visualization training conditions was significantly different ( $p < .001^*$ ).

## 4 Discussion

The goal of the present study was to implement the different types of multimodal training methods and evaluate the effectiveness of our visualization designs on learning acquisition. A significant effect of multimodal visualizations on learning performance was found. Evidence suggested that multimodal visualization training facilitated learning performance accuracy as compared to no training. Results provided strong support for meritorious, innovative, and interdisciplinary research by addressing a critical barrier to progress in the fields of behavioral sciences and visualization development of human-computer user interfaces.

Despite the medium flooring effect in performance accuracy, it should be noted that the audiovisual training condition showed considerably fewer errors as compared to the other conditions. This finding could be explained by the intersensory transfer between perception and action [22, 23]. Multimodal intersensory systems converged to coherently construct a final percept, which in turn enhanced learning performance. An alternative explanation was motivated by the hypothesis of intersensory integration where conflicting information provided by two perceptual modalities necessitated recalibration to form a final percept [1]. The notion that optimal learning depends neither on visual nor auditory modality, but rather both, reveals the bidirectional connection between sensory integration and memory dynamics. The

authors attempted to elucidate the sensory integration theory which predicted that sight and acoustic resonance converged to facilitate learning and guide subsequent transfer. It was found that performance accuracy was at least partially due to the overlapping of bimodal visualizations, which reinforces the existing literature on sensory integration [1, 2, 24-26]. To be more specific, the results from the preliminary study revealed that discourse information could be integrated between visual and auditory perceptual modalities to promote learning acquisition.

Nonetheless, the flooring effect in learning performance found in the present study might suggest that participants were not able to easily comprehend the construction terms with just one learning trial. Whether this effect could be remedied using multiple learning trials with and without feedback is the rationale behind the authors' next study [27].

Future studies should also examine the role domain-specific knowledge play on learning acquisition. Our visualization and simulation laboratory is currently exploring the extent to which domain-specific knowledge impact learning performance in web-based distance education. The purpose is to implement both the theoretical applications and pragmatic use of 3D interactive, dynamic visualizations on instructional curriculum. It is imperative to develop 3D visualizations effective for individual PC/MAC computers using a mouse and/or a joy stick as vehicles for navigation in virtual reality. Additionally, though stereoscopic 3D displays are available to the general public, 2D monitors are currently the standard and will likely continue to be used for the immediate future and relevant comparisons between 2D & 3D display should be factored into future work.

Inconsistent evidence has been reported regarding the effect of audiovisual perception in the design of dynamic visualizations. For example, perceptually integrated dynamic visualizations facilitated learning performance only when presented information was domain-specific, i.e., interactive learning environments were designed to complement the individual learners' cognitive processes, such as their different levels of knowledge expertise [28, 29]. Without taking into account the learner's knowledge base, dynamic visualizations in audiovisual modality are expected to attenuate learning compared to visual or audio alone [30]. The incorporation of domain knowledge into the design of dynamic visualizations has been largely overlooked. If not properly designed, interactive dynamic visualizations may result in a learning decrement due to memory overload and redundant information [31-33].

In summary, these results contribute two significant intellectual merits to the interdisciplinary field of cognition and visualizations. First, the preliminary findings serve as a conduit through which innovative teaching is built upon. Students will partake in research activities that facilitate the exchange of theoretical concepts and experiential learning knowledge reflected upon the lessons learned; such a collaborative infrastructure allows students to rise beyond the status quo of academia. Second, the prospective developments of dynamic visualizations in postsecondary education may facilitate learning curiosity in adult learners of various community outreach programs. The objective is to advance postsecondary education and college experiential learning experiences in new directions. Our preliminary findings provide innovative strategies to successfully design interactive, dynamic visualizations that optimize learning and teaching effectiveness. The proficient design of dynamic visualizations requires an understanding of the cognitive processes that underlie concept formations. Findings not

only enriches the usability of web-based learning tools in academic and professional settings, but presents knowledge-based visualization mechanisms that are central to teaching and training with state-of-the-art technologies.

## Acknowledgements

This research was supported in parts by the Purdue Research Foundation (PRF) and Northwest Indiana Computational Grid (NWICG) Project at Purdue University Calumet.

The authors wish to express their gratitude to Dr. Chenn Zhou for her invaluable assistance with the research. Thanks to the faculty members in the Department of Behavioral Sciences, the undergraduate/graduate research assistants, and many others who have contributed to our research project.

## References

1. Millar, S., Al-Attar, Z.: What aspects of vision facilitate haptic processing? *Brain and Cognition* 59, 258–268 (2005)
2. Ernst, M.O., Bulthoff, H.H.: Merging the senses into a robust percept. *Trends in Cognitive Sciences* 8, 162–169 (2004)
3. Mayer, R., Anderson, R.: The instructive animation: Helping students build connections between words and pictures in multimedia learning. *Journal of Educational Psychology* 84, 444–452 (1992)
4. Schwan, S., Garsoffky, B., Hesse, F.W.: Do film cuts facilitate the perceptual and cognitive organization of activity sequences? *Memory and Cognition* 28, 214–223 (2000)
5. Schwan, S., Riempp, R.: The cognitive benefits of interactive videos: Learning to tie nautical knots. *Learning and Instructions* 14(3), 293–305 (2004)
6. Bartlett, F.: *Remembering: A study in experimental and social psychology*. Cambridge University Press, London (1932)
7. Roth, S.F., Chuah, M.C., Kerpedjiev, S., Kolojejchick, J.A., Lucas, P.: Toward an information visualization workspace: combining multiple means of expression. *Human-Computer Interaction* 12, 131–185 (1997)
8. Covitz, P.A., Hartel, F., Schaefer, C., De Coronado, S., Fragoso, G., Sahni, H., Gustafson, S., Buetow, K.H.: caCORE: A common infrastructure for cancer informatics. *Bioinformatics* 19, 2404–2412 (2003)
9. Ainsworth, S., Van Labeke, N.: Multiple forms of dynamic representation. *Learning and Instruction* 14(3), 241–255 (2004)
10. Sweller, J.: Implications for cognitive load theory for multimedia learning. In: Mayer, R.E. (ed.) *Cambridge handbook of multimedia learning*, pp. 19–30. Cambridge University Press, New York (2005)
11. Mayer, R.E.: Principles for reducing extraneous processing in multimedia learning: coherence, signaling, redundancy, spatial contiguity, and temporal contiguity principles. In: Mayer, R.E. (ed.) *Cambridge handbook of multimedia learning*, pp. 183–200. Cambridge University Press, New York (2005)
12. Betrancourt, M.: The animation and interactivity principles in multimedia learning. In: Mayer, R.E. (ed.) *Cambridge handbook of multimedia learning*, pp. 287–296. Cambridge University Press, New York (2005)
13. Lowe, R.: Interrogation of a dynamic visualization during learning. *Learning and Instruction* 14, 257–274 (2004)

14. Chandler, P.: The crucial role of cognitive processes in the design of dynamic visualizations. *Learning and Instructions* 14, 353–357 (2004)
15. Linn, M.C., Eylon, B.S.: Science education: Integrating views of learning and instruction. In: Alexander, P.A., Winne, P.H. (eds.) *Handbook of educational psychology*, 2nd edn., pp. 511–544. Lawrence Erlbaum Associates, Mahwah (2006)
16. Tversky, B., Bauer-Morrison, J.B., Betrancourt, M.: Animation: Can it facilitate? *International Journal of Human-Computer Studies* 57, 247–262 (2002)
17. Dickey, M.D.: Teaching in 3D: Pedagogical affordances and constraints of 3D virtual worlds for synchronous distance learning. *Distance Education* 24, 105–121 (2003)
18. Dede, C., Salzman, M., Loftin, R.B.: The development of a virtual world for learning Newtonian mechanics. In: Brusilovsky, P., Kommers, P., Streitz, N.A. (eds.) *MHVR 1994. LNCS*, vol. 1077, pp. 87–106. Springer, Heidelberg (1996)
19. Mazza, R., Dimitrova, V.: Visualizing student tracking data to support instructors in web-based distance education. In: *Proceedings of the ACM Symposium on Computer Science Education*, New York, pp. 154–161 (2004)
20. Do, P.T.: Learning, generalization, and retention of haptic categories II (Doctoral dissertation, Arizona State University, Tempe) (2009)
21. Do, P.T., Ferguson, R., Kahol, K., Panchanathan, S., Homa, D.: Learning, generalization, and retention of haptic categories I. Poster presented at the Annual Conference of the Psychonomic Society in Chicago, IL (November 2008)
22. Gibson, J.J.: The senses considered as perceptual systems. Houghton Mifflin, Boston (1966)
23. Gibson, J.J.: The ecological approach to visual perception. Houghton Mifflin, Boston (1979)
24. Easton, R., Srinivas, K., Greene, A.: Do vision and haptics share common representations? Implicit and explicit memory within and between modalities. *Journal of Experimental Psychology: Learning, Memory, and Cognition* 23, 153–163 (1997)
25. Amedi, A., Jacobson, G., Hendler, T., Malach, R., Zohary, E.: Convergence of visual and tactile shape processing in the human lateral occipital complex. *Cerebral Cortex* 12, 1202–1212 (2002)
26. Amedi, A., Malach, R., Hendler, T., Peled, S., Zohary, E.: Visuo-haptic object-related activation in the ventral visual pathway. *Nature Neuroscience* 4, 324–330 (2001)
27. Do, P.T., Moreland, J., Korchek, D., Braun, D., Viswanathan, C.: The effect of multimodal visualizations, domain-specific knowledge, and rehearsal process on learning acquisition (in progress)
28. Do, P.T., Homa, D.: Transformational knowledge facilitated learning and transfer of abstract concepts. *Memory and Cognition* (2010) (manuscript submitted for publication)
29. Kalyuga, S., Ayres, P., Chandler, P., Sweller, J.: The expertise reversal effect. *Educational Psychologist* 38, 23–31 (2003)
30. Kalyuga, S., Chandler, P., Sweller, J.: Incorporating learner experience into the design of multimedia instruction. *Journal of Educational Psychology* 92, 126–136 (2000)
31. Sweller, J., van Merriënboer, J., Paas, F.: Cognitive architecture and instructional design. *Educational Psychology Review* 10, 251–296 (1998)
32. Mayer, R., Heiser, J., Lonn, S.: Cognitive constraints on multimedia learning: When presenting more material results in less understanding. *Journal of Educational Psychology* 93, 187–198 (2001)
33. Kirschner, P.A.: Cognitive load theory: implications of cognitive load theory on the design of learning. *Learning and Instruction* 12(1), 1–10 (2002)

# Visualizing Gene Co-expression as Google Maps

Radu Jianu and David H. Laidlaw

Computer Science Department, Brown University  
[{jr,dhl}@cs.brown.edu](mailto:{jr,dhl}@cs.brown.edu)

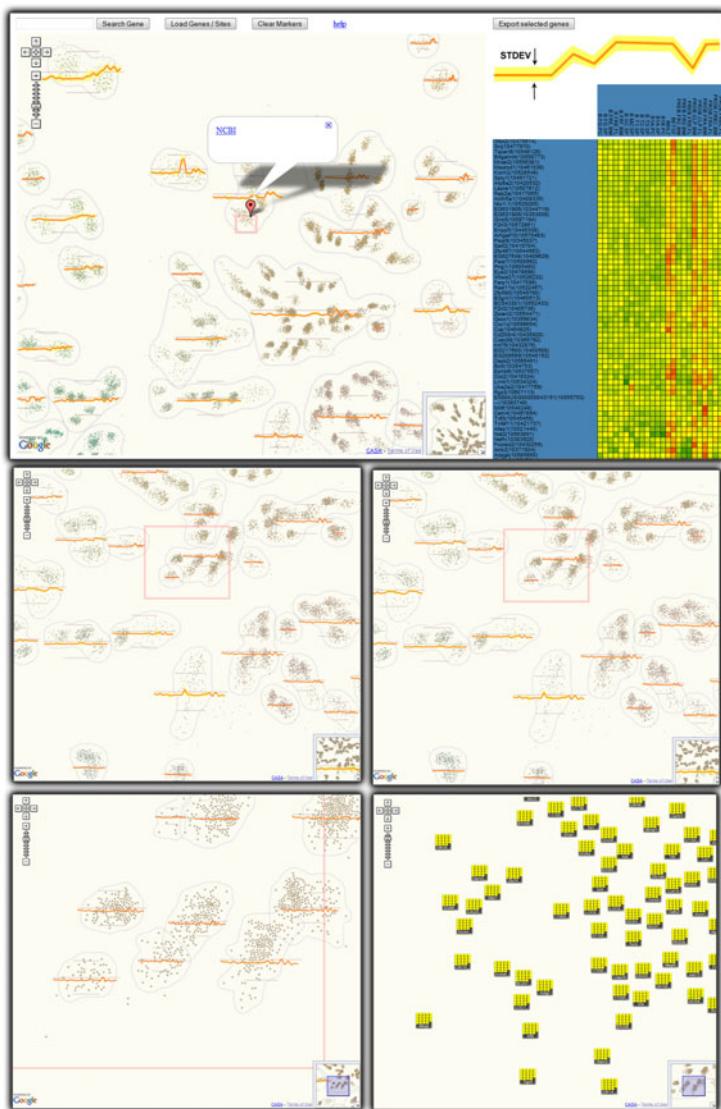
**Abstract.** We visualize gene co-regulation patterns by creating 2D embeddings from microarray data corresponding to complete gene sets from the mouse genome, across large numbers of cell types. We use google maps and client-side graphics to disseminate pre-rendered such visualizations with a small but intuitive set of interactions. We conduct an anecdotal evaluation with domain specialists and demonstrate that biologists appreciate this approach because it facilitates low-overhead access to readily analyzable perspectives of unfamiliar datasets and because it offers a convenient way of disseminating large datasets in visual form.

## 1 Introduction

Visualization of biological data ranges from websites that provide sparse, key-hole representations of database stored data, to complex stand-alone visualization systems with many options and analysis features. Both approaches have merit and are widely used, yet both have task-specific limitations. In terms of usability, the former have low visual expressivity, avoid complex computations and do not show large data volumes at once, while the latter have significant overhead associated with setting up and learning to control the environments. A drawback shared by both approaches is in disseminating data and results: biologists producing data lack the expertise required to set up and maintain a database-driven website; publishing raw data requires interested researchers to learn and operate specialized analysis software, increasing the overall invested effort.

In this context we introduce gene co-expression maps which are pre-rendered 2D embeddings of genomic multidimensional data. We show complete gene sets of around twenty-three thousand genes with expression measurements across tens of cell types (e.g. all members of a cell family). These images are served through the Google Maps API, and have a simple and intuitive set of interactions that can be learned with minimal overhead. An example of such a visualization is shown in Figure 1 (a demo can be accessed at [1]). We run an anecdotal user study and show that this approach is a viable solution for understanding genomic co-regulation patterns especially in large unfamiliar datasets, as well as for disseminating large micro-array datasets by data-intensive labs.

The motivation behind our work is to let labs publish data and results in visual form along with raw textual data so that users can access readily analyzable perspectives on the data without additional overhead. Specifically, we are



**Fig. 1.** Co-expression map of 23k genes over 24 cell types of the B-cell family exemplifies map concept. The top view illustrates how maps are combined with client-side graphics: the map is at the center of the display while selecting genes by drawing an enclosing rectangle generates a heatmap on the right. Maps have multiple levels of zooming (bottom 2 rows), each with a potentially different representation. For example, genes are drawn as heatmap glyphs at the high zoom (lower right), and as dots at low zoom. Expression profiles of collocated genes are aggregated and displayed as yellow glyphs over the map. As zoom increases, expression profiles are computed for increasingly smaller regions. Interactions are not limited to zooming and panning; pop-up boxes link out to extra data sources, and selections of genes bring up a heat map (top panel).

involved in the Immgen project [2], a collaborative effort aimed at generating a complete microarray dissection of gene expression in the immunological system of the mouse. The data-map concept lets us disseminate the project's microarray data as precomputed visualizations published on the project website.

The key differences between traditional approaches and our maps are as follows. Instead of the data-query-specification/recompute paradigm, co-regulation maps contain all accessible data, with data query and specification being done through zooming and panning during visualization. Traditionally, it is the end user's job to construct a visualization (query specification and parameter definition), while our visualizations are built by bioinformatics staff in larger labs. Finally, the goal of complex visualization systems is to give users complex functionality that answers a large array of questions. Co-regulation maps, on the other hand, aim to provide fast intuitive access to visual data; their functionality is therefore balanced with a sparse set of interactions, close to what is available in regular Google Maps. For users, including scientists browsing and analyzing data as well as those producing data, visualizations become easy to access, learning time is significantly reduced, users worry only about the data, and disseminating visual results is simplified.

**Contributions.** We introduce co-regulation maps served through the Google Maps API for visually disseminating large microarray datasets. We present design elements, challenges and opportunities that became apparent in our project, an evaluation demonstrating the usefulness of the approach and a set of design guidelines.

## 2 Related Work

Many advanced systems for biological data analysis have been developed over the past decade. Examples targeting microarray expression data include free software packages such as Clusterview [3], TimeSearcher [4], and Hierarchical Clustering Explorer (HCE) [5] or commercial systems such as Spotfire [6] and GeneSpring [7]. GenePattern [8] is a broad effort aiming to facilitate the integration of heterogeneous modules and data into a unitary, web-managed framework for microarray data analysis. Our goal is to offer no-overhead visualizations that will be used primarily for casual data exploration by users unable to spend time learning advanced systems. In that regard, our work comes closer to applications providing primarily look-up functionality such as tools provided on the NCBI website or the genome browser at USCS [9]. In contrast to these efforts, we aim to provide visualizations that include more computation and visual cues and less complicated query specifications.

In terms of web-accessible visualization ManyEyes [10, 11] paved the way for everyday data visualization and demonstrates the usefulness of the web as a dissemination and collaboration medium. Unfortunately, while web-development toolkits such as Protovis [12] greatly aid web development, large scale web visualization is hampered by inherent browser computing and rendering limitations [13]. Alternatively, stand-alone systems have been made available as applets

or to be run as client applications directly from websites [8, 14]. However, users still have to control the parameters involved in producing visualizations, specify their data queries and learn system features. This often constitutes an undesired overhead. Yet another approach, more similar to our work from an implementation standpoint, is to use Ajax (asynchronous JavaScript and XML) technology to do the rendering on the server side and serve images asynchronously to the client browser. A specific call for Ajax-based application in bioinformatics is made in [15] and [16] and [17] exemplify this approach. There is however only one essential element that differentiates this approach from traditional offline visualization systems: control and display happens in a separate place from rendering and computation. Our research differs by attempting to limit regular users' effort in creating visualizations and assigning this task to experienced personnel, by introducing visualizations that contain most of the data associated with a problem, and by using the Google Maps API, a readily available Ajax implementation of pre-rendered images. Closest to our work are X:MAP [18] and Genome Projector [19] which present implementations of genome browsers using the Google Maps API. We extend this idea to 2D embeddings and provide an evaluation that suggests design guidelines.

In our work we use multidimensional scaling (MDS), the process by which multi-dimensional data points are projected in a space with lower dimensionality. We use MDS to represent gene expression similarity over multiple biological conditions. Keim [20] provides a good overview of multidimensional visualization. Non-linear MDS methods, as in our work, use the similarity distance between data points to define an error measure that quantifies the amount of distance information lost during the embedding. Gradient descent or force simulation is then used to position the points in the low-dimensional space so as to minimize the error measure. A good example of such an approach is force directed placement (FDP) [21] which simulates a system of masses connected by springs of lengths equal to the distances to be embedded. Because an iteration of the original FDP model is  $O(n^3)$ , acceleration techniques have been proposed [22–24]. We use the last approach, an algorithm with linear iteration time proposed by Chalmers. Finally, relevant to our work is HiPP [25], an algorithm using a hierarchical clustering to drive a 2D embedding. In our work we use a combination of the original FDP, Chalmers' acceleration technique and HiPP.

### 3 Methods

Given genes with expression measurements over multiple biological conditions, we construct a 2D map where genes are placed so that their proximity is proportional to the similarity of their expression profiles. Scientists can use the B-cell co-regulation map in Figure 1 to find other genes that co-regulate with genes of interest and to understand how their genes of interest co-regulate given the set of conditions described by the map. Immunologists can browse co-regulation maps to understand expression patterns in the featured conditions. Finally, scientists interested in downloading unfamiliar data can perform a preliminary investigation using maps hosted on the project website.

Our embedding algorithm was inspired by HiPP [25] but employs a different layout technique. As in HiPP, we use bisecting k-means to create a hierarchical clustering of the data. We then compute the *clustering distance* of two genes as the length of the path between their corresponding nodes in the clustering tree. We multiply this *clustering distance* by the Euclidian distance between genes in the high-dimensional space described by the biological conditions. Finally, we use Chalmer's embedding [24] to project this combined distance in 2D. The discrete component introduced by the clustering tree is responsible for the clear demarcations between clusters observable in Figure 1. We initially used a standard projection of Euclidian distance alone but user feedback indicated that the lack of visible clusters detracted from analysis. Users considered the modified version preferable even when made aware that cluster boundaries were introduced artificially.

During rendering, glyphs are drawn over map regions, showing the aggregated expression profile of genes in that particular region along with the standard deviation. The size of aggregated regions is zoom-dependent: as zoom level increases averaging is performed over smaller clusters of genes. This is achieved by first clustering 2D gene-points, using a simple minimum-linkage condition, over a discrete range of clustering-thresholds. The clustering-threshold range was generated using the following formula:  $2^i * d$  with  $d$  a constant distance (small with respect to the whole projection) and  $i$  an integer such that  $0 <= i <= \text{maxZoom}$ . The zoom-level is discretized and used to index one of the clusterings obtained in the previous step. Finally, for each clusters an iso-contour is drawn around the members of the cluster by using the method in [26] to achieve a rough enclosing curve and then refining it using active contours [27].

In low-level zooms, genes are represented as heatmap glyphs with color-coded expression values of that gene at each condition, giving users access to individual data values. The chosen color scheme was blue-green-yellow-red to maximize perceived expression differences, following our users' request. To ensure that gene-glyphs are not overlapping, we apply a repulsive force between nodes at the end of the embedding stage. The force decays exponentially with inter-node distance such as to only affect the layout locally.

We use the Google Maps API, an Ajax framework used to render large maps, to display our visualizations. It receives as input image data in the form of a set of small images, called tiles, that when assembled together form the different zoom levels of the map. Each zoom level consists of a rectangular grid of tiles of size  $2^{zoom} X 2^{zoom}$ . The API decodes the zoom level and coordinates of the currently viewed map region to retrieve and display the visible tiles. The developer can load a custom set of tiles in the API by implementing a callback function that translates numerical tile coordinates and zoom level into unique paths to the custom tiles. The API provides basic functionality such as zooming and panning and allows programmatic extension or customization with markers and polyline overlays, information pop-ups and event management. The API can be easily integrated into any Javascript-powered web-page.

Our 2D embeddings are rendered to tiles, gene positions are exported to a text file, and gene expressions are coded as one-byte values to limit size and exported to a text file. These elements are used in the Javascript + Google Maps + Protopis map implementation in Figure 1. Users can search for a single gene and highlight it via a marker. Alternatively, an entire set of genes can be loaded as well through copy and paste. Genes can also be selected directly on the map by drawing a selection rectangle. If the selection is small enough (100 genes ensures interactive performance), a heatmap representation is rendered using the Protopis library. The list of selected genes can be exported for further analysis.

## 4 Results

We conducted an anecdotal evaluation of our co-regulation visualization with the help of the Immgen project coordinator and four geneticists working on regulation patterns in T-cells, B-cells and NK cells. The four geneticists were selected so that their computer operating abilities spanned a broad range, from active involvement in the lab's bioinformatics efforts to limited familiarity with analysis software. As part of the evaluation we introduced the approach and explained its limitations, then demonstrated our prototype while asking questions, and invited users to comment. Two subjects interacted with the prototypes themselves.

All users decided that the co-expression map is useful. The primary workflow that our users identified was to project their own genes of interest onto one or more cell spaces. One subject would also look for global patterns of co-regulation, possibly over multiple maps and suggested we link maps in separate browser tabs, such that selections of genes performed on one map are mirrored onto the others. One subject suggested using this application to create customized datasets by selecting subsets of co-regulated genes from explored datasets and exporting them in a convenient tabular form.

All users rated ease of use as higher than other systems they have worked or experimented with. They were excited to be able to access visualizations in a browser and several stated that this makes them more likely to use the visualizations. One of our subjects thought data maps could be useful for researchers new to a lab since they could start analyzing data right away. She then extended this idea to non-Immgen members and mentioned she would like such visualizations to be present in other data sources as well. She added that Immgen offered good technical support, but that being close to graduating and considering doing independent research, this approach seems appealing.

Most subjects said the available features are enough for quick data analysis. Two users explicitly complimented the superposed expression profiles, stating that they summarize data well and can guide exploration. All users were happy with the heatmap upon-selection mechanism, with the ability to export selected sets of genes and highlight personal genes of interest. Several users asked for more hyperlinking and metadata features.

A majority of our subjects identified the static nature of the maps as a non-issue. Two of them expressed the desire to customize the cell types over which

genes are projected. However, they agreed that there are relatively few cell subsets that they would choose from and that multiple maps covering these possibilities would probably work. We note that these two users were the ones most comfortable using analysis software in their daily research and were highly familiar with Immgen data, explaining the desire for increased flexibility. The Immgen coordinator commented about the benefits of being able to accompany raw data with relevant visualizations and the minimal overhead in deploying and maintaining the map system by simply copying a directory structure. He has since decided to switch the lab's database-driven distribution system to a map oriented one.

## 5 Discussion

**Design:** Instead of the traditional visualization flow of data specification followed by visualization recomputation, our co-regulation maps suggest a different approach: all data is shown at once and data specification/abstraction is done at the time of visualization through zooming and panning. Zooming can be used to summarize data at different abstraction levels, such that relevant information is available to the user at all zoom levels. Since the visual information conveyed is itself spatial, co-regulation visualization is suited for this approach.

Static maps can be synergistically coupled with interactive web elements implemented in Protopis. Focus+context visualizations can be created so that maps offer the context while focus views are implemented in Protopis. We note that we advocate for simplicity: merely replicating the complexity of stand-alone systems on the web was not our goal.

**Uses:** Co-regulation maps are not intended to compete with advanced microarray analysis systems. While biologists sometimes work intensely on specific



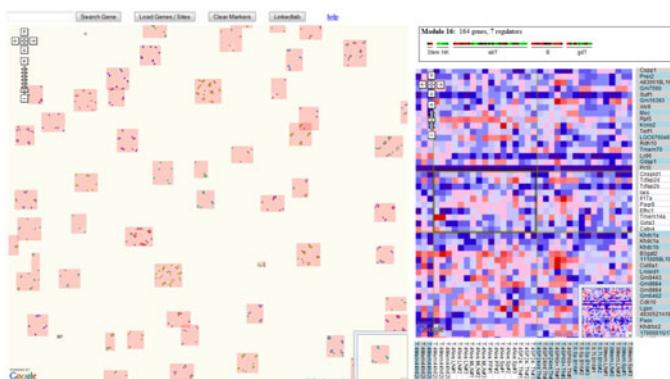
**Fig. 2.** Linked co-regulation maps of the T-cell (left) and B-cell (right) families. A selection in the T-cell map is reflected onto the B-cell map. A few groups of genes that are co-regulated in both cell families are noticeable by inspecting the upper part of the B-cell map.

data-sets for which the flexibility of an advanced analysis system is warranted, they often need to browse through related data-sets where the relevance cannot be clearly assessed. Running a time-consuming visualization on each piece of such data is an overhead which our maps eliminate. As our evaluation indicated, this might be especially relevant for researchers lacking access to a strong computational infrastructure. Similarly, users often want to relate their own data to existent data volumes, a task made easy by loading genes of interest on existent co-regulation maps. Finally, data intensive projects want to distribute readily available visualizations along with raw data so that their users can gain insight into the data without having to run their own analysis. The fact that our collaborator, a coordinator of a data intensive lab, has decided to replace his database-centric data distribution with a map setup supports this claim.

**Opportunities:** Linking multiple co-regulation maps together (e.g. for different cell families) could answer questions about conservation of gene function over multiple conditions, a question raised by one of our subjects. In Figure 2 we show an example of a preliminary implementation of this function using cookies to pass information between multiple browser tabs. This functionality was not evaluated yet.

During our evaluation, users were excited about the opportunities of collaboration offered by maps. Exchanging interactive images rather than static ones and sending links rather than datasets was positively received. Maps also support more integrated collaborative work, such as annotations, well since the static nature of maps ensures that each user has the same view of the data and that shared comments target the same visualization elements.

Finally, co-regulation maps can be extended to display other gene relationships, while the map concept can be applied to other visualization types, such as heat-maps. For example, following a computational analysis of Immgen microarray data, a collaborator separated genes into functional modules and



**Fig. 3.** Google map of gene modules and submodules embedded in 2D on the left. A module was selected by clicking on its enclosing rectangle prompting the display of information about the selected module, including a browsable heatmap, on the right.

submodules with associated regulators. Figure 3 shows a prototype representation of the module space. Instead of the hierarchical clustering from section 3, we use the two-level module/submodule hierarchy to draw genes belonging to the same submodules and modules closer together, with enclosing rectangles drawn over the modules subsequently. Information about a selected module is shown on the right together with a complete expression heatmap of the module’s genes and regulators. The analysis was performed on 346 cell types making dynamically generated heatmaps slow to render using client graphics. However, since genes in a module are predefined, heatmaps can be computed as browsable Google Maps themselves. Protovis implemented axes that are linked to the heatmap’s panning and zooming and thus stick with the map, are attached on the sides ensuring that users know what genes and cell types they are focused on.

## 6 Conclusion

We presented a low-overhead approach for browsing through large, unfamiliar micro-array datasets. We construct pre-computed planar embeddings of genes’ expression values over multiple conditions such as cell types. We then render them as static images and display them using the Google Maps API along with an intuitive set of interactions. The contributions of this work include design elements, uses and opportunities for this type of visualization, and an evaluation that indicates that such visualizations are desirable for exploring novel data, casual browsing, disseminating results and data, and relating small data-sets to existent data volumes.

## References

1. VRL, B.: (Website), <http://graphics.cs.brown.edu/research/sciviz/coexpressionmaps/>
2. Immgen: (Website), <http://www.immgen.org/>
3. Eisen, M., Spellman, P., Brown, P., Botstein, D.: Cluster analysis and display of genome-wide expression patterns. *Proceedings of the National Academy of Sciences* 95, 14863 (1998)
4. Hochheiser, H., Baehrecke, E., Mount, S., Shneiderman, B.: Dynamic querying for pattern identification in microarray and genomic data. In: *Proceedings of IEEE International Conference on Multimedia and Expo*. Citeseer, vol. 3, pp. 453–456 (2003)
5. Seo, J., Shneiderman, B.: Interactively exploring hierarchical clustering results. *Computer*, 80–86 (2002)
6. (Decision site for functional genomics), <http://www.Spotfire.com>
7. (Cutting-edge tools for expression analysis), [www.silicongenetics.com](http://www.silicongenetics.com)
8. Kuehn, H., Liberzon, A., Reich, M., Mesirov, J.: Using GenePattern for gene expression analysis. *Current protocols in bioinformatics/editorial board*, Baxevanis, A.D... [et al.] (2008)
9. Kent, W., Sugnet, C., Furey, T., Roskin, K., Pringle, T., Zahler, A., et al.: The human genome browser at UCSC. *Genome Research* 12, 996 (2002)

10. Viegas, F., Wattenberg, M., Van Ham, F., Kriss, J., McKeon, M.: Manyeyes: a site for visualization at internet scale. *IEEE Transactions on Visualization and Computer Graphics* 13, 1121 (2007)
11. Viégas, F., Wattenberg, M., McKeon, M., Van Ham, F., Kriss, J.: Harry potter and the meat-filled freezer: A case study of spontaneous usage of visualization tools. In: Proc. HICSS (2008)
12. Bostock, M., Heer, J.: Protovis: A Graphical Toolkit for Visualization. *IEEE Transactions on Visualization and Computer Graphics* 15, 1121–1128 (2009)
13. Johnson, D., Jankun-Kelly, T.: A scalability study of web-native information visualization. In: Proceedings of graphics Interface 2008, Canadian Information Processing Society Toronto, Ont., Canada, Canada, pp. 163–168 (2008)
14. Shannon, P., Markiel, A., Ozier, O., Baliga, N., Wang, J., Ramage, D., Amin, N., Schwikowski, B., Ideker, T.: Cytoscape: a software environment for integrated models of biomolecular interaction networks. *Genome Research* 13, 2498 (2003)
15. Aravindhan, G., Kumar, G., Kumar, R., Subha, K.: (AJAX Interface: A Breakthrough in Bioinformatics Web Applications)
16. Berger, S., Iyengar, R., Ma'ayan, A.: AVIS: AJAX viewer of interactive signaling networks. *Bioinformatics* 23, 2803 (2007)
17. Gretarsson, B., Bostandjiev, S., O'Donovan, J., Höllerer, T.: (WiGis: A Framework for Scalable Web-based Interactive Graph Visualizations)
18. Yates, T., Okoniewski, M., Miller, C.: X: Map: annotation and visualization of genome structure for Affymetrix exon array analysis. *Nucleic Acids Research* 36, 780 (2008)
19. Arakawa, K., Tamaki, S., Kono, N., Kido, N., Ikegami, K., Ogawa, R., Tomita, M.: Genome Projector: zoomable genome map with multiple views. *BMC Bioinformatics* 10, 31 (2009)
20. Keim, D.: Information visualization and visual data mining. *IEEE Transactions on Visualization and Computer Graphics* 8, 1–8 (2002)
21. Fruchterman, T., Reingold, E., Dept. of Computer Science, University of Illinois at Urbana-Champaign: Graph drawing by force-directed placement. *Software: Practice and Experience* 21, 1129–1164 (1991)
22. Tejada, E., Minghim, R., Nonato, L.: On improved projection techniques to support visual exploration of multi-dimensional data sets. *Information Visualization* 2, 218–231 (2003)
23. Morrison, A., Chalmers, M.: A pivot-based routine for improved parent-finding in hybrid MDS. *Information Visualization* 3, 109–122 (2004)
24. Chalmers, M.: A linear iteration time layout algorithm for visualising high-dimensional data. In: Proceedings of the 7th Conference on Visualization 1996. IEEE Computer Society Press, Los Alamitos (1996)
25. Paulovich, F., Minghim, R.: HiPP: A novel hierarchical point placement strategy and its application to the exploration of document collections. *IEEE Transactions on Visualization and Computer Graphics* 14, 1229–1236 (2008)
26. Watanabe, N., Washida, M., Igarashi, T.: Bubble clusters: an interface for manipulating spatial aggregation of graphical objects. In: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, p. 182. ACM, New York (2007)
27. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. *International Journal of Computer Vision* 1, 321–331 (1988)

# A New Approach for Lighting Effect Rendering

Catherine Sauvaget and Vincent Boyer

L.I.A.S.D University Paris 8

2 rue de la Liberté

93526 Saint-Denis France

{cath,boyer}@ai.univ-paris8.fr

**Abstract.** We propose a new approach to render different lighting effects on an image. Artists often use different stylizations for different lighting effects in the same image. However, existing work just try to extract shadows without distinction of their type. We introduce a lighting map to describe the different lighting effects and propose six non-photorealistic rendering based on artistic styles ranging from Chiaroscuro to comics. Giving an input image, we are able to automatically generate a lighting map which may be modified by the user to specify the types of shadow or light effects. Our model is flexible and specifically designed to help users and even amateur users, to semi-automatically stylize the different kinds of light effects in an image. It is designed to be integrated into an image editing tool.

## 1 Introduction

Artists have always used the lighting effects they perceive to create their illustrations. The depiction of the lighting conveys a special mood to the scene that creates psychological effects. For example, a front light prevents depth perception or, as is often the case, oblique light at 45°, produces an impression of depth and “ideal” volume. A low angle light creates unreal volumes and is often used for special effects [1] (see Figure [1]). These effects have been widely used in different styles ranging from Chiaroscuro to comics.

In computer graphics, actual researches only focus on general light or shadow detection despite many different types of lights or shadows exist. Some researches and tools only detect shadow in a scene or in images. Scherzer et al. [2] propose to calculate a physically exact map on a 3D scene. They sample the light source over different frames creating a shadow map for each one. A combination of temporal coherence and a spacial filtering is used to correct and speedup the final map creation. Other work propose to detect shadow in videos or image sequences [3], [4], [5]. Some work propose to detect shadow from a single image [6]. It is based on retinex theory (enhancement and illumination compensation of the lightness). They also propose to remove shadow from the input image. Note that few studies are based on one image and that such detections are often computationally heavy and do not permit to distinguish particular effects produced by lighting.



**Fig. 1.** Front light; oblique light and low angle light from [1]

After detecting shadow, one can stylize it. As mentioned by Stork [7], the knowledge of art historical problems is essential to create tools that can reproduce and extend traditional art. Most of existing tools or research for stylization are 3D-based or color-based. Existing 3D tools propose to vary the tone texture following the depth of the scene that supports level of abstraction, perspective or depth of field in toon shading [8]. Other work describes image-based methods to display soft shadows from 3D models [9]. They ensure that soft shadows are well-suited to image-based rendering techniques. Praun et al. [10] propose a system that creates hatching strokes in 3D scenes. This kind of drawing conveys lighting and properties of the material. There are parts in light like reflection which are important in images. A new approach for rendering and animating stylized highlights for 3D models has been proposed by Anjyo et al [11]. It is based on the halfway vectors. For now, this approach is not real time. Most of these papers only focus on one effect and do not propose multiple effects. Moreover, their visual interaction in the target image is never considered.

We propose an artistic based approach to stylize lighting in 2D images. Based on artistic movements and techniques, we introduce artistic shadow and lighting effects depicted by artists and six stylizations. We present our representation for the different lighting effects. Then, following the major artistic movements we detail our different stylization methods. Our results are given and commented.

## 2 Artistic Movements and Ligthing

Since graphical art tries to represent a scene according to relative object position, lighting has always been an issue. Photographs, stage directors, illustrators, painters use lighting to render the desired atmosphere on the scene. Furthermore, they enhance these effects giving them different stylizations such as the lighting effects represented in Chiaroscuro where light parts and dark parts are adjacent. This stylization increases the dramatic tension and creates the illusion of depth. We can cite Rembrandt, Georges de la Tour or le Greco as masters of this art. After this, impressionists played with reflection and transparency. They thought that color is light: “the principal character of a painting is light” (Edouard



**Fig. 2.** Natural light and artificial light from [II]

Manet). From these researches, one can deduce some specific aspects. There are different kinds of lighting effects. The appearance of the shadow depends on the light source. Two kinds of lights exist: natural lights, produced by the sun or the moon and artificial lights produced by punctual lights or fires. The direction and the appearance of the produced shadows depend on it. A light can be direct or diffuse. A directed light produces more violent contrasts (hard shadows) than a diffuse light which produces smoother transitions (soft shadows). That is called the “quality” of the light and the aspect of shadows also depend on it (see Figure 2). Certain surfaces allow the light reflection and, sometimes, dazzling effects can be created. For example, water and glasses often produce such effects. Different comics stylizations use these effects and propose also shadow stylizations with hatching, complementary colors and black flat areas.

It is easy to get the light source position in a 3D scene. The object characteristics are known and it is simple to apply different stylizations to the different parts of an object or to different kinds of shadows. The task becomes more difficult in 2D images because these information are no more available.

### 3 Lighting Map Description and Creation

We choose to represent the lighting effects with a map hereafter called **SL** map. The **SL** map is the same size as the input image and each lighting effect is represented by a color. These effects are hard and soft shadows that can be shades (unlit part of an object) or drop shadows and light that can be illuminated parts of the scene or dazzling effects.

We propose to create the lighting map, in two steps: detecting the shadow and refining the map to precise the different lighting effects. The first step is just a help for the user before refining the produced lighting map. As previously explained, we distinguish different sorts of shadows or lights. Automatically detecting shadow in any image is a very difficult task, particularly if we desire to obtain different variations.

Our main contribution in this paper is to stylize images getting the same information as the artists. Therefore, we consider that the representation of the **SL** map is the most important. For that reason, we propose to the user to refine the **SL** map to add the previous distinctions.

In the following, we present first a fast and easy method to produce a SL map. Then, we explain how we depict and refine the map to produce the defined sorts of shadows and lights.

### 3.1 Detecting Shadow

The first step of our method consists in a shadow detection of the input image. We choose to use the L1 norm which is well designed for shadow detection [12]. It is composed by three components: hue, saturation and lightness. A shadow is a decrease of the light intensity. The color (hue) and the saturation remain unchanged. Thus, we use lightness value to detect shadow. We propose to automatically detect a lightness threshold. We detect the maximum and the median values of lightness in the image. We calculate the difference between the maximum (max) and the median (med) values and divide it by  $\mathcal{T}$  which is the number of intensity levels on 8 bits. We call  $\mathcal{T}$  this threshold calculation. The global threshold  $\mathcal{G}$  is computed as the product of the previous threshold  $\mathcal{T}$  with the median value. For example, if max=247 and med=44,  $\mathcal{T}$  is equal to  $((247 - 44)/255) = 0.796$ .  $\mathcal{G} = \mathcal{T} \times 44 = 35.024$ . With this method, only a part of the pixels, lower than the threshold of lightness, is considered. These pixels are considered as shadow without distinction. The other pixels are considered as illuminated parts.

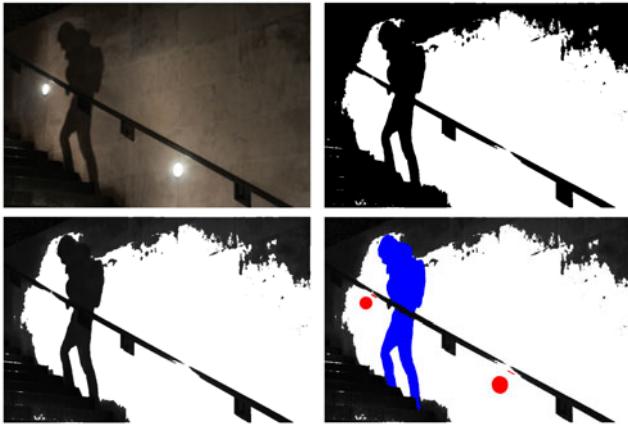
To visually produce the distinction between hard and soft shadows, we propose two methods to depict the shadows. The first method consists in a binary SL map where black areas are shadows and white areas are the lit parts. The second method consists in preserving the value of the shadow lightness of the original image in the SL map, thus obtaining shadings in the SL map.

The different maps are shown with the first three images of Figure 3. The results obtained do not allow us to distinguish shades from a drop shadows, or dazzling effects from other illuminated parts. Thus, it is not sufficient for a detailed stylization. The next subsection describes what we must add in this map to obtain artist-like stylization.

### 3.2 Refining the Map

Our SL map is already black and white or grey-level colored. We propose to make a distinction between shades and drop shadows. We keep the desaturated colors for the shades. The user may colorize and change the color of the drop shadows in blue (see right picture of Figure 3).

Using our detection, we cannot distinguish black areas from shadows. All these areas are considered as shadows in the SL map. For example, if one of the character is black-haired, the detection process associates the black color with shadow. So the user can remove these black areas which are not shadows and replace them by white color. White parts of our resulting map represent the illuminated part of the scene in the picture. The user may precise in red the dazzling effects (see right image of Figure 3).



**Fig. 3.** Original photograph; black and white SL map; shadows with shadings SL map and representation of a drop shadow on the SL map in blue and dazzling effects in red

## 4 Applying Stylization

Following artistic studies presented in section 2, we present our different methods to stylize an image. Input images can be photographs, drawings or paintings. We propose six different stylizations: Chiaroscuro, Impressionism, complementary color, hatching, black flat areas and dazzling effects. For each stylization, the user chooses to apply the effect on a specific component (“color”) of the SL map. We present each style and then explain our algorithm. Our results are presented section 5.

1. **Chiaroscuro:** Chiaroscuro technique was first introduced by Rembrandt to attract the viewer’s eye on a specific part of the painting. It consists in creating a strong contrast between light and shadow using bright and dark adjacent regions (see Figure 4).

To realize this effect on the image, we modify the light value of pixels which correspond to shadows in the SL map. We use the HSL model and for each pixel  $P(P_h, P_l, P_s)$  with  $P_h, P_l$  and  $P_s \in [0; 1]$ , the light value is computed as following:  $P_l = P_l - T / (P_l \times I)$  where  $T$  is the threshold (see section 3.1) and  $I$  the number of intensity levels. Since  $T$  is in  $[0; 1]$ , the light value decreases according to the number of intensity levels chosen by the user. Moreover, as  $T$  depends on the difference between the maximal and the median light values, this computation preserves illumination relations in the image (a dark input image produces a darker target image).

2. **Impressionism:** Impressionism is a technique from the 19th century. The impressionists emphasize the accurate depiction of the previously defined “quality” of light [13]. They totally avoid very dark areas from their paintings. Illuminated objects are often represented with pastel colors and shades or shadows with saturated colors and smooth diminution of light (see Figure 5).



**Fig. 4.** The new-born by Georges de la Tour and The Nightwatch by Rembrandt

To represent this style, we have to create pastel colors and smooth variations of light value:  $P_s = P_s + S \times T / (P_s \times I)$  and  $P_l = P_l - S \times T / (P_l \times I)$  with  $S = 1$  when  $P_s$  is in a shadow area of the SL map (resp.  $S = -1$  when  $P_s$  is in a lit area). As previously mentioned, we do not change the global illumination of the scene and preserve it, applying the same proportion for lit parts and shadow parts.



**Fig. 5.** Irises by Monet; The ballet by Degas; Moret, view of Loing, may afternoon by Sisley and The Yole by Renoir

3. **Complementary color:** Some comics creators like to play with shadows and complementary colors to obtain harmonious images. If we represent colors on a hue wheel, a complementary color is the opposed color on the chromatic hue wheel [14]. For example, yellow and violet are complementary colors (see Figure 6, left picture).

Thus, we propose to leave unchanged the saturation and lightness of shadows but to change the actual hue by its opposite.

4. **Hatching:** Hatching is often used to represent shadow in comics (see Figure 6, 2nd picture) or engraving style [15].

To give a hatching aspect and particularly to the shadow part, we first propose to increase the contrast between pixels using Pratt filter. Then we propose to apply halftoning with short colored lines using Floyd-Steinberg algorithm [16]. The Pratt filter permits to obtain more contrasted lines and therefore to obtain more dark lines. As contrast has been increased in the image, some pixels are lighter and others darker. Then, we compute our shadow detection algorithm that produces hatched shadows. Thus, only some lines are detected as shadow.



**Fig. 6.** Einstein's life 1. Childhood by Daniel Goossens; Blueberry by Moebius; America's Best Comics (1947) and Officer Down by Joe Casey and Chris Burnham.

5. **Black flat areas:** Black flat areas are very common in american comics to represent shadow [17] (see Figure 6, 3rd picture).

We simply propose to copy the black pixels of our SL map in the image.

6. **Dazzling effects:** Dazzling effects can be represented in various ways. It mainly depends on the object material and the light intensity. One can just represent it as a white area or add edges around it to enhance the contrast [15] (see Figure 6, 4th picture).

We propose to compute an edge detection on the image using Meer et al. method [18]. Only edges corresponding to red areas in the SL map are added to the image to enhance the dazzling effects. If necessary, these edges can be stylized or blurred to preserve the global stylization of the image.

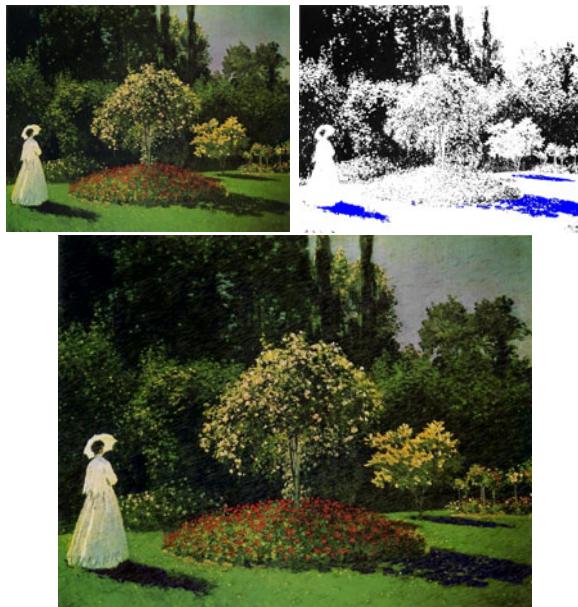
## 5 Results

This section presents the results produced using our SL map model. It allows the user to create different lighting effects on the same image following artist-like stylizations. This model is well adapted to general users and amateur users. All of these images have been produced in real-time on a Pentium 2.5GHz with 3Go of memory.

Figure 7 presents our result on a comics. Original highlights are not bounded by edges. Using the previously defined edge detection and our red areas for dazzling effects in the SL map, we add edges around the highlights. This is visible on the left of the armchair. A complementary color has been produced on the drop shadows represented by blue areas in the SL map. We can see that the color of the back of the armchair is no more red on drop shadows. We can notice that the user has not refined the black areas which do not correspond to shadows (original contours, hair and shirt). Even if the complementary color is computed on the black part of the SL map, that would not change the result in the black image areas since they are totally desaturated and without lightness. A Chiaroscuro effect has been added to the rest of the shadows, visible on the background wall.

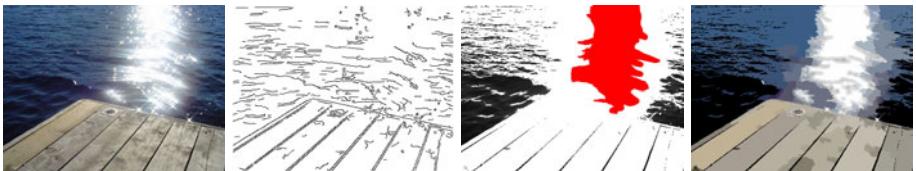


**Fig. 7.** Original image by Lepixx; edge map; shadow map with reflections in red and drop shadows in blue; result with edges for reflections and complementary color on drop shadows and Chiaroscuro on shades



**Fig. 8.** Jeanne-Marguerite Lecadre in the garden by Monet; shadow map with drop shadows in blue; result with Impressionism stylization, complementary color on drop shadows and hatching shades

We also modify a well-known painting of Monet (Figure 8). Using our grey-level SL map with the blue specification for drop shadows, we first apply our Impressionism stylization. Then we produce a complementary color on the drop shadows. The Impressionism stylization permits to enhance colors in the shadow



**Fig. 9.** Original; edge map; shadow map with reflection in red; result with segmentation, reflection and black flat shadows

areas and gives more saturated colors for the complementary stylization. Finally we achieve a hatching for all kinds of shadows.

Figure 9 is a photograph of water with a huge dazzling effect. This effect has been specified in red by the user in the SL map. The result image is a combination of black-flat areas for shadows and some blurred contours in the highlight part. The original photograph has been segmented to obtain a coherent colorization and to create a comics stylization.

## 6 Conclusion

We have proposed a new model for stylization of lighting effects on 2D images. The proposed stylizations are based on artistic studies ranging from Chiaroscuro to comics. While defining our model, we use different existing lighting effects to produce our lighting map. Some of these effects are specified by the user due to semantic and visual distinctions. Our model provides a helpful computer assisted tool for amateur users. This tool is flexible and allows different stylizations on different lighting effects.

In future work, we will improve our model by adding more stylizations and by adding colored light effects. We also plan to consider coupling our approach with a depth map in order to enhance the contrast between the different kinds of lighting.

## References

1. Parramón, J.: Ombres et lumières dans le dessin et la peinture. Bordas (1987)
2. Scherzer, D., Schwärzler, M., Mattausch, O., Wimmer, M.: Real-time soft shadows using temporal coherence. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Kuno, Y., Wang, J., Pajarola, R., Lindstrom, P., Hinkenjann, A., Encarnaçāo, M.L., Silva, C.T., Coming, D. (eds.) ISVC 2009. LNCS, vol. 5876, pp. 13–24. Springer, Heidelberg (2009)
3. Ibrahim, M., Anupama, R.: Scene adaptive shadow detection algorithm. In: Proceedings of World Academy of Science, Engineering and Technology, pp. 1307–6884 (2005)
4. Cavallaro, A., Salvador, E., Ebrahimi, T.: Detecting shadows in images sequences. In: European Conference on Visual Media Production, pp. 165–174 (2004)

5. Cucchiara, R., Grana, C., Piccardi, M., Prati, A.: Detecting moving objects, ghosts, and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 25, 1337–1342 (2003)
6. Sun, J., Du, Y., Tang, Y.: Shadow detection and removal from solo natural image based on retinex theory. In: Xiong, C.-H., Liu, H., Huang, Y., Xiong, Y.L. (eds.) ICIRA 2008. LNCS (LNAI), vol. 5314, pp. 660–668. Springer, Heidelberg (2008)
7. Stork, D.: Computer vision and computer graphics analysis of paintings and drawings: an introduction to the literature. In: Jiang, X., Petkov, N. (eds.) Computer Analysis of Images and Patterns. LNCS, vol. 5702, pp. 9–24. Springer, Heidelberg (2009)
8. Barla, P., Thollot, J., Markosian, L.: X-toon: An extended toon shader. In: International Symposium on Non-Photorealistic Animation and Rendering (NPAR). ACM, New York (2006)
9. Agrawala, M., Ramamoorthi, R., Heirich, A., Moll, L.: Efficient image-based methods for rendering soft shadows (2000)
10. Praun, E., Hoppe, H., Webb, M., Finkelstein, A.: Real-time hatching. In: SIGGRAPH 2001: Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques. ACM, New York (2001)
11. Anjyo, K., Hiramitsu, K.: Stylized highlights for cartoon rendering and animation. *IEEE Comput. Graph. Appl.* 23, 54–61 (2003)
12. Angulo, J., Serra, J.: Traitements des images de couleur en représentation lumi-nance/saturation/teinte par norme l1. In: Traitement du signal, pp. 583–604 (2004)
13. Gärtner, p.J.: Art and architecture, Musée d'Orsay. Könemann (2001)
14. Itten, J.: Kunst der Farbe. Otto Maier Verlag, Ravensburg (1961)
15. Duc, B.: L'art de la BD. Glénat (1983)
16. Strothotte, T., Schlechtweg, S.: Non-photorealistic computer graphics: modeling, rendering, and animation. Morgan Kaufmann Publishers Inc., San Francisco (2002)
17. McCloud, S.: Making Comics: Storytelling Secrets of Comics, Manga and Graphic Novels. Harper Paperbacks (2006)
18. Meer, P., Georgescu, B.: Edge detection with embedded confidence. *IEEE Trans. Pattern Anal. Mach. Intell.* 23, 1351–1365 (2001)

# SemaTime - Timeline Visualization of Time-Dependent Relations and Semantics

Christian Stab, Kawa Nazemi, and Dieter W. Fellner

Fraunhofer Institute for Computer Graphics Research,

Fraunhoferstrasse 5, 64283 Darmstadt, Germany

{christian.stab,kawa.nazemi,d.fellner}@igd.fraunhofer.de

<http://www.igd.fraunhofer.de>

**Abstract.** Timeline based visualizations arrange time-dependent entities along a time-axis and are used in many different domains like digital libraries, criminal investigation and medical information systems to support users in understanding chronological structures. By the use of semantic technologies, the information is categorized in a domain-specific, hierarchical schema and specified by semantic relations. Commonly semantic relations in timeline visualizations are depicted by interconnecting entities with a directed edge. However it is possible that semantic relations change in the course of time. In this paper we introduce a new timeline visualization for time-dependent semantics called SemaTime that offers a hierarchical categorization of time-dependent entities including navigation and filtering features. We also present a novel concept for visualizing time-dependent relations that allows the illustration of time-varying semantic relations and affords an easy understandable visualization of complex, time-dependent interrelations.

## 1 Introduction

Time appears in many different domains as a common attribute. For example the information of medical records [1] is usually tagged with timestamps to arrange the entries in a chronological order. But also in the area of biographical or historical information systems [2], criminal investigation and even in multimedia authoring tools [3] the temporal dimension plays a major role. Timeline based visualizations arrange these time-dependent entities along a time axis whereby the user is significantly supported to get an overview and to comprehend the temporal structure [4].

With the development of semantic technologies and modeling languages such as RDF or OWL also new requirements for information [5] and timeline visualizations arise. The entities in a semantically modeled domain are not only represented as a collection of various resources but also categorized in a hierarchical domain-specific schema and specified by linking them with different kinds of semantic relations [6]. Hence timeline visualizations for time-dependent semantics should be able to visualize beside the temporal dimension the hierarchical schema of the domain and semantic relations between the entities. Currently

there are different timeline visualizations which meet some of these requirements. For example there are timeline visualizations that divide the screen into horizontal slices to categorize the visualized entities in a flat or even in a hierarchical structure [7,8]. Other timeline visualizations are suitable for illustrating semantic relations as directed edges [9,4].

However semantic relations could change in the course of time. For example a person was employed at a company and changed to another after a certain period; the renter of an apartment has changed or a musician has changed to another band. Indeed it is possible to visualize these time-dependent relations with the traditional approach by interconnecting related entities with a directed and time-marked edge but this approach requires high cognitive effort to understand these temporal changes.

For this reason we introduce a novel interactive timeline visualization for time-dependent semantics called *SemaTime* that offers a hierarchical categorization of time-dependent entities including navigation and filtering features. SemaTime also provides different interaction methods like pan+zoom for navigating through the temporal dimension and an overview function for not losing the context of the visualized information. Also we present a novel concept for visualizing time-dependent relations that allows the illustration of time-varying semantic relations and affords an easy understandable visualization of complex, time-dependent interrelations.

The remaining paper is structured as followed: In the next section we examine existing timeline visualizations and in particular their suitability for visualizing time-dependent semantics and discuss their advantages and drawbacks. Afterward we present our novel approach for visualizing time-dependent semantics and give a detailed description of SemaTime. We conclude this paper with an application scenario, a discussion and a conclusion of our work.

## 2 Related Work

Nowadays there are many different approaches for visualizing time-dependent information (e.g. [10,11]). Most of these visualizations are based on timeline visualizations that arrange time-dependent entities along a time-axis. For example Allen [12] uses an interactive timeline for the chronological visualization of the content of a digital library. This timeline visualizes past events and periods. To indicate different categories, the visualized entities can be illustrated with different colors but the visualization of hierachic categories or semantic relations is not supported by this approach. A widely used timeline visualization is *SIMILE* [13], that is used from Alonso et al. [14] to visualize search results. SIMILE provides the presentation of events and periods along a horizontal time axis and offers different navigation features and an overview function. The *Context-Focus-Timeline* [2] visualizes events on a vertical time-axis and is used in a *History Event Browser*. For every event further information can be displayed whereas the timeline offers a clear overview of the temporal context.

However for the visualization of time-dependent semantics the visualization should be able to visualize the domain-specific, hierarchical schema and semantic

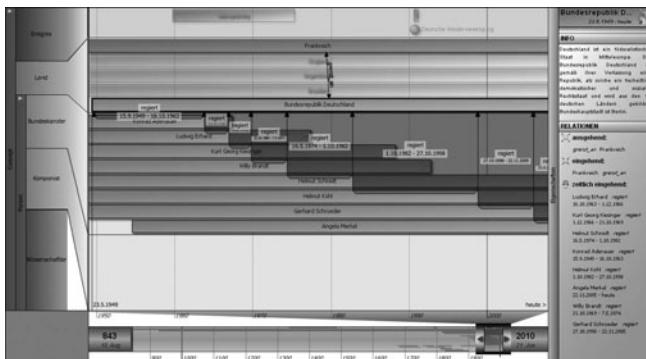
relations between the entities. One approach for visualizing these complex data structures is introduced by Plaisant et al. [8]. *Lifelines* is able to visualize different facets which are depicted as horizontal slices and optionally hierarchical structured. Lifelines even supports the visualization of semantic relations in an implicit way by highlighting related events or periods on demand. However there is no possibility for distinguishing different relation types or to understand the meaning of relations. Bui et al. introduce an interactive timeline visualization for visualizing medical patient records in hospital information systems [1]. This timeline visualization uses a similar approach for visualizing hierarchical categories of the time-dependent information but is not suitable for visualizing semantic relations. Another timeline visualization that supports the pictorial representation of hierarchical structured and time-dependent information is *Timeline Trees* [7] which is designed for visualizing sequences of transactions in information hierarchies. The hierarchical structure of the available data is depicted as horizontally oriented tree visualization. The nodes of the tree can be expanded or collapsed to support information filtering and to navigate through the hierarchical structure.

One of the first approaches for explicitly depicting semantic relations in timeline visualizations is introduced by Kumar et al. [4]. The *tmVIEWER* is able to visualize beside events and periods, semantic relations as directed edges between time-dependent entities. However this approach can only conditionally applied to visualize time-dependent relations. Jensen uses the same approach for depicting semantic relations in the *SemTime* visualization [9]. SemTime also provides stacking of timelines and supports the visualization of flat categories in different layers which can be independently set to different time intervals.

Beside the introduced timeline visualizations there are several other approaches that are especially designed for a particular application or a specific kind of time-dependent information. For example the timeline visualization from Bade et al. [15] is especially designed for visualizing high-dimensional, time-oriented data and is used in the area of intensive care units e.g. for depicting fever curves of patients. André et al. introduce the timeline visualization *Continuum* [16] that comes with user-determined controls over the level of detail, a histogram function and a comparative split view. Furthermore Continuum is able to visualize time-dependent, hierarchically structured entities by nesting. However this approach is only suitable for visualizing hierarchies in which every node is time-dependent and for this reason it is not applicable for visualizing a general, hierarchical categorization.

### 3 SemaTime - Visualization of Time-Dependent Semantics

SemaTime is an interactive timeline visualization especially designed for depicting time-dependent semantics. In this section we first describe the basic design and different parts of SemaTime and subsequently the visualization, navigation and interaction methods that are used for visualizing the temporal structure of the semantic entities, the hierarchical, domain-specific schema and semantic relations. Moreover we present our novel approach for visualizing time-dependent semantic relations.



**Fig. 1.** SemaTime - Timeline Visualization for Semantics

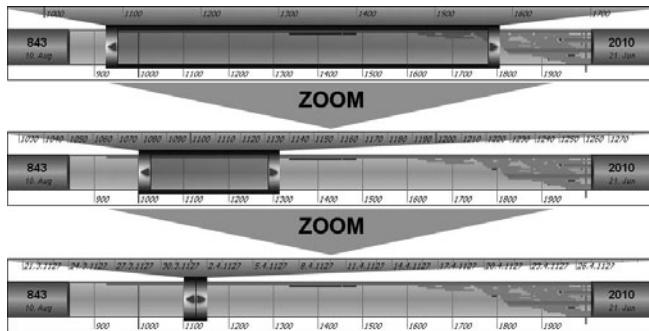
Figure 1 shows a screenshot of SemaTime that is divided in the following four parts:

1. *Temporal Navigation Panel*: Offers an overview for context visualization and different interaction methods for adjusting the time interval.
2. *Main View*: Visualizes time-dependent entities along a time-axis. On the navigation through the temporal dimension the Main View is adjusted accordingly.
3. *Hierarchical Categorization View*: Divides the Main View in horizontal slices and allows the navigation in the domain-specific, hierarchical schema.
4. *Detail View*: Shows detail information (e.g. descriptions or attributes) of a selected element.

In principle SemaTime is analogous designed as a typical coordinate system. On the horizontal axis is the temporal dimension while on the vertical axis the domain-specific, hierarchical schema is depicted. The Hierarchical Categorization View divides the Main View into different horizontal slices each of which corresponds to a concept of the hierarchy. Thus, the membership of a temporal information depicted in the Main View to the appropriate concept is easy to recognize.

### 3.1 Navigation in the Temporal Dimension and Overview

The Temporal Navigation Panel at the bottom of SemaTime is divided into two different time-scales. The upper scale shows the currently selected time interval of the Main View and the lower scale depicts the entire time interval of the given data set. For the navigation in the temporal dimension the lower scale contains a slider that can be used to set the visible time interval of the upper scale and accordingly the selected temporal area of the Main View. Currently there are zoom+pan interaction patterns implemented in SemaTime for navigating in the temporal dimension.



**Fig. 2.** Navigation in the Temporal Dimension

The temporal zoom allows the user to zoom in or out of the currently selected time interval by using the mouse wheel. So it is possible for the user, to adjust the granularity regarding the temporal dimension and to view the information e.g. in centuries, decades or month (Fig. 2). Since the temporal zoom influences both boundaries of the selected time interval, it is also possible to adjust the boundaries individually by dragging the left or right side of the slider on the lower scale. Thus it is easier to select a specific time interval and the navigation in the temporal dimension is alleviated for the users.

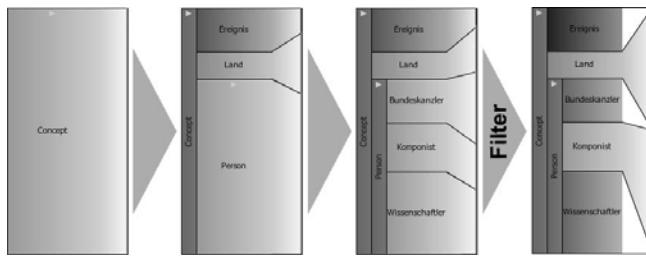
In contrast to the temporal zoom the pan navigation implemented in SemaTime allows the temporal navigation without influencing the granularity. For this type of navigation there are two different interactions available. On the one hand the user is able to pan the selected area by dragging the time slider on the lower scale either to the left or to the right side. The other option is to click and drag directly in the Main View.

Additionally to the navigation functionalities the temporal navigation panel offers an overview. For this purpose all time-dependent entities of the dataset are visualized along the lower scale. So users can easily be aware of entities that are currently not visible in the Main View due to the selected time interval.

### 3.2 Hierarchic Categorization and Filtering in SemaTime

As mentioned before the hierarchical schema of the domain is visualized in the Hierarchical Categorization View as a vertical axis. This component of SemaTime divides the Main View into horizontal slices each of which corresponds to a concept in the hierarchy. By clicking on a node in the hierarchy it is possible to expand or collapse the visualized concepts and it is also possible to filter specific nodes (Fig. 3). So users are able to select specific categories of interest and can easily adjust the visualized information according to their preferences.

The size of the displayed categories is calculated relative to the amount of entities pertaining to a category. For instance the dataset in Figure 3 contains more persons than countries. Due to the layout algorithm of the Main View, that calculates the minimal number of lines and displays every entity with the same



**Fig. 3.** Navigation and Filtering in the Hierarchical Categorization Schema

height, the heights of concepts in the Hierarchical Categorization View and the Main View are not identical. For this reason, a trapezoid view for connecting the Hierarchical Categorization View with the Main View is integrated to ensure ideal space utilization and visual concept assignment in the Main View even if elements are filtered. To increase the user experience with SemaTime, the transitions during navigation or filtering procedures are smoothly animated.

### 3.3 Visualization of Events and Periods

Having introduced the Temporal Navigation Panel and the Hierarchical Categorization View, we introduce the Main View in this section. In the Main View the time-dependent entities are visualized and placed according to their temporal attributes and associated concept. There are two different types of time-dependent information that are distinguished in SemaTime: Events and periods. In contrast to periods, events are characterized by a single time stamp. They correspond to a specific date on which the event occurred. On the other hand, time periods are characterized by a unique starting and end point, whereby a certain time interval is defined. In order to assure a clear visibility in each temporal granularity and a visible differentiation of these two entities, events are visualized as a circle and time periods are depicted as rectangles (Fig. 4). This representation has the advantage that events are still visible at large selected time intervals and are not visualized as thin lines. Figure 4 shows an example of an event (German reunification) and a period (gulf war) depicted in SemaTime.

Since the exact time of an entity is difficult to recognize in cases of large selected time intervals in the Main View, the user is able to display these information on demand. Therefor, the user selects an element by clicking on it whereby further information becomes visible (Fig. 4).



**Fig. 4.** Visualization of Events and Periods in SemaTime

### 3.4 Visualization of Semantic Relations

Semantic relations form the core of semantics. They are used to define connections between the modeled entities in order to describe the information more precisely. Commonly a semantic relation between two entities is characterized by a direction and a type. However in a temporal domain it is also possible that a semantic relation is only valid for a certain time interval or changes in the course of time. Hence in a temporal domain it is also possible that a semantic relation contains temporal information that should be adequately represented. For this reason we developed a new concept for visualizing time-dependent relations and implemented it in SemaTime. Beside the visualization of time-dependent relations (e.g. `lives_in`, `works_at`, etc.), SemaTime is also capable of visualizing semantic relations without time reference (e.g. `father_of`, `brother_of`, etc.) by connecting related entities with a directed and labeled edge (Fig. 5).



**Fig. 5.** Semantic Relation without Time Reference

Time-dependent relations are depicted as a rectangle between related entities which denotes the temporal validity of the relation. The direction of the relation is visualized by arrows pointing to the related entity. The type of the relation and the temporal information are visualized in a label inside of the rectangle. Figure 6 shows an example of a bilateral, time-dependent relation. It is easy to recognize that Marie Curie was married with Pierre until his death in 1906 and vice versa.



**Fig. 6.** Visualization of Time-Dependent Relations in SemaTime

To prevent visual clutter and to reduce the cognitive load of the user, SemaTime does not show semantic relations by default, rather the user is able to request them for an entity of interest on demand. By selecting an entity with a mouse click all entities that are not semantically related to the selected one will be reduced in their alpha values and the relations are shown (Fig. 6). So the

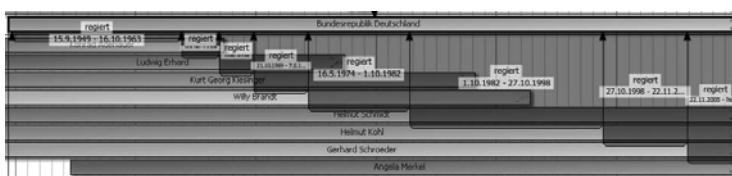
user is supported in perceiving semantic relations without losing the informational context. To denote the existence of relations, an icon is displayed within a related entity. So the user is able to gather an overview of given information and is not forced to search unnecessary.

## 4 Application Scenario

SemaTime was developed as a visualization component of the SemaVis Framework [17], a development of the Core-Technology-Cluster Innovative User Interfaces and Visualizations (CTC) of THESEUS [18]. THESEUS is a 60-month program partially funded by the German Federal Ministry of Economics and Technology.

The SemaVis Framework provides core technologies for visualization, editing and annotation of semantically annotated data. The main goal of SemaVis is the provision of core-technologies for heterogeneous users, data and use cases incl. the THESEUS Use Cases. For this reason SemaVis provides different data models for visual attributes, semantics, interaction events, etc. that are shared by different visualizations. This architecture allows for instance the visualization-independent adaptation of visual appearances. Therefor an interaction analysis system based on probabilistic methods [19] was integrated in SemaVis for analyzing interaction events of individual users. The extracted user information (e.g. preferences, predictions, activities) are used to adapt the shared data models to the individual user. Hence the logic of the user-centered adaptation is decoupled from the visualization itself and each of these SemaVis-compliant visualizations is therefore an *Adaptive User Interface* (AUI) that adapts the visual appearance to user's behavior.

Figure 7 shows an example of SemaTime visualizing a part of the political history of Germany. By using time-dependent relations it is more easier to recognize who, when and even how long a politician was or is Chancellor of the Federal Republic of Germany. Furthermore the appearance of the entities are adjusted to the user's behavior by using the SemaVis adaptation mechanism. In this example entities that have been identified as particularly relevant to the individual user, are presented with a more intense color.



**Fig. 7.** SemaTime visualizing the Chancellors of Germany

## 5 Discussion

SemaTime is a visualization concept for visualizing time-dependent semantics, so that users are able to easily understand the chronological structure of the given data. Our approach also provides the visualization of domain-specific schemata to allow the hierarchical categorization of time-dependent entities. Since temporal relations may change in the course of time, we introduced a novel approach for visualizing these time-varying relations as a rectangle between related entities. So SemaTime is able to visualize complex time-dependent interrelations between different entities. However in some cases, it is possible that time-dependent relations overlap. In such a case, the visualization would be very confusing and incomprehensible due to overlapping relation labels. One possible solution to solve this problem would be the insertion of a focus mechanism that allows scrolling through overlapping relations.

Currently we conduct several evaluations to enhance the usage of SemaTime. Particularly our evaluations are focused on the automatic adaption of SemaTime to the needs of the user. However the first usage tests regarding the user acceptance show great promises for SemaTime and particularly our novel approach for visualizing time-dependent relations.

## 6 Conclusion

In this paper we presented a novel interactive timeline visualization that is especially designed for visualizing time-dependent semantics. SemaTime offers a hierarchical categorization view that divides the screen into horizontal slices and provides the visualization of domain-specific schemata. We also present different interaction methods and an easy to use navigation panel for browsing the temporal dimension including an overview function for context visualization. Our new concept for visualizing time-dependent relations allows SemaTime to visualize time-varying relations and to impart complex time-dependent interrelations in an adequate way.

## References

1. Bui, A.A.T., Aberle, D.R., Kangarloo, H.: Timeline: Visualizing integrated patient records. *IEEE Transactions on Information Technology in Biomedicine* 11, 462–473 (2007)
2. Allen, R.B.: A focus-context browser for multiple timelines. In: *JCDL 2005: Proceedings of the 5th ACM/IEEE-CS Joint Conference on Digital Libraries*, pp. 260–261. ACM, New York (2005)
3. Kurihara, K., Vronay, D., Igarashi, T.: Flexible timeline user interface using constraints. In: *CHI 2005: CHI 2005 Extended Abstracts on Human Factors in Computing Systems*, pp. 1581–1584. ACM, New York (2005)
4. Kumar, V., Furuta, R., Allen, R.B.: Metadata visualization for digital libraries: interactive timeline editing and review. In: *DL 1998: Proceedings of the Third ACM Conference on Digital Libraries*, pp. 126–133. ACM, New York (1998)

5. Katifori, A., Halatsis, C., Lepouras, G., Vassilakis, C., Giannopoulou, E.: Ontology visualization methods—a survey. *ACM Comput. Surv.* 39, 10:1–10:43 (2007)
6. Studer, R., Davies, J., Warren, P.: Semantic Web Technologies: Trends and Research in Ontology-Based Systems. Wiley, Chichester (2006)
7. Burch, M., Beck, F., Diehl, S.: Timeline trees: visualizing sequences of transactions in information hierarchies. In: AVI 2008: Proceedings of the Working Conference on Advanced Visual Interfaces, pp. 75–82. ACM, New York (2008)
8. Plaisant, C., Milash, B., Rose, A., Widoff, S., Shneiderman, B.: Lifelines: visualizing personal histories. In: CHI 1996: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 221–227. ACM, New York (1996)
9. Jensen, M.: Visualizing complex semantic timelines. NewsBlib (2003)
10. Aigner, W., Miksch, S., Müller, W., Schumann, H., Tominski, C.: Visualizing time-oriented data—a systematic view. *Comput. Graph.* 31, 401–409 (2007)
11. Silva, S.F., Catarcı, T.: Visualization of linear time-oriented data: A survey. In: WISE 2000: Proceedings of the First International Conference on Web Information Systems Engineering (WISE 2000), Washington, DC, USA, vol. 1, pp. 310–319. IEEE Computer Society, Los Alamitos (2000)
12. Allen, R.B.: Interactive timelines as information system interfaces. In: Symposium on Digital Libraries, Japan, pp. 175–180 (1995)
13. SIMILE (2010), <http://www.simile-widgets.org/timeline>
14. Alonso, O., Baeza-Yates, R., Gertz, M.: Exploratory search using timelines. In: SIGCHI 2007 Workshop on Exploratory Search and HCI Workshop (2007)
15. Bade, R., Schlechtweg, S., Miksch, S.: Connecting time-oriented data and information to a coherent interactive visualization. In: CHI 2004: Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 105–112. ACM, New York (2004)
16. André, P., Wilson, M.L., Russell, A., Smith, D.A., Owens, A., schraefel, m.c.: Continuum: designing timelines for hierarchies, relationships and scale. In: UIST 2007: Proceedings of the 20th Annual ACM Symposium on User Interface Software and Technology, pp. 101–110. ACM, New York (2007)
17. Nazemi, K., Breyer, M., Burkhardt, D., Stab, C., Hofmann, C., Hornung, C.: D.ctc.5.7.1: Design und conceptualization semantic visualization framework. THESEUS Deliverables: CTC-WP5 Innovative User Interfaces and Visualization 2 (2009)
18. Theseus Program Home Page (2010), <http://theseus-programm.de>
19. Nazemi, K., Stab, C., Fellner, D.W.: Interaction analysis for adaptive user interfaces. In: Proceedings of the Sixth International Conference on Intelligent Computing (ICIC 2010) (2010) (in print)

# Comics Stylizations of 3D Scenes Using GPU

Jordane Suarez, Farès Belhadj, and Vincent Boyer

L.I.A.S.D. Université Paris 8

2 rue de la liberté

93526 Saint-Denis Cedex, France

{suarez,amsi,boyer}@ai.univ-paris8.fr

**Abstract.** We propose a new comics stylization model based on a very efficient depth map generation. It is designed to render large scenes with multiple objects as well as single object in real time through a complete GPU implementation. 3D comics stylizations are generally view-dependent and only use the camera field of view to render the scene. In all cases, the depth of objects is computed according to the near and far planes while they are almost without any relation with the range depth of these objects present in the scene. Our model solves this problem by computing minimal and maximal values in the depth map. Moreover, it reproduces and improves better comics stylizations proposed for 2D images. Results show that our model is suitable for different kinds of 3D scenes and to produce various comics stylizations.

## 1 Introduction

Toon shading is one of the most well-known effect in Non-Photorealistic rendering. It consists in reproducing simple cartoon shading on a 3D scene and is largely used in production software or video games. Extensions and improvements have been proposed to produce different comics styles but two main problems still remain: large scenes with multiple objects are never considered and one of the fundamental effects, the color desaturation according to the object depth can be greatly improved. We propose a comics stylization model to render 3D scenes in real-time using GPU implementations. It is able to solve this two problems. A new formulation to compute depth map is proposed and is well adapted for comics stylizations of both large scene including multiple objects and single object. We also present the implementation of different comics styles. In the next section, we present previous work, our model with the depth map generation and the stylization process and finally we discuss our results.

## 2 Related Work

Several methods have been proposed to create comics stylizations of 3D scenes. The goal of the authors is to obtain real-time methods that can be used for

example in video games. Usually, comics stylizations are only produced using a particular non-physical lighting model. The proposed lighting model and its associated shading are used to render the 3D scene.

The first method have been published by Lake et al. [1] and is available in many software solutions and called toon shading. It uses the Lambertian reflection (computing for each fragment the dot product between its normal and the normalized light direction vector pointing from the surface to the light source) as a 1D toon texture index. This technique is easily understandable, implementable, uses only diffuse reflection term but is view-independent and does not take into account the remoteness of the considered object.

Different approaches have been studied to provide depth in toon shading. Texture-based approaches use mip-mapping [2] [3] or a 2D texture [4]. Unfortunately, when the distance grows between the textured object and the viewer artifacts appear and are more visible using hatching or strokes based rendering.

Mip-maps methods proposed by Klein et al. [2] and Praun et al. [3] provide art-maps to preserve constant-sized strokes or hatching patterns at different depths and levels of lighting. Unfortunately these techniques only focus on patterns used to render the object in the scene. However, the rendering of different objects in the same scene is never studied.

The Xtoon model [4] exploits the Lambert's term on the same principle as the toon shading and uses a 2D texture index by adding a notion of details which vary according to the object depth or its orientation. As its goal is more the visual abstraction than the rendering speed, it introduces the Level Of Abstraction notion instead of classical LOD [5] [6]. Level Of Abstraction can be achieved through tone or shape details. These can be easily realized by a designer using a 2D texture. A GPU implementation is also proposed for real time renderings. But even if this process is view-dependent, it does not transmit the desired effect when the viewpoint is inside a large scene. It can be used for a single object but is not well-adapted for a large scene with multiple objects.

Other 3D methods try to improve toon shading focusing on particular effects. Anjyo et al. [7] have proposed an approach that renders comics-stylized highlights on 3D objects. Starting with an initial highlight design, using the Blinn's specular model, highlight shapes are created through several functions. Nevertheless the comics stylization for a global 3D scene is never studied.

A 2D comics stylization model has been proposed by Sauvaget et al. [8]. This method generates comics images from a single photograph using a depth map to avoid the depth-less color problem and colorizes images with a specified atmosphere. It extracts image structures from the original photograph, generates a depth map from them and finally performs a treatment on a segmented image to give a comics style. Note that the generated depth map can be enhanced by the user and different comics styles have been proposed using depth information. This technique works well with a specific depth map but needs to be adapted to 3D scenes and even more animations.

### 3 Our Model

Our model is designed to render a 3D scene including multiple objects with a view-dependent comics style. Based on the stylization process described by Sauvaget et al. [8], we propose to generate a scene depth map to render it with different comics styles. We apply a stylization shader using this map and the image of the rendered scene. We convert the texture RGB information into HSV. Then, based on the depth map, we desaturate these values and, if needed, we apply an ambiance color. Moreover, we can use an edge detection algorithm as Prewitt or Laplace to finally compose the result image.

However, one of the main problems is the calculation of the appropriate depth map for a comics stylization rendering. This map is used both for the rendering and the edge detection and its computation is a crucial step. As we demonstrate hereafter, a depth map linearly or logarithmically interpolated does not give suitable results. In the following, we present our method to generate a suitable depth map and our rendering process including desaturation, ambiance color and edge stylization.

#### 3.1 Depth Map Generation

Classical computation of the depth map consists in calculating the fragment depth according to the camera field of view (near and far plane).

It is well-known that Z-buffer is non linear:

$$\text{depth} = \frac{\text{far} + \text{near}}{\text{far} - \text{near}} + \frac{-2 \times \text{far} \times \text{near}}{Z \times (\text{far} - \text{near})} \quad (1)$$

A better precision is obtained close to the camera near plane. Moreover, this precision is increased according to the near/far ratio (more this ratio is greater, more the Z values are densely grouped around the near plane). Thus, the Z-buffer is not convenient to computations that need an uncompressed precision along the Z axis. Unfortunately, solutions that try to interpolate the Z-buffer values fall into a depth buffer precision problem. In fact, Z-buffer values are in [0.0 ; 1.0] and have been already discretized. All following proposed solutions use the depth fragment value, hereafter noted  $zfrag$ .

Intuitively, a linear interpolation can be proposed as:

$$\text{depth} = 1.0 - \frac{zfrag - \text{near}}{\text{far} - \text{near}} \quad (2)$$

In that case, the depth depends on the camera field of view (near and far plane) and does not take into account the position of objects in the scene (for example, when near plane is 0.1, far plane is 1000.0 and objects in the scene are in [0.1 ; 100], depth values are around 0.9 and 1.0 for all objects) and the depth map is not suitable for any stylizations (see Figure 2).

Xtoon proposes a logarithmic interpolation :

$$\text{depth} = 1.0 - \frac{\log \frac{z_{\text{frag}}}{z_{\text{near}}}}{\log \frac{z_{\text{far}}}{z_{\text{near}}}} \quad (3)$$

It produces a detailed and suitable depth map only when the objects are close to the near plane. As one can see in Figure 2, the problem still remains when the objects are close to the far plane.

To address this problem, we propose to calculate the depth map according to the minimum and maximum Z values of objects in the camera field of view as follow:

1. We render the scene and store the depth of each fragment into a texture. To preserve the 32-bits depth precision, either we use a 32-bit float-valued intensity texture or each depth is stored in four components (RGBA) texels using shifting and mask operations. At the end of this step, we obtain an initial RGBA texture  $T_0$  in which each texel contains the depth of the fragment. The texture size is  $(W, H)$  where  $W$  and  $H$  could be different but must be  $2^n$  for some integer  $n$ ;
2. Starting with  $T_0(W, H)$ , our shader creates a new texture  $T_1(W, \frac{H}{2})$  in which we store alternatively minimal and maximal values of four texels of  $T_0$  as follow:

$$\forall x \in 2\mathbb{N}, \forall y \in \mathbb{N}, t'(x, y) = \min_{i,j \in \{0,1\}} t(x+i, 2y+j)$$

$$\forall x \in 2\mathbb{N} + 1, \forall y \in \mathbb{N}, t'(x, y) = \max_{i,j \in \{0,1\}} t(x-i, 2y+j)$$

Where  $t'$  is a texel of  $T_1$  and  $t$  a texel of  $T_0$  (see Figure 1).

3. As we obtain a texture  $T_1(W, \frac{H}{2})$  with alternatively minimal and maximal values, we construct a texture  $T_2(\frac{W}{2}, \frac{H}{4})$  in which we store alternatively the minimal and the maximal values of the four previous minimal and maximal values stored in  $T_1$ :

$$\forall x \in 2\mathbb{N}, \forall y \in \mathbb{N}, t''(x, y) = \min_{i,j \in \{0,1\}} t'(2(x+i), 2y+j)$$

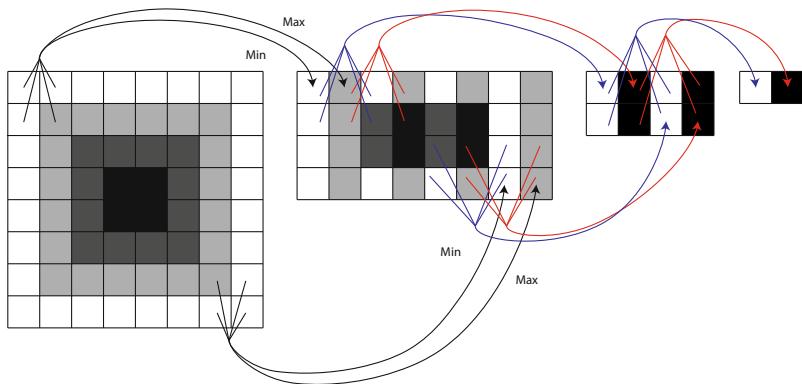
$$\forall x \in 2\mathbb{N} + 1, \forall y \in \mathbb{N}, t''(x, y) = \max_{i,j \in \{0,1\}} t'(2(x+i) - 1, 2y+j)$$

We repeat this process until we obtain a texture of size  $(2, 1)$ .

As shown in Figure 1, minimum and maximum values are finally stored in the first and the second pixels. The complexity of this algorithm is linear.

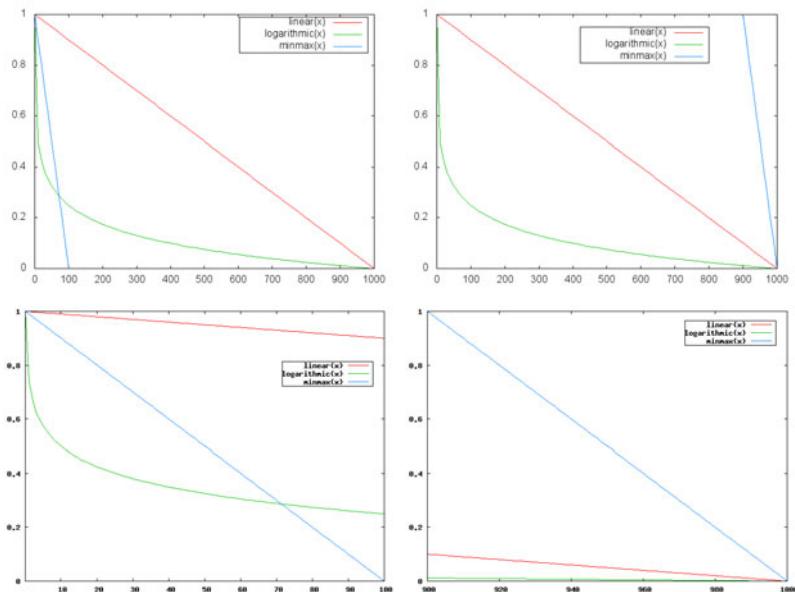
An interpolation based on the minimum and maximum depth such as the following formula, produces a map, hereafter called minmax, covering the entire depth values.

$$\text{depth} = 1.0 - \frac{z_{\text{frag}} - \text{min}}{\text{max} - \text{min}} \quad (4)$$



**Fig. 1.** Minimal and maximal depth computation from left to right:  $T_0$ ,  $T_1$ ,  $T_2$  and final result

Figure 2 presents a comparison between linear, logarithmic and minmax depth computation. On the left, we consider a scene with near plane at 0.1, far plane at 1000.0 and objects close to near plane. On the right side objects are close to the far plane. The top presents a global view of the three interpolations and the bottom shows only the object depth range. As one can see, our method ensures that produced depth map values are always uniformly distributed.



**Fig. 2.** Comparison of the different depth interpolation methods

### 3.2 Stylization

We use the depth map previously generated to stylize our rendering. We apply desaturation, ambiance colorization and contours drawing to produce different comics stylized renderings. At this step, we have an interpolated minmax Depth Map  $DM$  and a texture  $T_S$  representing the scene. Let  $t(x, y, h, s, v)$  be the texel  $(x, y)$  of  $T_S$  and  $d(x, y, l)$  be the texel  $(x, y)$  of  $DM$ . We draw a quad that covers entirely the viewport to obtain one fragment  $f$  per pixel. Our rendering process is realized in the image space. Since the comics stylization do not influence the result image values:

$$f_v = t_v \quad (5)$$

*Desaturation:* According to a depth map information, the desaturation is one of the most important things in comics stylization. We propose two different stylizations. The first one reproduces comics stylization proposed by Sauvaget et al. [8]:

$$f_s = t_s \times (1.0 - d_l) \quad (6)$$

To enhance the contrast with the object distance, we also propose a quadratic computation:

$$f_s = t_s \times \sqrt{1.0 - d_l} \quad (7)$$

*Colorization:* Four different models are proposed, two of them reproduce Sauvaget et al. method [8]. The first one follows a classical comics stylization scheme and preserve the hue in the result image:

$$f_h = t_h \quad (8)$$

The second one reproduces an atmosphere comics stylization. In that case, the user gives a hue  $a_h$  to the atmosphere and the result image is colored as:

$$f_h = a_h \quad (9)$$

As we have a high precision depth map, we are able to colorize the result image with an atmosphere according to the depth of the object. We propose a linear and a quadratic interpolation:

$$f_h = t_h \times (1.0 - d_l) + a_h \times d_l \quad (10)$$

$$f_h = t_h \times \sqrt{1.0 - d_l} + a_h \times \sqrt{d_l} \quad (11)$$

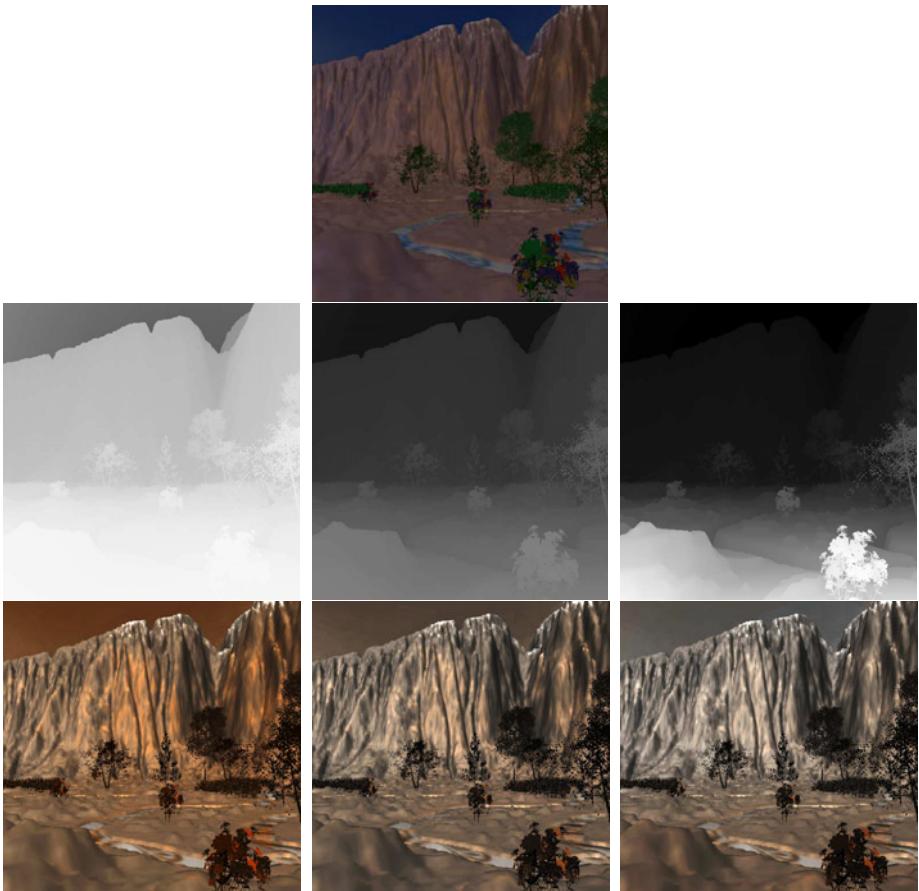
Note that the desaturation and the colorization are independent from each other and can be also combined according to user choices.

*Contour drawing:* We are able to enhance the result image using contours. In comics stylizations, contours are often produced to depict a small depth

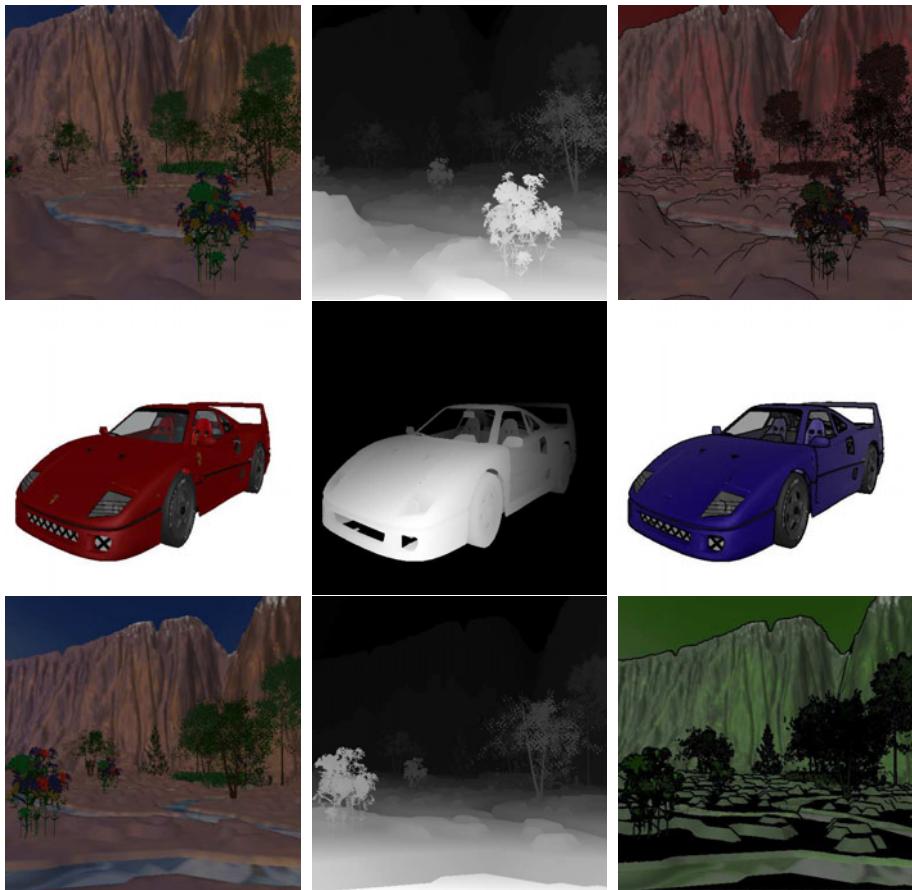
difference between two close objects. Therefore, according to our depth map, we implement image based solutions to create contours. Different algorithms are proposed: Prewitt and Laplace. The threshold is given by the user and contours are created when the detection algorithm finds on the  $DM$  a value greater than the threshold. This solution provides many different effects. A small threshold produces contours between close objects while a more important value creates black flat areas.

## 4 Results

As a preliminary result, Figure 3 presents a comparison between the linear (left), the logarithmic (center) and the minmax (right) methods. For a given scene (top of the figure), depth map produced for the three different methods (see the second



**Fig. 3.** Results produced using linear, logarithmic and minmax methods



**Fig. 4.** Various examples realized with our model

line). As one can see, only our minmax depth map covers the distance object range while the other approaches take always into account near and far planes. Thus, the image produced by our method is more detailed and contrasted. We finally add a sepia atmosphere to render the scene (see the third line).

Figure 4 illustrates different stylizations realized by our model. At the first line, we use the quadratic interpolation (see equation 11), a red atmosphere to compute the resulting hue and a Laplace contour detection. We can see that, at the foreground, colors of flowers are much more preserved while the background mountains are colored with the atmosphere color. At the third line, we produce other effects including black flat areas with Prewitt algorithm and a green atmosphere. These results show that our model is more suitable than previous ones to realize comics stylization effects on large scenes. Finally, at the center-line, we demonstrate that this model is also very efficient for a single object. In that case,

a quadratic desaturation (see equation 7), an atmosphere effect (see equation 9) and a Prewitt contour detection algorithm are used.

## 5 Conclusion

We have presented a new comics stylization model to render both single object or large scenes with multiple objects. Our model is real time, the frame rate is almost divided by 2 compared to the graphic pipeline (mountains scene is composed by one million vertices, the frame rate is 512 fps using OpenGL pipeline and 260 fps with our model using a depth map of 512 x 512 pixels). A linear depth map is computed in which the closest and farthest objects in the camera view are considered. Thus, we are able to produce a large variety of comics stylizations including desaturation, atmosphere, contours and black flat areas. Our model is suitable for any users since it is completely user-definable through a convenient GUI. As a future work, we plan to realize a treatment on the depth map to ensure temporal coherence during animation. In fact, if a closest or fare's object in the scene appears in a frame, our depth map will be affected and a brutal transition will probably disturb the viewer. Moreover, we aim to improve our model integrating new styles like blurred, complementary colors or many other styles depending on the depth map.

## Acknowledgements

This work has been performed within the Virtual Clone Studio project which is one of the *Serious Games* projects sponsored by the Ministry of Economy, Industry and Employment of the French Government.

## References

1. Lake, A., Marshall, C., Harris, M., Blackstein, M.: Stylized rendering techniques for scalable real-time 3d animation. In: NPAR 2000: Proceedings of the 1st International Symposium on Non-Photorealistic Animation and Rendering, pp. 13–20. ACM, New York (2000)
2. Klein, A.W., Li, W., Kazhdan, M.M., Corrêa, W.T., Finkelstein, A., Funkhouser, T.A.: Non-photorealistic virtual environments. In: SIGGRAPH 2000: Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques, pp. 527–534. ACM Press/Addison-Wesley Publishing Co., New York, NY, USA (2000)
3. Ewins, J.P., Waller, M.D., White, M., Lister, P.F.: Mip-map level selection for texture mapping. IEEE Transactions on Visualization and Computer Graphics 4, 317–329 (1998)
4. Barla, P., Thollot, J., Markosian, L.: X-toon: an extended toon shader. In: NPAR 2006: Proceedings of the 4th International Symposium on Non-Photorealistic Animation and Rendering, pp. 127–132. ACM, New York (2006)
5. Lindstrom, P., Koller, D., Ribarsky, W., Hodges, L.F., Faust, N., Turner, G.A.: Real-time, continuous level of detail rendering of height fields. In: SIGGRAPH 1996: Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques, pp. 109–118. ACM, New York (1996)

6. Olano, M., Kuehne, B., Simmons, M.: Automatic shader level of detail. In: HWWS 2003: Proceedings of the ACM SIGGRAPH/EUROGRAPHICS Conference on Graphics Hardware, Aire-la-Ville, Switzerland, Switzerland, Eurographics Association, pp. 7–14 (2003)
7. Anjyo, K.-i., Hiramitsu, K.: Stylized highlights for cartoon rendering and animation. *IEEE Comput. Graph. Appl.* 23, 54–61 (2003)
8. Sauvaget, C., Boyer, V.: Comics stylization from photographs. In: Bebis, G., Boyle, R., Parvin, B., Koracin, D., Remagnino, P., Porikli, F., Peters, J., Klosowski, J., Arns, L., Chun, Y.K., Rhyne, T.-M., Monroe, L. (eds.) *ISVC 2008, Part I. LNCS*, vol. 5358, pp. 1125–1134. Springer, Heidelberg (2008)

# Discovering Novelty in Gene Data: From Sequential Patterns to Visualization

Arnaud Sallaberry<sup>1</sup>, Nicolas Pecheur<sup>2</sup>, Sandra Bringay<sup>3</sup>,  
Mathieu Roche<sup>2</sup>, and Maguelonne Teisseire<sup>4</sup>

<sup>1</sup> LaBRI & INRIA Bordeaux - Sud Ouest & Pikkو, France  
[arnaud.sallaberry@labri.fr](mailto:arnaud.sallaberry@labri.fr)

<sup>2</sup> LIRMM - Université Montpellier 2, France  
[{pecheur,mathieu.roche}@lirmm.fr](mailto:{pecheur,mathieu.roche}@lirmm.fr)

<sup>3</sup> LIRMM - Université Montpellier 3, France  
[bringay@lirmm.fr](mailto:bringay@lirmm.fr)

<sup>4</sup> CEMAGREF - UMR TETIS, France  
[maguelonne.teisseire@cemagref.fr](mailto:maguelonne.teisseire@cemagref.fr)

**Abstract.** Data mining techniques allow users to discover novelty in huge amounts of data. Frequent pattern methods have proved to be efficient, but the extracted patterns are often too numerous and thus difficult to analyse by end-users. In this paper, we focus on sequential pattern mining and propose a new visualization system, which aims at helping end-users to analyse extracted knowledge and to highlight the novelty according to referenced biological document databases. Our system is based on two visualization techniques: Clouds and solar systems. We show that these techniques are very helpful for identifying associations and hierarchical relationships between patterns among related documents. Sequential patterns extracted from gene data using our system were successfully evaluated by two biology laboratories working on Alzheimers disease and cancer.

## 1 Introduction

DNA microarrays have been successfully used for many applications (diagnosis and characterisation of physiological states). They allow researchers to compare gene expression in different tissues, cells, or conditions and provide some information on the relative expression levels of thousands of genes that are compared in a few samples, usually less than a hundred (e.g., Affymetrix U-133 plus 2.0 microarrays measure 54,675 values). Nevertheless, due to the huge amount of data available, the way to process and interpret them in order to make biomedical sense of them remains a challenge. Data mining techniques, such as [1], have played a key role in discovering previously unknown information and shown that they could be very useful to biologists to identify subsets of microarray data, which could be relevant for further analysis.

However, the number of results is usually so huge that they cannot easily be analysed by the experts concerned. In [2], the authors propose a general process, called GeneMining, based on the DBSAP algorithm for extracting sequential

patterns from DNA microarrays. They obtained patterns of correlated genes ordered according to their level of expression. Although this method is useful, the way to select relevant patterns remains inefficient. For instance, depending on the values of parameters, they extract between 1,000 and 100,000 patterns that are not easy to interpret. Thus, the main aim of this work is to propose new visualization techniques to help biologists to navigate through the patterns. Biologists are also faced with the problem of locating relevant publications about the genes involved in the patterns. Even if some tools are now available to automatically extract information from microarray data (e.g., [3]), there are still no user-friendly literature search tools available for analysing patterns.

In this paper, we present an efficient tool to help biologists focus on new knowledge by navigating through large numbers of sequential patterns (i.e., sequences of ordered genes). Our contribution is twofold. First, we adapt two different techniques (i.e., point clouds and solar systems) to deal with data organized as a sequence and to produce an effective solution to the problem mentioned above. Second, using our system, the biologist can now be automatically provided with relevant documents extracted from the PubMed repository. Although the methods described in this paper mainly focus on sequences extracted from DNA arrays, they could easily be adapted to any other kind of sequential data.

The paper is organized as follows. In Section 2, we present the data we are working with and give an overview of related work. In Section 3, we describe our proposal and the associated tool. The evaluation of our systems are discussed in Sections 4 and 5. Section 6 concludes.

## 2 Preliminaries

In the framework of the PEPS-ST2I Gene Mining project<sup>1</sup>, we mined real data produced by the analysis of DNA microarrays (Affymetrix DNA U133 plus 2.0) to study Alzheimer's disease (AD) using the DBSAP algorithm [4]. This dataset was used to discover classification tools to distinguish between two *classes* (AD animals and healthy animals). In [4], the authors proposed to extract patterns of correlated genes ordered according to their level of expression. An example of pattern is  $<(MRV11)(PGAP1, GSK3B)>$  meaning that “the level of expression of gene *MRV11* is lower than that of genes *PGAP1* and *GSK3B*, whose levels are very similar”.

Although this method was interesting since they proved that sequential patterns could be very useful for biologists, the way of selecting relevant patterns remained a challenge. Actually, depending on the values of parameters, 1,000 to 100,000 patterns could be extracted and were consequently not easy to interpret. Biologists still needed a visualization tool to enable them to navigate through the huge amount of sequences, to select and order relevant innovative sequences (e.g. sequences where new gene correlations may exist), and to automatically

---

<sup>1</sup> This work was conducted in collaboration with the MMDN 'Molecular mechanisms in neurodegenerative dementias' laboratory, University of Montpellier 2, France. <http://www.mmdn.univ-montp2.fr>

query specific publications from Pubmed (or other publication database) on the selected genes. To summarize, an appropriate visualization tool needs to explore two kinds of data:

1. Gene sequences described by an ordered list of sets of genes and the class *supports* (i.e., the number of occurrences of this class in the database respecting this expression). As already mentioned, too many patterns are extracted. By using the k-means clustering algorithm with a sequence-oriented measure (S2MP [5]), we are able to identify groups of similar sequences and highlight a representative sequence called the center.
2. Documents in the literature dealing with genes from sequences. The documents are obtained from the Pubmed bibliographical database with or without gene synonyms [2]. We define a distance between a document and the gene sequence taking into account the publication date as well as the number of genes mentioned in the paper. The more recent the document and the more genes described in the paper, the closer the document will be to the concerned sequence.

The visualization tool, which is described in the following section, combines all these elements: Support, class, groups, and sets of documents. To facilitate specific tasks, it proposes two different visual representations [6]. The “Point Cloud” representation is mainly used to show the set of sequences while the “Solar System” is mainly used to focus on a specific sequence.

In [7], a visualization tool based on point clouds representing groups of sequences, is proposed. Sequences are placed according to an alignment in a 3-dimensional space. However, this approach is not able to take into account the hierarchy of sequences. Indeed, most previous works concerning visualization of biological sequences mainly focus on the representation of sequence alignments [8]. To the best of our knowledge, no method is currently available to visualize sequences and associated documents.

### 3 SequencesViewer

SequencesViewer<sup>2</sup> helps biomedical experts to browse and explore sequences of genes. In the following we describe the main representations.

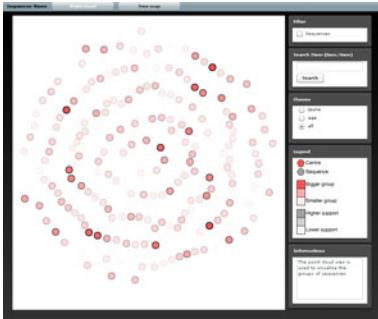
#### 3.1 Point Cloud

The Point Cloud representation allows biologists to visualize groups of gene sequences (see figures [1] [2]). It gives an overview of the centers of the groups, the distance from the centers, and associated sequences. Three steps are required to compute relevant positions of centers to limit the number of occlusions. We combine three algorithms and adapt them to our problem. An efficient interaction mode is also added to help users find the information they require.

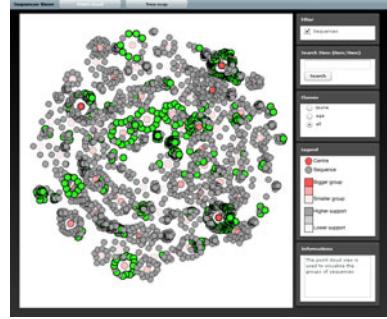
---

<sup>2</sup> All the screenshots of our system are available on the following web page:

<http://www.labri.fr/perso/sallaber/publications/isvc10/SequencesViewer.html>



**Fig. 1.** Point cloud without sequences



**Fig. 2.** Point cloud with sequences and highlighted researched items

**Main placement of the centers.** The basic idea is to place the centers in such a way that the Euclidean distances between them are proportional to the distances between the sequences given by a matrix of distances  $D$  containing S2MP measures [5]. Let  $d_{ij}$  be the matrix value for a center  $i$  and a center  $j$ . We want to find the coordinates  $p_i = (x_i, y_i)$  for each center  $i$  so that  $\|p_i - p_j\| \approx d_{ij}$  where  $\|p_i - p_j\|$  is the Euclidean distance between the centers  $i$  and  $j$ .

Different techniques are described in the literature to assign a location to items in an  $N$ -dimensional space. Multidimensional Scaling (MDS) technique [9] is often used in information visualization. It produces representations that reveal similarities and dissimilarities in the dataset using a matrix of ideal distances. In our application, we want to find positions in a 2-dimensional space. We use a MDS optimization strategy called Stress Majorization, which consists of minimizing a cost function (i.e. stress function) that measures the square differences between ideal distances and Euclidean distances in 2-dimensional space:

$$\sigma(p) = \sum_{i < j \leq n} \omega_{ij} (d_{ij} - \|p_i - p_j\|)^2 \quad (1)$$

where  $\omega_{ij} = d_{ij}^{-\alpha}$  and  $p = p_1, p_2, \dots, p_n$  is the actual configuration. We use  $\alpha = 2$ , which appears to produce good results in most cases, as shown by [10].

Several techniques have been developed to minimize the stress function (see [9] for an overview). In our application, we use a method introduced in [10] due to its simplicity, fast convergence, and quality of results. It consists of successively computing a simple function that returns position  $p_i$ :

$$p_i^{[t+1]} \leftarrow \frac{\sum_{j:j \neq i} \omega_{ij} \left( p_j^{[t]} + s_{ij}^{[t]} \cdot (p_i^{[t]} - p_j^{[t]}) \right)}{\sum_{j:j \neq i} \omega_{ij}} \quad (2)$$

where  $p_i^{[t]}$  is the position of the center  $i$  at time  $t$  and  $s_{ij}^{[t]} = \frac{d_{ij}}{\|p_i^{[t]} - p_j^{[t]}\|}$  if  $\|p_i^{[t]} - p_j^{[t]}\| \neq 0$  or  $s_{ij}^{[t]} = 0$  otherwise.

This iterative updating is performed for each node and repeated until a stable configuration is reached. At each step,  $\sigma(p)^{[t]} \geq \sigma(p)^{[t+1]}$  and the stress function converges to a local minimum [11].

**Initial placement of the centers.** One important aspect of these methods is to find an initial placement of the centers before performing the iterative process. Random placement is not efficient because every time the algorithm is executed for the same data, the final layout changes. Moreover, the stress majorization converges slowly and it can fall into local minima. In our system, we use the fold-free embedding defined in [12]. The algorithm selects four centers  $c_1, c_2, c_3$  and  $c_4$  so that they are in the periphery of the point cloud. The pair  $(c_1, c_2)$  has to be roughly perpendicular to the pair  $(c_3, c_4)$  in the layout. A fifth center  $c_5$  is selected so that it lies in the middle of the point cloud. A complete description of the selection process is available in [12].

Then, let  $x_i$  be  $d_{c_3i} - d_{c_4i}$  and let  $y_i$  be  $d_{c_1i} - d_{c_2i}$ . We can use  $(x_i, y_i)$  coordinates directly to place each center  $i$ . Unfortunately, this solution disregards the distance between  $i$  and  $c_5$ . To overcome this lack, the method computes the polar coordinate  $(\rho_i, \theta_i)$  of a center  $i$  so that  $\rho_i = d_{c_5i} \times R$  and  $\theta_i = \tan^{-1}\left(\frac{y_i}{x_i}\right)$ .

**Removal of center overlap.** The MDS method we have implemented does not avoid overlapping of centers. Node occlusions can mislead the user by hiding information. We thus run a node overlap removal algorithm after the MDS placement step described above. Gansner and Hu [13] implemented a simple but effective solution based on a nice adaptation of the stress majorization process.

This solution is based on a Delaunay triangulation computed for the set of centers and their current positions. A Delaunay triangulation is a triangulation that maximizes the minimum angle of all the angles of the triangles. We can represent the results of a triangulation on our centers as a planar graph  $G(V, E)$  where  $V$  is the set of the centers and  $E$  is the set of the edges of triangles. The node overlap is removed iteratively:

1. First, we compute a Delaunay triangulation on the current layout. Let  $G^{DT}(V, E^{DT})$  be the graph produced by the triangulation.
2. For each  $\{i, j\} \in E^{DT}$  an *overlap factor* is computed:  $t_{ij} = \max\left(\frac{a_i + a_j}{\|p_i - p_j\|}, 1\right)$  where  $a_i$  is the radius of the center  $i$ .  $t_{ij} = 1$  if the centers  $i$  and  $j$  do not overlap. If  $t_{ij} < 1$ , we can remove the overlap by extending the length of the edge  $\{i, j\}$  by this factor. A new ideal distance matrix is then computed:  $d_{ij}^{DT} = s_{ij}^{DT} \|p_i - p_j\|$  where  $s_{ij}^{DT}$  is a factor computed from  $t_{ij}$  to damp it:  $s_{ij}^{DT} = \min\{s_{max}, t_{ij}\}$ , with  $s_{max} > 1$  (1.5 in our implementation).  $s_{max}$  is the maximum amount of overlap we can remove at each step while keeping the same global configuration.
3. We now minimize the stress function using the process described above (see equation 2) with  $d_{ij}^{DT}$  and  $s_{ij}^{DT}$  in spite of  $d_{ij}$  and  $s_{ij}$ .

$$\sigma^{DT}(p) = \sum_{i < j \leq n} \omega_{ij} (d_{ij}^{DT} - \|p_i - p_j\|)^2 \quad (3)$$

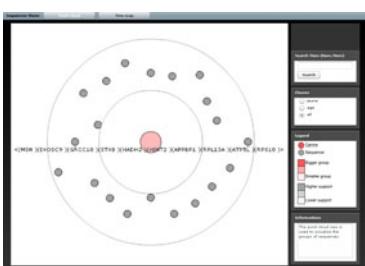
**Interactions and navigation.** The user can choose to visualize centers (figure 2) or both centers and their associated sequences using the check box labelled *Sequences*. The color of the centers is of different intensity which is proportional to the number of sequences associated with the center. The legend on the right helps the user evaluate the size of the groups. The research can be refined by applying different filters. First, sequences can be hidden by clicking on a filter button (see figure 1). An item can be searched and the sequences containing the search term are highlighted. The screenshot in figure 2 represents a map with the highlighted sequences (in green) resulting from a search operation.

### 3.2 Solar System

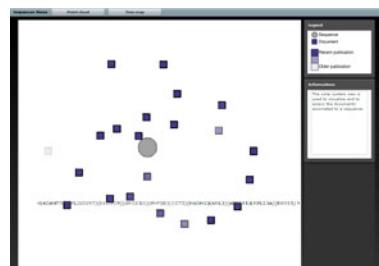
When the user double-clicks on a center in the point cloud view, he/she accesses a second view (figure 3) based on a solar system metaphor [14]. This view allows only the group of the selected center to be explored. The user can Zoom In/Out, move the whole map, search for an item or display a tooltip. The legend is also available. The center of the group is positioned in the middle of the visualization area (position  $(0, 0)$ ). Then, each sequence  $i$  is placed as follows:  $\theta_i = i \cdot \frac{2\pi}{n}$  where  $n$  is the number of sequences.

A second solar system based view is reachable from the first one by double-clicking on a sequence (figure 4). This view represents the sequence and its associated set of text documents, i.e. scientific papers dealing with the genes of the concerned sequence. These papers are extracted from the Pubmed biomedical library.

The sequence is positioned in the middle of the visualization as the center in the previous view. Documents are positioned around it. The distance between a document and the sequence is proportional to its proximity. The proximity depends on the year of publication and on the number of genes of the sequence referenced in the paper. The year of the publication is represented by different color intensities. A tooltip containing node information is displayed when the user clicks on the sequence or on a document. The document can be opened by double clicking on it.



**Fig. 3.** Group of sequences



**Fig. 4.** Sequence of genes and its associated documents

This type of visualization is convenient in the case of documents associated with sequences because the position of the documents helps the user select the sub-sets of documents that interest him/her. Of course, other ways of visualizing text documents are described in the literature. There are two main approaches: The visualization of specific subsections contained in large documents or the visualization of clustered collections [15]. Here, we focus on the second situation.

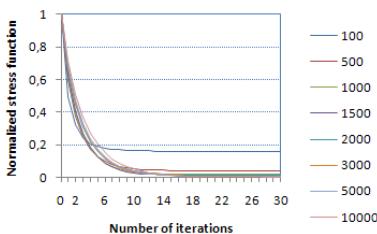
## 4 Discussion

In this section, the complexity of the algorithms and their limitations are discussed.

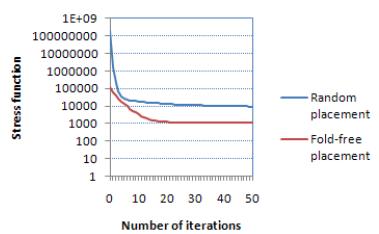
**Complexity of Point Cloud.** The point cloud remains the most complex view to produce. The calculation of the positions  $P^{[t]} = \{p_1^{[t]}, p_2^{[t]}, \dots, p_n^{[t]}\}$  (see equation 2) needs  $O(n^2)$  time where  $n$  is the number of centers. We tested the convergence of the iterative process using several random datasets (see figure 5). Empirical results indicate that no significant improvement in the placement occurs after 15 steps for each dataset. Thus, the algorithm used for the main placement, and the node overlap removal executes in  $O(n^2)$  time.

We already mentioned that a deterministic initial placement is more appropriate than a random one to obtain the same final layout for the same dataset, to make the stress majorization converge quickly and to avoid getting trapped in local minima. Figure 6 highlights the two last remarks. The values were computed using the stress function values obtained with a random dataset of 500 centers. We chose the fold-free layout because of the quality of its results and low time complexity ( $O(n)$ ,  $n$  is the number of centers).

**Complexity of the Solar System.** The solar system algorithm is performed in linear time. In the point cloud view, each group is placed using this method. Thus, it runs in  $O(n)$  where  $n$  is the total number of sequences. The complexity of the solar system view is rather insignificant as the number of sequences/documents is small.



**Fig. 5.** Convergence of the *stress majorization*: the numbers in the legend correspond to the number of the centers



**Fig. 6.** Convergence of the *stress majorization* using the initial placement *fold-free*

**Limitations of the visualization.** We developed our application in ActionScript 3. The complexity of the point cloud view prevents the user from displaying more than 500 groups. On the other hand, it is possible to visualize up to 25,000 sequences. Unfortunately, the representation of more than 5,000 sequences makes the navigation slow and tedious.

## 5 Evaluation

Evaluating a visualization system is complex, but in the context of a business application, when experts such as biologists and health professionals are involved, the evaluation should focus not only on technical and human aspects but also on the impact of the new system on their practice [16]. In our context, the aim of the evaluation was to measure to what extent our tool answered the needs of two teams of biologists. We undertook a semi-realistic evaluation in collaboration with the potential users to check the interest of the two visualizations. We worked with two laboratories on building the protocol and implemented it with the team working on Alzheimer's disease.

Our evaluation protocol is summative (i.e. the evaluation is conducted at the end of the design stage of the tool and just before its release), experimental (the evaluation is conducted on an usable tool), empirical (the evaluation is based on behavioural knowledge collected when the users actually use the tool) and non-automatic (the observations are made by a human observer). The evaluation is based on a cooperative technique. This one is a variant of the "think aloud" method during which an observer asks the user to use the tool and encourages him/her to think aloud when interacting with the system. It is called cooperative because the observer does not remain silent during the evaluation process but guides, explains and questions the user. A cooperative assessment enables 1) interactions between the user and the tool to be evaluated in controlled conditions and the user's perception of the different functionalities to be recorded; 2) Questionnaires are used to complement experimental methods. They quantify the user's impressions before and after using the system (satisfaction, anxiety, etc.) and often help him to take a step back because he/she is no longer involved in handling the tool.

Our protocol was given to the first biologist team working on Alzheimer's disease to test. The interview lasted approximately three hours per biologist. At the beginning of the test, we invited him to fill in a pre-evaluation form. We used this form to identify his profile, data-processing competences and current use of visualization tools. We then gave him only a very brief demonstration of the tool because we wanted him to discover its functionalities on his own. We asked him to carry out some tasks based on realistic scenarios. In so doing, he used the functionalities just as he would do for his work. During the test, one observer guided him and observed the way he used the functionalities. A second observer noted down the information given orally by the user, his reactions and his mimicries. At the end of the test, the user filled in a post-evaluation form.

With this evaluation, we collected 104 marks about the usefulness and the usability of the system (actually three tools have been evaluated). Usefulness focuses on how the system answers the user's needs. The user judges the usefulness according to his perception of a result/effort ratio. Usability focuses on the ease with which the expert used the system: Were the functionalities easy to use and to memorize? Did they include any errors? Did he find the system satisfactory? A system can succeed in fulfilling all the criteria of usability, but be completely useless. On the other hand, a system can be useful but too difficult to use. A successful system should be both useful and usable. This evaluation revealed the quality of our system, especially the solar system graded 3.75/5 for its utility and 3.70/5 for its ease (3.00/5 and 3.17/5 respectively for the point cloud). Two future directions are envisaged: 1) Organizing the sequences into groups according to their similarity did not prove to be useful to the users. Other types of organization, for example based on a discrimination measure of a sequence, may be more useful; 2) We will integrate other criteria to identify the most relevant documents associated with a sequence (e.g. the species involved in the studies, or the type of the document).

We are currently working with the second team of biologists, which also use DNA microarrays but to study breast cancer. This second evaluation will help us to generalize these first results. Indeed, we need to take into account the specificity of the evaluated functionalities, their specific context of use depending on the experts domain and the context of the evaluations themselves.

## 6 Conclusion

In this paper, we describe a new approach that helps the biologists to access and interpret sequential patterns extracted from DNA microarrays. Our system was developed in collaboration with biologists and with *Pikko*<sup>3</sup> company (specialized in information visualization). We combined and adapted two techniques from the information visualization domain. A point cloud view provides experts with a global representation of the sequential patterns. Combined with a first solar system view, it helps the biologist to navigate through groups of patterns and to compare and evaluate the relevance of the discovery correlations. Users can also access publications concerning each gene sequence through a second solar system view. This functionality improves the rapidity of searches and makes them less tedious. The algorithms we used were selected on the basis of their efficiency and their low complexity.

Our future work will be aimed at analysing the tests to evaluate the efficiency of the application in collaboration with biologists. After which, we will look for other applications to test its generalizability. Indeed, many data-mining algorithms used in different domains of application produce large amounts of information that cannot be used directly by experts. Whether our application is useful and adaptable to other data sets needs to be evaluated.

---

<sup>3</sup> <http://www.pikko-software.com/>

**Acknowledgments.** We would like to thank Mr. Guillaume Aveline and Mr. Faraz Zaidi for their technical assistance and the members of the Pikko company who provided us material resources.

## References

1. Cong, G.A., Tung, X., Pan, F., Yang, J.: Farmer: Finding interesting rule groups in microarray datasets. In: SIGMOD Conference, pp. 143–154 (2004)
2. Salle, P., Bringay, S., Teisseire, M., Chakkour, F., Roche, M., Rassoul, R.A., Verdier, J.M., Devau, G.: Genemining: Identification, visualization, and interpretation of brain ageing signatures. In: MIE, pp. 767–771 (2009)
3. Zeeberg, B.R., Feng, W., Wang, G., Wang, M.D., Fojo, A.T., Sunshine, M., Narasimhan, S., Kane, D.W., Reinhold, W.C., Lababidi, S., Bussey, K.J., Riss, J., Barrett, J.C., Weinstein, J.N.: Gominer: a resource for biological interpretation of genomic and proteomic data. *Genome Biol.* 4, 28 (2003)
4. Salle, P., Bringay, S., Teisseire, M.: Mining discriminant sequential patterns for aging brain. In: Combi, C., Shahar, Y., Abu-Hanna, A. (eds.) *Artificial Intelligence in Medicine*. LNCS, vol. 5651, pp. 365–369. Springer, Heidelberg (2009)
5. Saneifar, H., Bringay, S., Laurent, A., Teisseire, M.: S2mp: Similarity measure for sequential patterns. In: AusDM, pp. 95–104 (2008)
6. Schneiderman, B.: The eyes have it: A task by data type taxonomy for information visualizations. In: VL, pp. 336–343 (1996)
7. Chi, E.H.-h., Riedl, J., Shoop, E., Carlis, J.V., Retzel, E., Barry, P.: Flexible information visualization of multivariate data from biological sequence similarity searches. In: IEEE Visualization, pp. 133–140 (1996)
8. Lungu, M., Xu, K.: Biomedical information visualization. In: Kerren, A., Ebert, A., Meyer, J. (eds.) *GI-Dagstuhl Research Seminar 2007*. LNCS, vol. 4417, pp. 311–342. Springer, Heidelberg (2007)
9. Brog, I., Groenen, P.: *Modern multidimensional scaling: Theory and applications*. Springer, New York (1997)
10. Gansner, E.R., Koren, Y., North, S.: Graph drawing by stress majorization. In: *GDRAWING: Conference on Graph Drawing (GD)* (2004)
11. de Leeuw, J.: Convergence of the majorization method for multidimensional scaling. *J. Classification* 5, 163–180 (1988)
12. Priyantha, N.B., Balakrishnan, H., Demaine, E.D., Teller, S.J.: Anchor-free distributed localization in sensor networks. In: Akyildiz, I.F., Estrin, D., Culler, D.E., Srivastava, M.B. (eds.) *SenSys*, pp. 340–341. ACM, New York (2003)
13. Gansner, E.R., Hu, Y.: Efficient node overlap removal using a proximity stress model. In: Tollis, I.G., Patrignani, M. (eds.) *GD 2008*. LNCS, vol. 5417, pp. 206–217. Springer, Heidelberg (2009)
14. Nguyen, T., Zhang, J.: A novel visualization model for web search results. *IEEE Trans. Vis. Comput. Graph* 12, 981–988 (2006)
15. Jacquemin, C., Folch, H., Garcia, K., Nugier, S.: Visualisation interactive d'espaces documentaires. *Information Interaction Intelligence* 5, 59–84 (2005)
16. Ammenwerth, E.: Can evaluation studies benefit from triangulation? a case study. *International Journal of Medical Informatics* 70, 237–248 (2003)

# A Differential-Geometrical Framework for Color Image Quality Measures

Mourad Zéraï<sup>1</sup> and Olfa Triki<sup>2</sup>

<sup>1</sup> Ecole Supérieure Privée d'Ingénierie et de Technologies LAMSIN-ENIT, Tunis Al Manar University, Tunisia

<sup>2</sup> Ecole Supérieure Privée d'Ingénierie et de Technologies GRIFT, ENSI, Manouba University, Tunisia

**Abstract.** We propose a differential-geometrical framework for color Image Quality Measures (IQMs). Our approach is based on the definition of a relevant image distortion measure in a Riemannian way. To do this, we use the concept of geodesic distance and apply the theoretical setting to exhibit closed-forms for all the differential geometric attributes of two well-known color spaces: Helmholtz and Stiles manifolds. With these formulæ, we generalize some useful IQMs from the Euclidean framework to the Riemannian one. Finally, we present some experiments performed on real images, gradually distorted by different kinds of noise to conclude that the Riemannian IQMs are meaningful and relevant.

## 1 Introduction

Over the past fifteen years, dealing with color-images as manifold-valued signals has gained a major interest. Indeed, several authors have proposed methods and algorithms for image processing in a Riemannian setting [10,11,14,2]. Nevertheless, no Riemannian method is available to assess the quality of images resulting from such methods. Researchers and engineers therefore measure the image quality by Euclidean indicators while the images are processed in Riemannian ways.

This paper is a contribution to the settling of a framework for Riemannian Image Quality Measures (RIQMs). The proposed framework is then applied over two Riemannian geometries defined for color spaces by Helmholtz [8] and Stiles [15]. These geometries have been proposed in order to give metrics which are compliant with the human color vision. They, therefore, fit image processing problematics, and in particular human perception of image degradations and similarities.

The paper is organized as follows: first we recall some general basic aspects of Riemannian geometry used in the sequel sections. Then, we introduce some manifold statistics descriptors such as Riemannian mean and Riemannian variance which will be used to derive some Image Quality Measures (IQMs). This differential geometric material is then used to study two well-known color manifolds based on line elements definition: Helmholtz and Stiles' color manifolds. After, we present some experiments using the Normalized Mean Square Error to show that the framework is meaningful and effective.

## 2 Riemannian Framework for Color IQMs

### 2.1 Basic Aspects of Riemannian Geometry

We introduce some fundamental properties and notations for Riemannian manifolds, that will be applied in next sections, where we study the geometric properties of some color manifold. Those basic facts can be encountered, for example, in Jost [9]. Let  $M$  be a differential manifold. We denote by  $T_x M$  the tangent space of  $M$  at  $x \in M$  and  $TM = \cup_{x \in M} T_x M$ . In this paper, we are concerned only with real manifolds. Thus  $T_x M$  is isomorphic to  $\mathbb{R}^n$ , where  $n$  is the dimension of the manifold. If  $M$  is endowed with a Riemannian metric  $g$  then  $M$  is a Riemannian manifold and we denote it by  $(M, g)$ . The inner product of two vectors  $u, v \in T_x M$  is written  $\langle u, v \rangle_x := g_x(u, v)$ , where  $g_x$  is the metric at the point  $x$ . The norm of a vector  $v \in T_x M$  is defined by  $\|v\|_x := \sqrt{\langle v, v \rangle_x}$ .

The metric can be used to define the length of a piecewise smooth curve  $C : [a, b] \rightarrow M$  joining  $p$  to  $q$  through  $L(C) = \int_a^b \|C'(t)\|_{C(t)} dt$ , where  $C(a) = p$  and  $C(b) = q$ . Minimizing this length functional over the set of all curves we obtain a Riemannian distance  $d_g(p, q)$  which induces the original topology on  $M$ .

Given two vector fields  $V$  and  $W$  in  $M$ , the covariant derivative of  $W$  in the direction  $V$  is denoted by  $\nabla_V^g W$ . In this paper  $\nabla^g$  is the Levi-Civita connection associated to  $(M, g)$ . This connection defines an unique covariant derivative  $\frac{D}{dt}$ , where for each vector field  $V : M \rightarrow TM$ , along a smooth curve  $C : [a, b] \rightarrow M$ , another vector field is obtained, denoted by  $\frac{DV}{dt}$ . A curve  $C$  is a geodesic, starting from the point  $x$  with direction  $v \in T_x M$ , if  $C(0) = x$ ,  $C'(0) = v$  and (Einstein's summation convention is assumed to hold)

$$\frac{d^2 C_k}{dt^2} + \Gamma_{ij}^k \frac{dC_i}{dt} \frac{dC_j}{dt} = 0, \quad \text{for } k = 1, \dots, n, \quad (1)$$

where  $\Gamma_{ij}^k$  are the Christoffel's symbols expressed by

$$\Gamma_{ij}^k = \frac{1}{2} g^{km} \left( \frac{\partial g_{jk}}{\partial x_i} + \frac{\partial g_{ki}}{\partial x_j} - \frac{\partial g_{ij}}{\partial x_k} \right)$$

$(g^{ij})$  denotes the inverse matrix of the metric  $g = (g_{ij})$ , and  $x_i$  and  $v_i$  are the coordinates of  $x$  and  $v$ , respectively. Note that a Riemannian manifold is complete if its geodesics are defined for any value of  $t \in \mathbb{R}$ .

Finally, we define the Riemannian curvature : given the vector fields  $V, W, Z$  on  $M$ , we denote by  $R$  the curvature tensor defined by

$$R(V, W)Z = \nabla_W \nabla_V Z - \nabla_V \nabla_W Z + \nabla_{[V, W]} Z,$$

where  $[V, W] := WV - VW$  is the Lie bracket. If  $R(V, W) = 0$ , for all vector fields  $V, W$ , then  $M$  is called a flat Riemannian manifold.

## 2.2 Some Color Manifold Statistics

Statistics on manifolds is a topic that was extensively studied and applied by several authors. Fréchet [6] studied the distance, and by the way, the distance of two random variables in a general probabilistic manifold. In his monography [1], Amari defined the basic quantities of a statistical manifold, such as the Riemannian metric, the  $\alpha$ -connection, curvature, etc. Moakher [11], extended the variational definition of the mean of a data set from the Euclidean setting to the Riemannian one and applied this variational definition to derive the intrinsic distance for the manifold of symmetric positive definite matrices. In [12] Pennec established a general framework of Riemannian statistics in the context of signal processing. Fletcher used the intrinsic definition of mean and variance to study the statistical variability in nonlinear spaces and applied it to shape analysis and DT-MRI [5].

**Intrinsic Mean Calculation in Color Manifolds.** The main issue, in the intrinsic mean calculation in color manifolds, is the fact that a color manifold does not form a vector space, therefore the notion of an additive mean is not necessarily well defined. However, like the Euclidean case, the mean of a set points on  $M$  can be formulated as the point which minimizes the sum-of-squared distances to the given points (see Moakher [11], for instance). This formulation depends, obviously, on the definition of distance. A natural choice of distance is the Riemannian geodesic distance on  $M$ . This definition depends only on the intrinsic geometry of  $M$ . Following [11], we define the intrinsic mean of a collection of points  $x_1, \dots, x_N \in M$  as the minimizer in  $M$  of the sum-of-squared Riemannian distances to each point. Thus the intrinsic mean is

$$\bar{x} = \arg \min_{x \in M} \sum_{i=1}^N d(x, x_i)^2, \quad (2)$$

where  $d(\cdot, \cdot)$  denotes Riemannian distance on  $M$ . This is the definition of a mean value that we use in this paper.

**Intrinsic Variance Calculation in Manifolds.** Following the work of Fréchet [6], we define the variance of a given data points  $x_1, \dots, x_N$  on a complete, connected manifold  $M$  as the value of the squared Riemannian distance from the mean. That is, the variance is given by:

$$\sigma^2 = \frac{1}{N} \sum_{i=1}^N d(x, \bar{x})^2, \quad (3)$$

where  $\bar{x}$  is the mean of the data set as defined by (2).

**Riemannian Mean Square Error.** Similarly to the Fréchet [6] definition of variance, we define the Riemannian Mean Square Error (RMSE) between a distorted version  $\hat{x}$  and the true value of a data set  $x$  as the value of the squared

**Table 1.** Some Euclidean and Riemannian quality measure formulæ

Evaluation Technique	Euclidean Form	Riemannian Form
Average Difference	$\frac{\sum_{i,j}^{M,N} [F(i,j) - \hat{F}(i,j)]}{MN}$	$\frac{\sum_{i,j}^{M,N} d_R(F(i,j), \hat{F}(i,j))}{MN}$
Structural Content	$\frac{\sum_{i,j}^{M,N} [F(i,j)]^2}{MN}$	$\frac{\sum_{i,j}^{M,N} d_R(F(i,j), \mathbf{1})^2}{d_R(\hat{F}(i,j), \mathbf{1})^2}$
N. Mean Squared Error	$\frac{\sum_{i,j}^{M,N} [F(i,j) - \hat{F}(i,j)]^2}{[F(i,j)]^2}$	$\frac{\sum_{i,j}^{M,N} d_R(F(i,j), \hat{F}(i,j))^2}{d_R(F(i,j), \mathbf{1})^2}$
Maximum Difference	$\max_{i,j} (\ F(i,j) - \hat{F}(i,j)\ )$	$\max_{i,j} d_R(F(i,j), \hat{F}(i,j))$
Image Fidelity	$1 - NMSE$	$1 - RN MSE$
Peak MSE	$\frac{1}{MN} \frac{\sum_{i,j}^{M,N} [F(i,j) - \hat{F}(i,j)]^2}{[Max_{i,j} F(i,j)]^2}$	$\frac{1}{MN} \frac{\sum_{i,j}^{M,N} d_R(F(i,j), \hat{F}(i,j))^2}{[Max_{i,j} F(i,j)]^2}$

Riemannian distance between  $x$  and  $\hat{x}$  divided by the cardinal of the sample. More precisely RMSE is defined by

$$RMSE(x, \hat{x}) = \frac{1}{N} \sum_{i=1}^N d(x, \hat{x})^2.$$

Table 1 presents a set of IQM expressed under Euclidean geometry [4], and their equivalent formulæ in the Riemannian framework (note that  $\mathbf{1}$  in  $d(., \mathbf{1})$  is the neutral element for the image addition operator under the considered Riemannian manifold metric).

### 3 Differential Geometry of the Helmholtz Color Space

Hermann von Helmholtz (1821-1894), was the first to attempt to mathematically formulate the distance between colors by the concept of line element [8]. He defined the following line element as:

$$ds^2 = \left( \frac{dR}{R} \right)^2 + \left( \frac{dG}{G} \right)^2 + \left( \frac{dB}{B} \right)^2, \quad (4)$$

where R, G and B are the three color channels: Red, Green and blue.

In local coordinates, this can be expressed as a positive definite symmetric matrix:

$$(g_{ij})_{i,j=1,2,3} = \begin{bmatrix} \frac{1}{x_1^2} & 0 & 0 \\ 0 & \frac{1}{x_2^2} & 0 \\ 0 & 0 & \frac{1}{x_3^2} \end{bmatrix}, \quad (5)$$

where we use the coordinate notation  $x_1 = R$ ,  $x_2 = B$  and  $x_3 = G$ .

The color space is defined as a domain  $\Omega$  in the positive orthant  $\mathbb{R}_+^3$  defined by:

$$\mathbb{R}_+^3 = \{x \in \mathbb{R}^3 \mid x_i > 0, \quad i = 1, 2, 3\}. \quad (6)$$

Having the expression of the metric, we can now give the  $3^3$  Christoffel symbols using formula (7). It that the non-vanishing Christoffel symbols for the Helmholtz metric are:

$$\Gamma_{11}^1(\mathbf{x}) = -\frac{1}{x_1}, \Gamma_{22}^2(\mathbf{x}) = -\frac{1}{x_2}, \Gamma_{33}^3(\mathbf{x}) = -\frac{1}{x_3}, \quad \mathbf{x} \in \mathbb{R}^3. \quad (7)$$

Solving the geodesic equation, and considering the initial conditions  $C_i(0) = x_i$  and  $C'_i(0) = v_i$ ,  $i = 1, 2, 3$ , gives

$$C_i(t) = x_i e^{\frac{v_i}{x_i} t}, \quad t \in \mathbb{R}, \quad i = 1, 2, 3. \quad (8)$$

Equation (8) is the equation of the geodesic starting from  $\mathbf{x}$  in the direction of  $\mathbf{v} \in T_{\mathbf{x}} M$ . This geodesic curve is well defined for all  $t \in \mathbb{R}$ , so the Helmholtz manifold  $(\mathbb{R}^+)^3$  is complete and geodesic convex. Besides, the Riemannian distance from the color  $\mathbf{x} = C(0)$  to the color  $\mathbf{y} = C(t_0)$ ,  $t_0 > 0$ , is given by:

$$d_H(\mathbf{x}, \mathbf{y}) = \int_0^{t_0} \|C'(t)\|_H dt = \left\{ \sum_{i=1}^3 [\log(y_i) - \log(x_i)]^2 \right\}^{\frac{1}{2}},$$

where the subscript  $H$  in  $d_H$  stands for “with respect to the Helmholtz metric”.

If we consider two arbitrary colors  $\mathbf{x}$  and  $\mathbf{y}$  in the Helmholtz color manifold, such that  $C(0) = \mathbf{x}$  and  $C(1) = \mathbf{y}$  then the geodesic arc joining them is

$$\begin{aligned} C_i(t) &= x_i e^{(\log(y_i) - \log(x_i))t}, \\ &= x_i^{1-t} y_i^t, \quad t \in [0, 1], \quad i = 1, 2, 3. \end{aligned} \quad (9)$$

We can extend this *two points* geodesic interpolation equation to a *three points* one, i.e. between three color values  $\mathbf{x}, \mathbf{y}$  and  $\mathbf{z}$ , by:

$$C_i(a, b, c) = x_i^a y_i^b z_i^c, \quad a + b + c = 1, \quad (10)$$

where  $a, b$  and  $c$  are the real non-negative barycentric coordinates of the interpolated color in the interior of the triangle having  $\mathbf{x}, \mathbf{y}$  and  $\mathbf{z}$  as vertex. More generally, the *n-points* interpolation equation in the interior of the convex polyhedron having  $m$  vertexes  $\{\mathbf{x}^k\}_{k=1}^m$  is:

$$\forall i \in \{1, 2, 3\}, \quad C_i(a_1, a_2, \dots, a_m) = \prod_{k=1}^m (x_i^k)^{a_k}, \quad (11)$$

where

$$\sum_{k=1}^m a_k = 1, \quad a_k \geq 0, \quad k = 1, \dots, m.$$

### 3.1 Helmholtz Color Mean

If we consider an  $r \times c$  image  $\mathbf{I}(u, v) = (I_R(u, v), I_G(u, v), I_B(u, v))$ , where  $I_R$  is the red channel,  $I_G$  the green channel,  $I_B$  the blue channel, and  $(u, v)$  is the cartesian

coordinate of the considered pixel. The mean color value  $\bar{\mathbf{I}} = (\bar{I}_R, \bar{I}_G, \bar{I}_B)$  of this image corresponds to the *center of mass* of all the  $r \times c$  color pixel intensities, which can be obtained by plugging in (11) an equal value for all the  $m$ -th barycentric coordinates, where  $m = r \times c$ , i.e.:

$$\bar{I}_i = \prod_{k=1}^m (I_i^k)^{\frac{1}{m}}, \quad i = R, G, B. \quad (12)$$

### 3.2 Stiles' Color Metric

Walter W. Stiles modified the Helmholtz's proposal in order to better account for observations of threshold values (see [15] p. 660). Thus he proposed the following form of color-metric:

$$(ds)^2 = \left[ \frac{\zeta(R)}{\rho} dR \right]^2 + \left[ \frac{\zeta(G)}{C} dG \right]^2 + \left[ \frac{\zeta(B)}{\beta} dB \right]^2,$$

where:

$$\zeta(R) = \frac{9}{1+9R}, \quad \zeta(G) = \frac{9}{1+9G}, \quad \zeta(B) = \frac{9}{1+9B}.$$

The functions  $\zeta(R)$ ,  $\zeta(G)$  and  $\zeta(B)$  are determined experimentally. The constant  $\rho$ ,  $C$  and  $\beta$  are proportional to the limiting Weber fractions of the three cone responses at high luminance and Stiles obtained the following values:

$$\rho = 1.28, \quad \gamma = 1.65, \quad \beta = 7.25.$$

At high luminance, the Stiles' metric reduces to

$$(ds)^2 = \left( \frac{dR}{\rho R} \right)^2 + \left( \frac{dG}{\gamma G} \right)^2 + \left( \frac{dB}{\beta B} \right)^2,$$

and in this form its relationship with the Helmholtz's metric is obvious.

With the same notations as the previous section and using equation (7) we have

$$\Gamma_{11}^1 = -\frac{9}{1+9R}, \quad \Gamma_{22}^2 = -\frac{9}{1+9G}, \quad \Gamma_{33}^3 = -\frac{9}{1+9B}.$$

Another simple computation shows that the color-manifold endowed with Stiles' metric is flat.

Using (11), we can derive the differential equation of the geodesic curve

$$x_i''(t) = \frac{9x_i'(t)^2}{1+9x_i(t)}, \quad t \in [t_1, t_2], \quad i = 1, 2, 3, \quad (13)$$

where  $t$  is the arc-length parameter.

Solving the geodesic equation, and considering the initial conditions  $C_i(0) = x_i$  and  $C'_i(0) = v_i$ ,  $i = 1, 2, 3$ , gives

$$C_i(t) = \frac{1}{9} \left[ (1+9p_i) \exp \left( \frac{9v_i t}{1+9p_i} \right) - 1 \right], \quad t \in \mathbb{R}, \quad i = 1, 2, 3. \quad (14)$$

This geodesic curve (14) is well defined for all  $t \in \mathbb{R}$ , so the Stiles color manifold is complete. The Riemannian distance from the color  $\mathbf{x} = \mathbf{C}(0)$  to the color  $\mathbf{y} = \mathbf{C}(t_0)$ ,  $t_0 > 0$ , is then given by

$$d_S(\mathbf{x}, \mathbf{y}) = \left\{ \left[ \frac{1}{\rho} \log \left( \frac{1+9y_1}{1+9x_1} \right) \right]^2 + \left[ \frac{1}{\gamma} \log \left( \frac{1+9y_2}{1+9x_2} \right) \right]^2 + \left[ \frac{1}{\beta} \log \left( \frac{1+9y_3}{1+9x_3} \right) \right]^2 \right\}^{\frac{1}{2}}, \quad (15)$$

where the subscript  $S$  stands for “with respect to the Stile’s metric”.

The geodesic arc interpolating between two point  $\mathbf{C}(0) = \mathbf{x}$  and  $\mathbf{C}(1) = \mathbf{y}$  in the Stile’s color manifold is then given by

$$C_i(t) = \frac{1}{9} \left[ (1+9x_i)^{1-t} (1+9y_i)^t - 1 \right], \quad i = 1, 2, 3. \quad (16)$$

Using the barycentric coordinate  $\{\alpha_k\}_{k=1}^3$  with  $\sum_{k=1}^3 \alpha_k = 1$ , the geodesic interpolating formula inside a triangle with vertices  $\{\mathbf{x}^1, \mathbf{x}^2, \mathbf{x}^3\}$  is

$$y_i(\alpha_1, \alpha_2, \alpha_3) = \frac{1}{9} \left[ (1+9x_i^1)^{\alpha_1} (1+9x_i^2)^{\alpha_2} (1+9x_i^3)^{\alpha_3} - 1 \right], \quad i = 1, 2, 3. \quad (17)$$

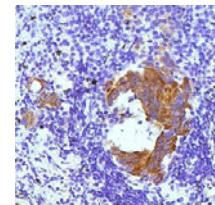
The generalization for a convex polyhedron with  $N$  vertices  $\{\mathbf{x}^k\}_{k=1}^N$  is straightforward:

$$y_i(\{\alpha_k\}) = \frac{1}{9} \left[ \prod_{k=1}^N (1+9x_i^k)^{\alpha_k} - 1 \right], \quad \text{with} \quad \sum_k \alpha_k = 1, \quad i = 1, 2, 3.$$

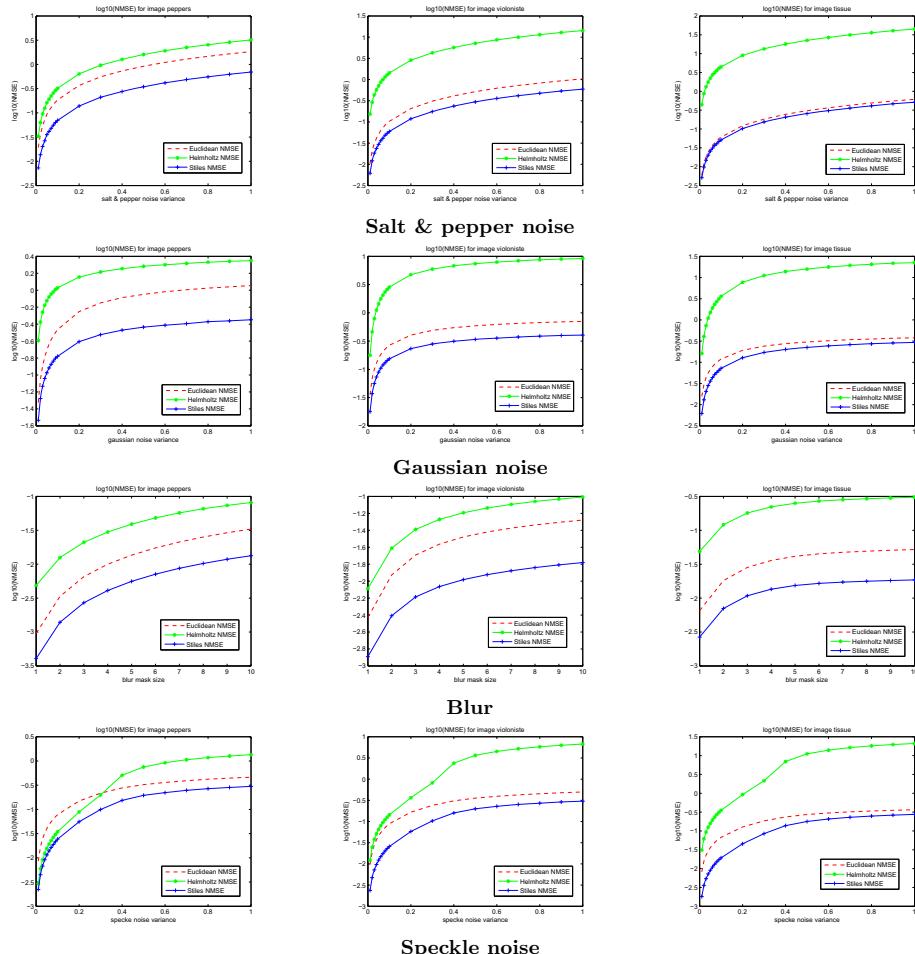
### 3.3 Experimental Results

In order to show that the settled Riemannian IQMs are meaningful and effective, a series of tests is performed on different image distortions and different images. Figure 11 presents NMSE obtained on a set of three distorted images: ‘Peppers’, a photo of Shagal painting ‘The Violinist’, and ‘Tissue’. Distortions applied are: (i)Salt & pepper noise, (ii)Gaussian noise, (iii)Gaussian blur and (iv)Speckle noise. For each case, different distortion rates are applied in order to get a set of degraded images. The plots represented for an  $< \text{image}, \text{distortion} >$  pair correspond to measured Mean Squared Error under three metrics: Euclidean, Helmholtz and Stiles’. In order to get clear graphics, the MSE is represented with a logarithmic transformation for all metrics.

The first erasing remark is the similarity of curve shapes for Euclidean and Riemannian NMSE, except for the speckle noise where we can notice a shape difference. This phenomenon could be explained by the fact that the first three distortions (salt & pepper, gaussian noise, and gaussian blur) are generated independently from the pixel color. Speckle noise, in contrary, is a multiplicative distortion, which results in bigger distosions for clear pixel colors, in the Euclidean metric. Riemannian IQMs therfore present a curve shape varying in a relatively uniform manner, whilst the Euclidean MSE presents a ‘jump’, probably reflecting the correlation between noise and image colors. In fact, comparing



input image

**Fig. 1.** IQM for distorted images ‘Peppers’, Shagal painting ‘The Violonist’ and ‘Tissue’

the Euclidean IQMs obtained for the three tested images (in the case of speckle noise) show that the darker the image, the less the measured noise. This observation confirms our analysis, and shows that Riemannian NMSE is more robust to the noise nature and to the input image properties.

These results constitute a way of further investigations on the behavior of Riemannian and Euclidean quality measures, which will include a wider range of image distosions and image samples. Note that evaluation of IQMs is generally performed by means of subjective methods such as MOS(Mean Opinion Score) [3][20], where statistics are used to compare quality indicators with human evaluation of image distortion. To our knowledge, there is no objective method to compare quality measures and their performance.

Note also that many authors [3][18] have pointed out the lacks of MSE-like measures in many situations. Research is currently underway to settle more per-formant quality indicators such as SSIM [18], which detects better image distor-tions with a good stability.

## 4 Conclusion

In this paper, we presented a framework for IQM in a Riemannian context. This work was motivated by two facts: on the one hand, there is an increasing interest in Riemannian methods for image processing, and on the other hand, there is a lack of evaluating image quality methods consistent with Riemannian setting. Many color manifolds defined by line elements were proposed in the last century, which are consistent, for some degree of success, with the human visual system. We have studied in this paper two of them: Helmholtz and Stiles color manifolds.

The experiments performed using the Riemannian Normalized Mean Square Error, show that the Riemannian setting is not only meaningful and consistent, but also deserves to be studied more extensively and compared with the classi-cal Euclidean IQM. No method is absolutely the best, but for each application there are more adapted methods than others. Our work is a contribution and an extension of available evaluation methods and requires a more exhaustive study in order to define the best field and conditions of use. The Riemannian quality measures should be evaluated in a broader context to verify their efficiency. It is, for instance, particularly interesting to generalize more efficient IQMs [19][18] like SSIM(Structural Similarity) to the Riemannian framework and compare the results with subjective measures like MOS(Mean Opinion Score).

## References

1. Amari, S.: Differential-Geometrical Methods in Statistics. Lecture Notes in Statistics. Springer, Heidelberg (1985)
2. Androulatsos, D., Plataniotis, K.N., Venetsanopoulos, A.N.: Distance measures for color image retrieval, pp. 770–774 (1998)
3. Eskicioglu, A.M.: Quality Measurement For Monochrome Compressed Images In The Past 25 Years (2000)

4. Eskicioglu, A.M., Fisher, P.S.: Image quality measures and their performance. *IEEE Transactions on Communications* (43:12), 2959–2965 (1995)
5. Fletcher, P.T.: Statistical Variability in Nonlinear Spaces: Application to Shape Analysis and DT-MRI, Ph.D Thesis (2004)
6. Fréchet, M.: Les éléments aléatoires de nature quelconque dans un espace distancié. *Annales de l'I.H.P.* 10:4, 215–310 (1948)
7. Frese, T., Bouman, C.A., Allebach, J.P.: A Methodology for Designing Image Similarity Metrics Based on Human Visual System Models: SPIE/IS&T, pp. 472–483 (1997)
8. Von Helmholtz, V.: *Handbuch der Physiologischen Optik*. Voss, Hamburg (1896)
9. Jost, J.: Riemannian Geometry and Geometric Analysis, 5th edn. Springer, Heidelberg (2008)
10. Kimmel, R.: A natural norm for color processing. In: Chin, R., Pong, T.-C. (eds.) *ACCV 1998*. LNCS, vol. 1351, pp. 88–95. Springer, Heidelberg (1997)
11. Moakher, M.: A Differential Geometric Approach to the Geometric Mean of Symmetric Positive-Definite Matrices. *SIAM J. Matrix Anal. Appl.* 26:3, 735–747 (2005)
12. Pennec, X.: Probabilities and statistics on Riemannian manifolds: basic tools for geometric measurements. In: *IEEE Workshop on Nonlinear Signal and Image Processing* (1999)
13. Sakuldee, R., Udomhunsakul, S.: Objective Performance of Compressed Image Quality Assessments. *International Journal of Computer Science* (2:4), 258–267 (2007)
14. Sochen, N., Zeevi, Y.Y.: Using Vos-Walraven line element for Beltrami flow in color images, EE-Technion and TAU HEP report, Technion and Tel-Aviv University (1992)
15. Stiles, W.S., Wyszecki, G.: *Color Science Concepts and Methods, Quantitative Data and Formulae*. John Wiley & Sons, Inc., Chichester (2000)
16. Vos, J.J.: From lower to higher colour metrics: a historical account. *Clinical & Experimental Optometry* (86), 348–360 (2006)
17. Vos, J.J., Walraven, P.L.: An analytical description of the line element in the zone-fluctuation model of colour vision II. The derivative of the line element, *Vision Research* (12), 1345–1365 (1972)
18. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: From Error Visibility to Structural Similarity. *IEEE Transactions on Image Processing* (13), 600–612 (2004)
19. Wang, Z., Bovik, A., Lu, L.: Why is image quality assessment so difficult. In: *ICASSP 2002*, pp. 3313–3316 (May 2002)
20. Sheikh, H.R., Bovik, A.C., de Veciana, G.: An information fidelity criterion for image quality assessment using natural scene statistics. *IEEE Trans. Image Processing* (14:12), 2117–2128 (2005)
21. Zéraï, M., Moakher, M.: Riemannian Curvature-Driven Flows for Tensor-Valued Data. In: Sgallari, F., Murli, A., Paragios, N. (eds.) *SSVM 2007*. LNCS, vol. 4485, pp. 592–602. Springer, Heidelberg (2007)

# Three Dimensional Reconstruction Using Vertical Constraints from a Photograph

Satoru Morita

Faculty of Engineering, Yamaguchi University

**Abstract.** Two photographs are necessary to reconstruct three dimensions conventionally. A problem to reconstruct three dimensions from one piece of photograph is a poor setting problem, that a right solution cannot be found uniformly. Even if human watches one piece of photograph, human feels depth. This is so that human has prior knowledge. We propose a technique to reconstruct three dimensions from one piece of photograph in this paper. We use a rectangle and perpendicular relations between quadrangles for prior knowledge here. We actually reconstruct three dimensions from one piece of photograph and show the effectiveness.

## 1 Introduction

Walz divided a line and an object from structure of line drawing and showed that whether line is a shadow or a crack route can be identify. [1]. Shirai showed how Knowledge about a building block found an object and how help to find a line from an arrangement of brightness to make line drawing [2]. Winston explained how a machine could learn essence of general ideas such as simple placement such as an arch [1]. We interpret line drawing here and we acquire knowledge of a building block from line drawing. We pay our attention to this in this paper and use line drawing on a photograph for the reconstruction.

On the other hand, the method using more than two photographs to reconstruct three dimensions are proposed. It is necessary to perform camera calibration beforehand, and to estimate internal parameter of a camera. With the photograph which does not understand internal parameter of a camera, it is necessary to estimate internal parameter of a camera from correspondence of feature point of two pieces of photographs [3][4][5]. With the internal parameter that was provided for two photographs, we reconstruct three dimensions from correspondence of feature point of two pieces of photographs. Photographs in large quantities are collected by internet, and a method to reconstruct a huge building is proposed [6][7]. Modeling and rendering architecture from photographs is proposed using both geometry and image based approaches [8]. We need two more images for 3D Modeling.

We reconstruct three dimensions from a photograph without using photographs more than two pieces without calibrating a camera before taking a photograph.

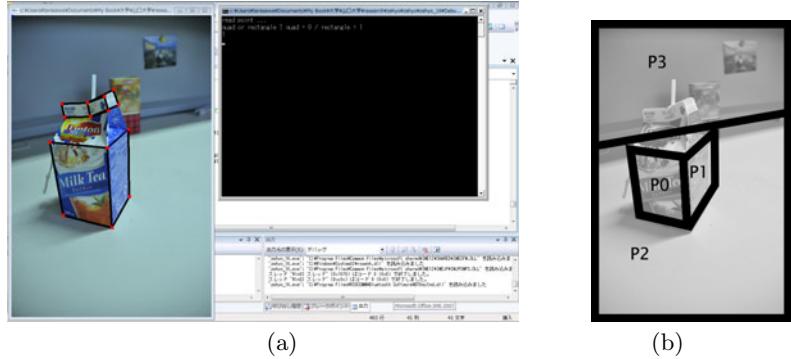
A problem to reconstruct three dimensions from one piece of photograph is a poor setting problem, and it is the problem that cannot demand a right solution uniformly. Even if human watches one piece of photograph, we feel depth. This is so that there is prior knowledge. Because human has this prior knowledge, we reconstruct the three dimensions from one piece of photograph in this paper. In other words we appoint a rectangular and a quadrangular regions except it to be included in one piece of photograph and propose a technique to reconstruct three dimensions by giving perpendicular relations between provided quadrangles.

Metrology for calibration needed 3D graphical modeling from single images was proposed based on vanishing line of a reference plane and a vanishing point for a direction not parallel to the plane [9]. We must find a vanishing point and a vanishing line in an image. In general, it is difficult to find a vanishing point and a vanishing line in an image. Modeling using geometric constraints through parallelepiped is proposed [10]. We must find parallelepiped in an image. Technique to reconstruct three dimensions is proposed by one piece of photograph [11]. It is a method to reconstruct three dimensions from an infinity point. Calibration was discussed but the input support of prior knowledge and automatic 3D graphical modeling were not discussed in them. Automatic 3D graphical modeling using learning algorithm based on Markov Random Field model is discussed in [12]. We find that it is difficult to generate 3d graphical model from a still image automatically. We do not use 3D modeling automatically, but we input the prior knowledge in the system. It is important that the input of the prior knowledge is easy and clear using support system for human. A user finds out a rectangle and an aspect of a quadrangle from all over the image without automating it in this study forcibly, and it is a technique to precisely reconstruct three dimensional space by marking a rectangle and a quadrangle using a user interface.

In an urban photograph with many artifacts or an indoor photograph, a rectangle can be found easily. In this study, we create a user interface to identify a quadrangle existing in a photograph and simplify the complicated work. We really reconstruct three dimensions from a photographed photograph and check the effectiveness.

## 2 Vertical Constraints between a Rectangle and a Square

In an urban photograph with many artifacts and an indoor photograph, a rectangle is found easily. In an outdoor photograph, buildings tend to be perpendicular to the ground. There are many things which are perpendicular to an aspect from the stability that indoor desk and chairs support human and a thing. On the other hand, even if a lot of rectangle is included in a photograph and a rectangle and a square are identified, human does not understand length such as 30 centimeters or a ratio of length such as 2:3; nor a small angle of 30 degrees or 40 degrees although there is meaning to a perpendicular angle. It is not necessary to include an quantitative angle and a quantitative length as prior knowledge in this paper. Therefore, we use a rectangle and perpendicular relations between quadrangles for prior knowledge.



**Fig. 1.** (a)User interface for generating the relation descriptions from a photograph.  
(b)The relation description of quadrangles from a photograph of a paper pack.

$$\begin{aligned}
 & [(P0V2)(P1V1)] \\
 \rightarrow & [(P0V1)(P1V2)(P0V3)(P1V0)] \\
 \rightarrow & [(P0V0)(P1V3)] \\
 \rightarrow & [(P2V0)(P3V1)(P2V3)(P3V2)(P2V1)(P2V2)] \\
 \rightarrow & [(P3V0)(P3V3)]
 \end{aligned}$$

**Fig. 2.** The order of constraints propagation

We identify a quadrangle on a screen. Next, we pick a rectangle from among the quadrangles. In addition, we choose it when we include perpendicular in a quadrangle. We create a user interface to simplify the complicated work. Figure 1(a) shows a user interface which generates a description from a photograph. Figure 1(b) shows the description of a quadrangle from a photograph of a paper pack. Two quadrangles which there is not come out on an image by two rectangles of  $P_0, P_1$  and a rectangle of  $P_2, P_3$  here. In the next section, we explain the calculation of propagation of constraints based on this example. Figure 2(a) shows the perpendicular relations between a quadrangle and a rectangle. Figure 2(b) shows the connection relations between the given quadrangles. Vertices  $v_0, v_1, v_2$  and  $v_3$  are four vertices which are identified from left top to right bottom along the line of the quadrangle in the photograph. Because four vertices of a rectangle are all perpendicular, a rectangle has relations as show in figure 2(c).The case often exists that something is only perpendicular to a vertice even in the case of a quadrangle. We use such relations, if they are self-evident.

The order of constraint propagation shows the order in which to reconstruct a vertex. Each vertex contributes to multiple perpendicular relations and connection relations because the vertex is a part of the lines and vertices forming a quadrangle and a rectangle. We calculate the perpendicular constraint and number of connections for a vertex. Then, using this vertex as a route, the adjacent vertex which has the perpendicular and connection relations is calculated.

$\text{rectangle} : P0P1$ $\text{quad} : P2P3 \quad P0(v2v3) = P1(v1v0) \quad v0v1 \perp v2v1$ $P0 \perp P1 \quad P0(v1v2) \text{ on } P2 \quad v1v2 \perp v3v2$ $P1 \perp P2 \quad P1(v1v2) \text{ on } P2 \quad v2v3 \perp v4v3$ $P0 \perp P2 \quad P2(v0v3) = P3(v1v2) \quad v1v0 \perp v3v0$ $P2 \perp P3 \quad v0v1 = v2v3$	$P0(v2v3) = P1(v1v0) \quad v0v1 \perp v2v1$ $P0(v1v2) \text{ on } P2 \quad v1v2 \perp v3v2$ $P1(v1v2) \text{ on } P2 \quad v2v3 \perp v4v3$ $P2(v0v3) = P3(v1v2) \quad v1v0 \perp v3v0$ $v0v1 = v2v3$	$v0v1 \perp v2v1$ $v1v2 \perp v3v2$ $v2v3 \perp v4v3$ $v1v0 \perp v3v0$ $v0v1 = v2v3$
(a)	(b)	(c)

**Fig. 3.** (a)The description of relations between quadrangles. (b)The connection relations between quadrangles. (c)The rectangles' properties.

This process is repeated for all vertices. Figure 2 shows the order of constraint propagation for figure II(b).

### 3 Camera Calibration Using Rectangle

We use a perspective camera model with internal parameter  $\mathbf{M}$ , and refer to pixel point  $\mathbf{m}$  on the photograph, with  $\mathbf{X}$  as the three dimensional coordinate of pixel point  $\mathbf{m}$ .

Following are the relations between a three-dimensional coordinate  $\mathbf{X}$  and a pixel point  $\mathbf{m}$  on a photograph taken by a camera with internal parameter  $\mathbf{M}$ ,

$$\lambda\mathbf{m} = \mathbf{MX}. \quad (1)$$

We define pixel point  $\mathbf{m}$  on a photograph and a three-dimensional coordinate  $\mathbf{X}$  as

$$\mathbf{m} = [x, y, 1]^\top$$

$$\mathbf{X} = [X, Y, Z]^\top.$$

Internal parameter  $\mathbf{M}$  of a camera is

$$\mathbf{M} = \begin{bmatrix} f \cdot k_u & f \cdot k_s & u_0 \\ 0 & f \cdot k_v & v_0 \\ 0 & 0 & 1 \end{bmatrix}, \quad (2)$$

where  $f$  is the focus distance,  $k_u, k_s$  and  $k_v$  are correspondence coefficients of the image pixel and  $u_0$  and  $v_0$  are the image center.

When we represent the four vertices of a rectangle as  $\mathbf{X}_0 \mathbf{X}_1 \mathbf{X}_2 \mathbf{X}_3$ , it follows that from a parallelogram condition that

$$\begin{aligned} X_0 - X_1 &= X_3 - X_2 \\ Y_0 - Y_1 &= Y_3 - Y_2 \\ Z_0 - Z_1 &= Z_3 - Z_2, \end{aligned} \quad (3)$$

where,

$$\mathbf{X0} = [X_0, Y_0, Z_0]^\top$$

$$\mathbf{X1} = [X_1, Y_1, Z_1]^\top$$

$$\mathbf{X2} = [X_2, Y_2, Z_2]^\top$$

$$\mathbf{X3} = [X_3, Y_3, Z_3]^\top$$

It follows from a perpendicular angle condition that

$$(X_3 - X_0) \cdot (X_1 - X_0) + (Y_3 - Y_0) \cdot (Y_1 - Y_0) + (Z_3 - Z_0) \cdot (Z_1 - Z_0) = 0. \quad (4)$$

We substitute (7) and (5) for (4), and where (4) is the following relations

$$\begin{aligned} Z_0^2((A_3 U_{Z3} - A_0)(A_1 U_{Z1} - A_0) + (B_3 U_{Z3} - B_0) \\ \cdot (B_1 U_{Z1} - B_0) + (U_{Z3} - 1)(U_{Z1} - 1)) = 0. \end{aligned} \quad (5)$$

Therefore, the condition to become a rectangle is written as the following equation.

$$\begin{aligned} E = (A_3 U_{Z3} - A_0)(A_1 U_{Z1} - A_0) + (B_3 U_{Z3} - B_0) \\ \cdot (B_1 U_{Z1} - B_0) + (U_{Z3} - 1)(U_{Z1} - 1) = 0 \end{aligned} \quad (6)$$

When we assume the number of rectangle included in an image as  $n$  and represent the evaluation equation of a  $i$ th quadrangle as  $E_i$ , we define the problem that determine affine conversion matrix  $M$  minimizing the following equation  $F$ .

$$F = \text{Min}_M(\sum_{i=0}^n E_i \cdot E_i) \quad (7)$$

Internal parameter  $M$  of the camera is found using the steepest descent method.

## 4 Three-Dimensional Reconstruction Based on Perpendicular Constraints

Vertices of a quadrangle are reconstructed according to the order of the constraint propagation.

### 4.1 Reconstruction of Three Adjacent Vertices to Form Perpendicular Relations Including One Vertex

When we assume three vertices  $\mathbf{X0X2X3}$  forming perpendicular relations to each other adjacent to vertex  $\mathbf{X1}$ ,

$$(X_0 - X_1) \cdot (X_2 - X_1) + (Y_0 - Y_1) \cdot (Y_2 - Y_1) + (Z_0 - Z_1) \cdot (Z_2 - Z_1) = 0 \quad (8)$$

$$(X_0 - X_1) \cdot (X_3 - X_1) + (Y_0 - Y_1) \cdot (Y_3 - Y_1) + (Z_0 - Z_1) \cdot (Z_3 - Z_1) = 0 \quad (9)$$

$$(X_3 - X_1) \cdot (X_2 - X_1) + (Y_3 - Y_1) \cdot (Y_2 - Y_1) + (Z_3 - Z_1) \cdot (Z_2 - Z_1) = 0 \quad (10)$$

$$A_0 \cdot Z_0 = X_0, B_0 \cdot Z_0 = Y_0, A_1 \cdot Z_1 = X_1, B_1 \cdot Z_1 = Y_1, A_2 \cdot Z_2 = X_2, B_2 \cdot Z_2 = Y_2, \\ A_3 \cdot Z_3 = X_3, B_3 \cdot Z_3 = Y_3.$$

$$\alpha_0 \cdot Z_0 \cdot Z_2 + \alpha_1(Z_1) \cdot Z_2 + \alpha_2(Z_1) \cdot Z_0 + \alpha_3(Z_1) = 0$$

$$\beta_0 \cdot Z_3 \cdot Z_2 + \beta_1(Z_1) \cdot Z_3 + \beta_2(Z_1) \cdot Z_2 + \beta_3(Z_1) = 0$$

$$\gamma_0 \cdot Z_0 \cdot Z_3 + \gamma_1(Z_1) \cdot Z_0 + \gamma_2(Z_1) \cdot Z_3 + \gamma_3(Z_1) = 0 \quad (11)$$

From this simultaneous equation  $Z_0$  can be found by erasing  $Z_2, Z_3$ , and solving a quadratic equation of  $Z_0$ , and  $Z_2, Z_3$  can be found.

## 4.2 Reconstruction of the Fourth Vertex of Rectangle

If three vertices **X1** **X2** **X3** are found among four vertices of a rectangle, we calculate remaining vertex **X0** from the following parallelogram conditions.

$$\begin{aligned} X_0 - X_1 &= X_3 - X_2 \\ Y_0 - Y_1 &= Y_3 - Y_2 \\ Z_0 - Z_1 &= Z_3 - Z_2 \end{aligned} \quad (12)$$

Because  $A_0 \cdot Z_0 = X_0, B_0 \cdot Z_0 = Y_0, A_1 \cdot Z_1 = X_1, B_1 \cdot Z_1 = Y_1, A_2 \cdot Z_2 = X_2, B_2 \cdot Z_2 = Y_2, A_3 \cdot Z_3 = X_3$  and  $B_3 \cdot Z_3 = Y_3, Z_0$  can be found, and  $X_0, Y_0$  can also be found.

## 4.3 The Reconstruction of a Vertex on a Plane

When we calculate vertex **X4** stepping on quadrangle **X0 X1 X2 X3**,

$$\mathbf{X0X1} \times \mathbf{X2X1} = \mathbf{N}$$

becomes a normal vector. From a plane equation, the following equation is satisfied.

$$N_x(X - X_0) + N_y(Y - Y_0) + N_z(Z - Z_0) = 0, \quad (13)$$

where

$$\mathbf{N} = [\mathbf{N_x}, \mathbf{N_y}, \mathbf{N_z}]^\top$$

$$\mathbf{X0} = [X_0, Y_0, Z_0]^\top.$$

Substitute **X4** for **X** of a plane equation.

Because  $A_4 \cdot Z_4 = X_4$  and  $B_4 \cdot Z_4 = Y_4$ ,  $Z_4$  is gotten from the next equation.

$$Z_4 = \frac{N_x \cdot X_0 + N_y \cdot Y_0 + N_z \cdot Z_0}{N_x \cdot A_4 + N_y \cdot B_4 + N_z}, \quad (14)$$

where

$$\mathbf{X4} = [\mathbf{X}_4, \mathbf{Y}_4, \mathbf{Z}_4]^\top.$$

#### 4.4 The Reconstruction of a Vertex on the Straight Line That Is Perpendicular to a Plane

When we calculate **X4** on quadrangle **X0 X1 X2 X3**, the normal vector **N** is provided from the next equation.

$$\mathbf{X0X1} \times \mathbf{X2X1} = \mathbf{N} \quad (15)$$

When we understand endpoint **X4** of the straight line that is perpendicular to a plane, an equation of the straight line is as follows.

$$\frac{x - X_4}{N_x} = \frac{y - Y_4}{N_y} = \frac{z - Z_4}{N_z} \quad (16)$$

Substitute **X5** for **X** of the equation. Because  $B_5 \cdot Z_5 = Y_5$ ,  $Z_5$  is provided from the next equation.

$$Z_5 = \frac{Y_4 \cdot N_z - Z_4 \cdot N_y}{N_z \cdot B - N_y}, \quad (17)$$

where

$$\mathbf{X5} = [\mathbf{X}_5, \mathbf{Y}_5, \mathbf{Z}_5]^\top$$

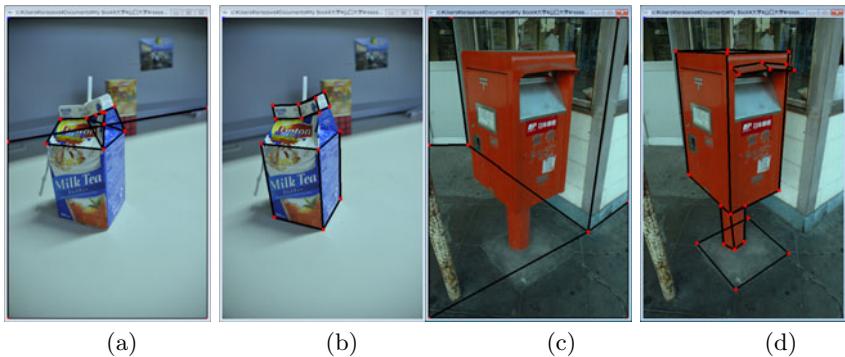
Three dimensions are reconstructed by solving these presumed equations according to the turn that constraint propagates. When we reconstruct three-dimensional world from one piece of a two-dimensional image, the three dimensions world is not determined with a few perpendicular constraint between given segments of a line and between planes.

A necessary condition for all perpendicular angles is that a grand total of inner products must be zero. In recent years, photograph composition flourishes.

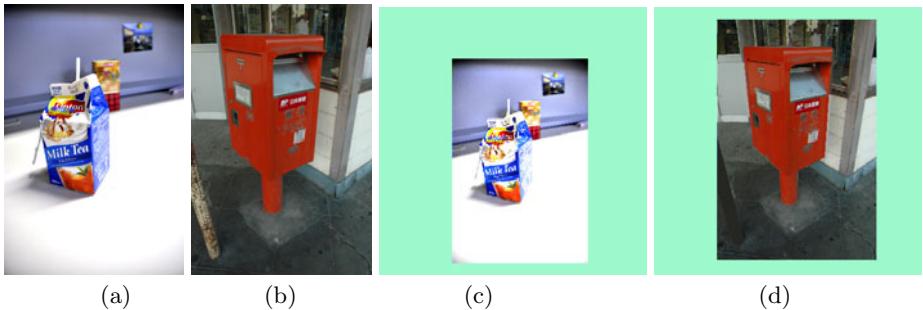
It is easy to change an effect of light, and to reconstruct three dimensions satisfying the position relations of a photograph by adding a source of light. In addition, this method is useful to generate the three-dimensional contents of an object appearing in a photograph.

### 5 Three-Dimensional Reconstruction from a Photograph

Figures 5(a) and (b) show a photograph of a paper pack and a post, respectively. Figures 4(a) and (b) show a quadrangle and a rectangle in a photograph of a paper pack. Figures 4(c) and (d) show a quadrangle and a rectangle in the photograph of the post. We made a CG model using this method based on this input. A CG model is reconstructed to satisfy all constraints. We put the texture of an image on the surface of a CG model, but put the texture of a photograph only on a thing in this side from a viewpoint in a quadrangle piled up most. A few part considers it in texture in this side from a viewpoint most and sticks texture of a part in this side on the part which there is not in the figures toward you. It is this side, and the person that  $Z$  is small can calculate context of each aspect from depth  $Z$  from a viewpoint easily. As figure 6(b) and 6(d) showed a photograph taken at the same position as a photograph using the same camera

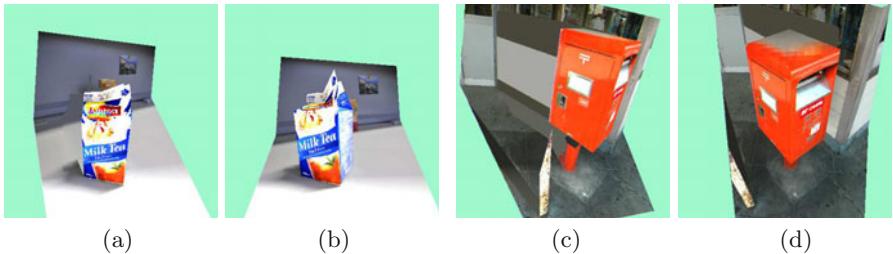


**Fig. 4.** (a)The quadrangles from a photograph of a paper pack. (b)The rectangles from a photograph of a paper pack. (c)The quadrangles from a photograph of a post. (d)The rectangles from a photograph of a post.

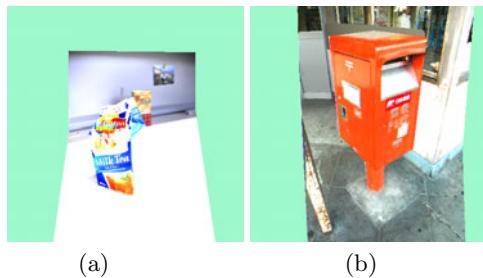


**Fig. 5.** (a)A photograph of a paper pack, (b) a photograph of a post. (c)The photograph taken from CG model using the same parameter as from a photograph of a paper pack. (d)The photograph taken from CG model using the same parameter as from a photograph of a post.

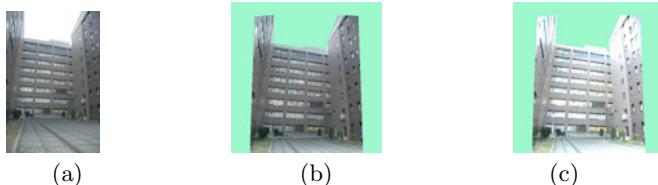
as a photograph, inadequate texture in the photograph is not found in figure 6(b) and 6(d). Figures 5(c) and (d) show the photograph which photographed a CG model again from a position same as a photograph of figure 5(a)and (b). Figures 6(a) and (b) show a photograph of the paper pack which photographed a generated CG model again from a photograph and a different position of figure 5(a). Figures 6(c) and (d) show a photograph of the post which photographed a generated CG model again from a photograph and a different position of figure 5(b). We understand that an object is reconstructed adequately. Figures 7(a) and (b) change a direction of light of figure 5(a) and (b), and it is the photograph which photographed a generated CG model again. It is found that that a direction of light can be changed. Figure 8(a) shows original photograph of the building. Figure 8(b), and (c) show the photographs taken for the generated CG model. The lighting of (c) is different from the lighting of (b).



**Fig. 6.** (a)(b)The photograph taken from a CG model using different parameter as from a photograph of a post. (c)(d)The photograph taken from a CG model using different parameter as from a photograph of a paper pack.



**Fig. 7.** (a)The photograph taken from CG model by the different parameter from a photograph of a paper pack. (b)The photograph taken from CG model by the different parameter from a photograph of a post.



**Fig. 8.** (a)Original photograph(b)The photograph taken for the generated CG model.(c) The lighting of (c) is different from the lighting of (b).

## 6 Conclusion

We proposed a technique to reconstruct three dimensions from a photograph because humans have prior knowledge. When sufficient constraints exist, three dimensions can be reconstructed precisely to the original state. When constraints are insufficient, the reconstruction is possible interactively using user interface. An object, such as an electric light pole, can be reconstructed easily to its original state by using prior knowledge of a circle. In addition, it is possible to easily reconstruct by adding information such as a square which is often included in an image and prior knowledge.

## References

1. Winston, P.H.: *The Psychology of Computer Vision*. McGraw-Hill Inc., New York (1975)
2. Shirai, Y.: A Context Sensitive Line Finder for Recognition of Polyhedra. *Artif. Intel.* 4(2), 95–120 (1973)
3. Tsai, R.Y., Huang, T.S.: Uniqueness and estimation of three dimensional motion parameters of a rigid objects with curved surfaces. *IEEE PAMI* 6(1), 13–27 (1984)
4. Hartley, R.I.: Stereo from uncalibrated cameras. In: *CVPR*, pp. 761–764 (1992)
5. Zhaing, Z.: Determining the epipolar geometry and its uncertainty: A reviews. *IJCV* 27(2), 161–195 (1998)
6. Snavely, N., Seitz, S., Szeliski, R.: Photo Tourism: Exploring photo collections in 3D. In: *ACM SIGGRAPH*, pp. 835–846 (2006)
7. Snavely, N., Garg, R., Seitz, S.M., Szeliski, R.: Finding Paths through the World's Photos. In: *ACM SIGGRAPH*, pp. 11–21 (2008)
- 8.Debevec, P.E., Taylor, C.J., Malik, J.: Modeling and Rendering Architecture from Photographs: A Hybrid Geometry and Image-Based Approach. In: *SIGGRAPH 1996*, pp. 11–20 (1996)
9. Criminisi, A., Reid, I.D., Zisserman, A.: Single View Metrology. *International Journal of Computer Vision* 40(2), 123–148 (2000)
10. Wilczkowiak, M., Sturm, P.F., Boyer, E.: Using Geometric Constraints through Parallelepipeds for Calibration and 3D Modeling. *IEEE Trans. Pattern Anal. Mach. Intell.* 27(2), 194–207 (2005)
11. Hoiem, D., Efros, A.A., Hevert, M.: Automatic Photo Pop-up. In: *ACM SIGGRAPH*, pp. 577–584 (2005)
12. Saxena, A., Sun, M., Ng, A.Y.: Learning 3-D Scene Structure from a Single Still Image. In: *ICCV 2007*, pp. 1–8 (2007)

# A Framework for Visual and Haptic Collaboration in Shared Virtual Spaces

Lei Wei<sup>1</sup>, Alexei Sourin<sup>1</sup>, and Herbert Stocker<sup>2</sup>

<sup>1</sup> Nanyang Technological University, Singapore

<sup>2</sup> Bitmanagement Software GmbH, Germany

**Abstract.** We propose a framework for visual and haptic collaboration in X3D/VRML shared virtual spaces. In this collaborative framework, two pipelines—visual and haptic—complement each other to provide a simple and efficient solution to problem requiring collaboration in shared virtual spaces on the web. We consider shared objects defined as virtual object with their visual and physical properties rendered synchronously on each client computer. We introduce virtual tools which are shared objects associated with interactive and haptic devices. We implemented the proposed ideas as a server-client framework with a dedicated viewer. We discuss two implementation frameworks based on the strong and thin server concepts.

## 1 Introduction

Extensible 3D (X3D) and its predecessor Virtual Reality Modeling Language (VRML) are open standard file formats and run-time architectures to represent and communicate 3D scenes and objects (<http://www.web3d.org>). For many years VRML and X3D have successfully been used for storage, retrieval and playback of real time graphics content embedded in applications supporting a wide array of domains and user scenarios. However, both VRML and X3D lack an ability to define tangible physical properties of virtual objects and render them with haptic devices. Also when it comes to setting collaborative shared scenes, VRML and X3D require 3<sup>rd</sup> party software tools to be used.

Compared to vision, very little has been done to make touch a regular part of communication with a computer in VRML and X3D virtual spaces. Thus, H3D (<http://www.h3d.org>) uses X3D file format as a container for the 3D scene definition which can be then locally rendered and interacted haptically. Besides H3D, some other research works on incorporating haptics with X3D have also been done. A survey of medical applications that make use of Web3D technologies including haptic interfaces can be found in [6]. In [7], an X3D extension was proposed for volume rendering in medical education and surgical training, which also incorporates haptic devices for immersive interactions. In [1], a prototype molecular visualization application with haptic interaction has been developed based on Web3D standards. In [5], an X3D-based haptic approach to e-learning and simulation was proposed. In [9], several haptic modes have been introduced to do volume haptics, such as viscosity, gradient, force, vector follow, vortex tube, surface/friction and combined mode. In [10], an X3D-based 3D Object Haptic Browser was proposed and implemented to

augment the user experience of accessing the network. It can be also noted that attention of some general haptic research works shifts towards integration with X3D [2-4]. In [11], it was proposed to define virtual objects with tangible physical properties by concurrent using of implicit, explicit and parametric function definitions and procedures.

There are several currently available software tools supporting visual collaboration in X3D and VRML scenes such as ABNet (<http://kimballsoftware.com/abnet>), Planet 9 GeoFeeder Server (<http://www.planet9.com>), Octaga Collaboration server (<http://www.octaga.com>) and BS Collaborate server (<http://www.bitmanagement.com>). These are server-client tools which allow for setting up shared virtual spaces, make some objects shared and provide for text communication between the users. No haptic collaboration is supported by these tools.

In Section 2, we introduce our approach to visual and haptic collaboration in 3D web. In Section 3, we present our implementation framework and illustrate it with a few examples. Finally, we draw conclusions and outline further work.

## 2 Challenges of Setting Networked Visual and Haptic Collaboration

Networked collaborative applications require *information transmission, shared objects and events, synchronization, and consistency control*.

*Visual information transmission* is performed either by streaming images, or by model transmission followed by its rendering on client computers. *Touch* is implemented through the force feedback provided by various haptic devices such as desktop robotic arms. Haptic refresh rate (1000 Hz) is much higher than the visual refresh rate (30-50 Hz) which imposes a big challenge for setting up networked haptic collaboration due to the networked bandwidth.

Ideally, each object in a shared virtual scene has to become a *shared object*, i.e. changes of its location, geometry, appearance and tangible physical properties have to be synchronously seen and felt by all the users in the scene. However, in reality only a limited number of objects in the scene are declared shared (mostly their location and orientation) because of the network bandwidth limitation. The scene can be shared by downloading it to the client computer and by running locally different scripts controlling object behavior that can be synchronized by either timer signals common for internet connected computers or event messages sent between the clients and/or server. In shared virtual scenes there are often a few shared objects which are visual avatars of the users implemented at the viewer level as either standard objects stored on the client computer or web-located objects, which URLs are provided to the viewer via different ways (e.g. scripts, html pages, and databases associated with the scene).

In contrast to commonly used approach, we define a *shared object* as an object which visual and physical properties are synchronously rendered by all the client computers as they change. We propose to define tangible physical properties of the virtual objects as two components: *surface properties (friction, elasticity)* and *inner properties (density, force field)*. Like visual properties, the physical properties are associated with either geometry of the visual shared object or an invisible container.

For haptic rendering, there is a need in an object that can be used as a 3D visual representation (avatar) of the virtual haptic tool. In our considerations, such object is a shared object that is associated with the respective haptic device to change its location and orientation and possibly even geometry and appearance while it is being applied. The motion of this object can be seen by all the participants synchronously and it can interact with other objects that possess tangible physical properties. The way how such *shared tool* object is associated with the haptic device should be flexible since these devices have limited angles of rotations for their actuators and may need to re-attach the tool-avatar to reach the point of interest at a required angle.

Shared objects and tools have to be used together with *shared events*. Shared events are inherited from the event transmission where an event from one client is sent to other clients either through server or directly.

No matter which way the collaborative platform adopts, *synchronization* is performed to allow multiple users to immerse into the virtual scene and share it. A crucial issue here is how to prevent concurrent operations on shared objects performed by different clients.

To ensure *synchronization* among all clients, *consistency control* algorithms such as locking and serialization mechanisms are required. Shared objects can be *locked* by the user for their private use. Hence, a shared object selected by the user as a tool can be then set as unavailable for other users. They will be able to see how this object moves following the motion of the user's haptic device actuator. However, we permit to use a shared object as a tool by several users. The motion of the object will be then a resulting motion controlled by the haptic devices connected to it. *Locking* can also be useful when interactive changes to the object are being performed by one of the users while others are not expected to make any change to it. Last but not least, each user may require more than one haptic device with different tools associated and used concurrently. These can be either different haptic devices or devices with multiple actuators, which can be considered as different virtual tools in the scene.

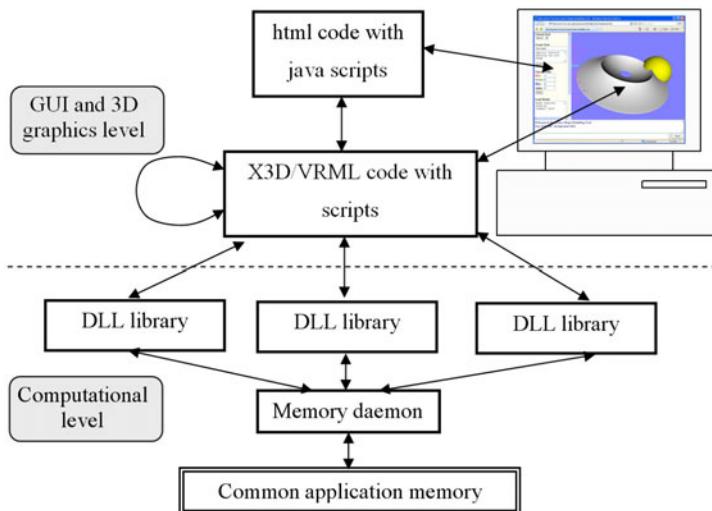
Strong locking means that a client has to release the lock before another client can succeed in requesting the lock. Weak locking indicates that if a client requests the lock while another owns it, the owning client loses the lock and the requester gains the lock.

## 3 Setting Collaboration in VRML/X3D Scenes

### 3.1 Making Interactive Collaborative Applications with X3D and VRML

To develop an efficient, computationally intensive VRML/X3D-based interactive application that can be called from a web browser, a so-called native script execution method can be used. By writing a specific string in the *url* field, the browser automatically loads a library file and executes the code compiled from Java or C++ languages. The library files for different browsers have only a slight difference, so the source code can be ported to them with only minor modifications.

Hence, there can be 3 parts in such collaborative application: *html-code*, *X3D/VRML code with scripts*, and *binary libraries* (Fig. 1). All these components can be stored on the server and client machines. These three parts should be able to exchange data during the run-time of the application.



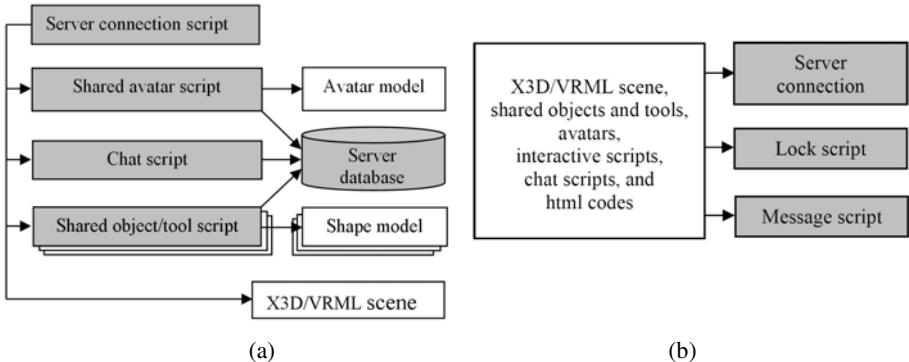
**Fig. 1.** Data flows in VRML/X3D-based interactive web applications

The data between the html code and VRML/X3D with scripts is exchanged both ways using the special functions of the java script language.

The data within VRML/X3D with scripts is exchanged through the standard mechanism of VRML/X3D data exchange such as event variables and exposed fields. A significant portion of the interactive software can be written using scripts and SDK functions provided by the respective VRML/X3D viewer. However this way is normally limited in the way how the applications GUI can be implemented and restricts computational complexity to the limits of scripts which are abridged java script codes.

The data between the VRML/X3D with scripts and Java or C++ functions stored in DLL libraries can be exchanged both ways. The functions in the binary libraries can take most of the computational load of the interactive application while still using the graphics pipeline implemented at the VRML/X3D level. If the application is designed in such a way that no further extensions is expected, all the data flows can be planned in advance and implemented through the function parameters and function values. However, if new components of the application can be developed in future, there is a challenging problem of exchanging the data between the binary functions stored in different DLLs which have been independently compiled without making references to each other. This can be done by setting up an exchange protocol in the RAM and running a special daemon process on each client computer which should start before any component of the application is loaded and continue running while the application is being executed. When any of the components needs to access the data, it will query the daemon. The daemon will then return a respective pointer to the application component. After that, the component will use the returned pointer as a local pointer for accessing the data it needs in the shared memory.

To support collaboration in VRML/X3D applications, we developed two frameworks based on the concepts of the strong and thin servers (Fig. 2). The strong server (Fig. 2.a) controls all the issues concerned with the synchronous visualization of



**Fig. 2.** Setting shared and collaborative features of the scene with the strong (a) and thin (b) servers used. Shaded blocks are provided by the server.

shared objects on each client computer including storage of the parameters of the objects being shared. The thin server (Fig. 2b) only provides locking and relaying messages between the client computers while the shared objects are implemented by exchanging models between the applications on the client computers.

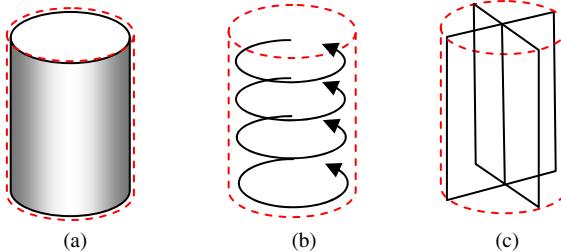
As an implementation platform we used a new pilot version of BS Collaborate server and BS Contact VRML/X3D viewer which were developed within the course of our project. These software tools work as a server-client pair to support visual and haptic collaboration in shared virtual spaces defined by X3D and VRML and, optionally, by their extensions such as the function-based extension that we use for defining physical properties of the virtual objects. BS Collaborate Server is a networked application which supports information transmission, shared events, shared objects and locking. Compared to other collaborative platforms, it does not impose restrictions on the shared scene, such as fixed file framework and window layout. It leaves the developers very much in control of most of the collaborative issues. BS Contact VRML/X3D viewer is a client for Microsoft Internet Explorer and Mozilla Firefox web browsers. It can make an X3D and VRML scene a part of an html page. The viewer also allows for performing haptic interaction with the X3D and VRML scene using one or several interactive and haptic devices.

### 3.2 Collaboration Based on the Strong Server

When this method is used, any VRML/X3D scene can be made a shared collaborative scene by adding to the scene root file a few script modules that set sharing and collaborative features and parameters (Fig. 2a). This approach allows the developers to control all aspects of the collaborative application while using all of the server features for supporting the collaboration.

BS collaborate allows for defining the viewer's avatar, text and text-to-speech (TTS) chat, as well as shared objects and tools associated with various interactive and haptic devices. Physics properties can be optionally added to shared objects and tools by using the function-based extension of VRML and X3D [11]. In this extension surface and solid haptic effects as well as ubiquities forces in the shared virtual scenes can be defined for mixed geometric models, including polygon meshes, point clouds,

image billboards and layered textures, voxel models and functions-based models of surfaces and solids. Each haptic effect has a certain geometric container within which this effect can be haptically rendered. Such a container may be a geometric surface of the object which is augmented with the haptic property (Fig. 3a). It can also be an invisible surface specially defined for the haptic effect and not necessarily coinciding with the geometric surface of the object (Fig. 3b and 3c).



**Fig. 3.** Haptic containers (dashed line): (a) Actual surface of the object, (b) Surface containing haptic forces, (c) Surface defining an object rendered without showing the actual surface (impostors, layered textures, point clouds)

This approach allows for creating haptic effects for objects which do not have any surface at all such as point clouds or objects displayed as layered textures (e.g. MRI images). This approach also allows for adding ubiquitous forces to the scenes by encapsulating them into the invisible haptic containers. For each of the haptic container, be it a real surface of the object or a specially defined surface for the haptic effect, it is allowed for concurrent definition of surface properties (tension and friction), solid properties (density), and a force field.

More advanced features allowing for user management can be implemented by using 3<sup>rd</sup> party web-servers, script generators and databases. Each of the shaded modules in Fig. 2a are scripts and prototypes which only require to add names (URLs) of the X3D/VRML files defining the actual avatars, prototypes of the shared objects and tools and their initial positions and orientations.

The core part of the server implementation is a specific data stream for transmitting objects and events in the shared environment between a network stream and the VRML/X3D event graph. All the values that are transmitted and stored on the server resemble the scene state. By incorporating this mechanism, any field values in an VRML/X3D scene can send and receive updates via a TCP/UDP socket or HTTP server.

Shared objects and tools have to be defined as they shape prototypes with several instances available concurrently in the shared environment. The exposed fields of the prototypes can be position, orientation, geometry, appearance, physical properties, or even the whole shape node. During the run time, shared events will be generated by different user interactions, such as movements, clicks or modification commands. These shared events are then received and interpreted as scripts which are sent into the shared object/tool module prototypes, triggering the corresponding field changes. By doing this, all instances of a certain shared object/tool prototype will be updated simultaneously. Since this procedure is broadcast by the server and identically

executed on all the clients, they all will be updated adequately. Besides, since only the update events are transmitted, the server will not be overloaded. We have tested the maximum possible shared events update rate by using haptic device interval as the trigger (1000HZ) which is beyond normal speed of user interaction. The result was quite acceptable for making interactive shared collaboration (less than a 1 sec delay).

In order to ensure that all new clients are displaying the current state of the scene when they connect, the server makes a back up of all the shared fields in a database to send them to the new clients when they log in.

Server-side locking is another important feature of the server. To allow for a server side locking mechanism, special fields have been added to the data stream to define different modes of server-side locking (strong, weak), as well as information of lock owner and lock states.

### 3.3 Collaboration Based on the Thin Server

This collaboration framework (Fig. 2b) expects very few server functions to be used for maintaining collaboration while relying on the clients to implement the rest of them. The server only provides locking mechanism and information transmission between the clients. The client-based software is responsible for sending to the clients through the server all the modified object models whenever any of such modifications occur. This approach makes the application software independent of the collaboration server used and allows the developers to easily implement the maximum sharing ability by exposing all shared fields to run-time changes: all properties of shared objects and tools can be dynamically changed and synchronized, including geometry, appearance, and physical properties. This method is particularly efficient when relatively small function-defined (procedural) models are used for defining shared virtual objects and their properties however we used it with standard VRML/X3D shapes as well.

All collaborative modifications are controlled by the clients. For each of the properties that are going to be shared, a special routing script has to be set up for monitoring events and recomposing output. When one client initiates an action, it will first request the lock from the server. After that, the interactions such as geometry modification and color modification will be received by the local routing script, and a corresponding output based on the client's status (e.g. object's location, orientation, etc.) will be composed. The interaction will cause the server to relay the updates which will be finally received and executed by all the clients. The server is not responsible for storing scene status. Instead, one of the users in the scene must be responsible for temporarily storing current status of the scene. When all the clients leave the scene, its status will be lost unless it is somehow stored manually before (e.g. 3D snapshot of the scene).

The main events in the collaborative framework are: *New client joins the modeling session; Model is modified; New model is loaded; Model has to be saved or exported for further use*. Models are exchanged between clients as messages therefore the most efficient communication can be achieved when the model sizes are small. This is what we can efficiently achieve by using function-defined models for the shared objects and tools.

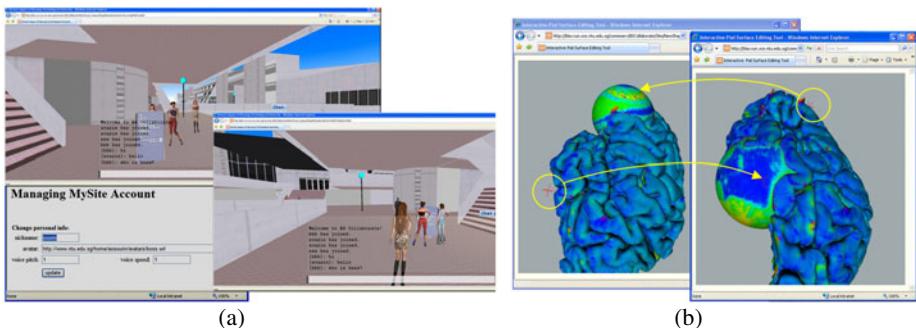
When a new client has been initialized, it first detects whether the joining session is a new session or an existing session. The primary client who joined the session first

will be responsible to send the current modeling object to new clients joining the same shared scene. When the primary client left the session, his duty will be assigned to one of the available clients. When the scene is modified by any of the clients, the modifications must propagate to all other clients and update the scene as they see it on their computers. When an object modification is being done by any of the clients, it is then converted into a message and sent through the server to all the clients participating in the collaborative session. When a new scene is loaded or reset by any of the clients, the new scene will be sent as a message to all the clients participating in the design session. Finally, the users should be able to save the design at any time on their client computers since the server is not responsible for storing such information as in the case of the strong server. This can be done as simple as displaying the scene code in the console window and saving it in a file.

To set up a shared virtual space the developers have to use a template of the core VRML or X3D file which establishes the link with the server, defines the login and chat modules and finally links to the VRML or X3D codes of the shared space.

### 3.4 Examples of Collaborative Scenes

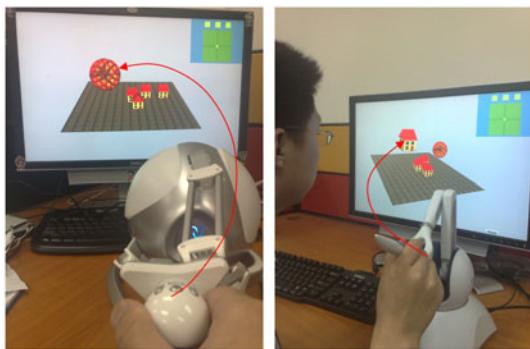
A very typical application of collaborative software for setting up a shared virtual space is illustrated in Fig. 4a. Here, the thin server allows each user to maintain a personal database with the parameters of avatars and login information and provides the text and text-to-speech chat option.



**Fig. 4.** Shared virtual scenes set with the thin server

Another example of using the thin server within the project on interactive segmentation of brain MRI data [8] is illustrated in Fig. 4b. Here, the model of a human brain is reconstructed from the MRI data and collaboratively explored and edited by several networked users. The surface of the brain is declared tangible to feel it with the haptic device. The collaborative users see each other's sampling tool as a crosshair while their own sampling tool is displayed as a sphere with dynamically changing colors mapped from the brain MRI data. The users are also able to see how the surface of the brain is changing while being edited by all the participating parties. The colors on the surface indicate the probability of segmentation error. There are no avatars of the users displayed as they will have no meaning and be rather obstacles. The users are able to communicate via text messages.

An example of using the strong server is illustrated in Fig. 5. Here, two networked users are working in a shared collaborative scene with several virtual shared objects defined in it. Any of the objects can be assigned to be a tool by linking from the shared object modules of the root file (Fig. 2a), it will then move following the motion of the respective device and both users will synchronously see its motion. Each user can have one or several tools concurrently which have to be associated with the respective interactive or haptic devices. If two users with haptic devices share the same tool object, it will result in a synchronized motion of the networked haptic devices and coordinated motion of the tool on the computer screens. The networked users then will be able to physically feel each other's presence and motion.



**Fig. 5.** Visual and haptic collaboration with the strong server

## 4 Conclusion

We have proposed a visual and haptic collaborative framework for shared VRML and X3D scenes. We define virtual objects with physical properties, which can be explored haptically with various force-feedback devices. Virtual objects can be declared as shared objects which visual and physical properties are rendered synchronously on each client computer. We introduce virtual tools which are shared objects associated with interactive and haptic devices. The proposed framework can be used for making shared collaborative environments with both standard VRML and X3D as well as their extensions. The proposed ideas have been implemented in new versions of Bit-management BS Collaborate server and BS Contact VRML/X3D viewer. We developed two implementation frameworks based on the strong and thin server concepts. Video clips illustrating some of the experiments with the developed software are available at <http://www.ntu.edu.sg/home/assourin/fvrmr/>.

## Acknowledgment

This project is supported by the Singapore Ministry of Education Teaching Excellence Fund Grant “Cyber-learning with Cyber-instructors”, by the Singapore National Research Foundation Interactive Digital Media R&D Program, under research Grant

NRF2008IDM-IDM004-002 “Visual and Haptic Rendering in Co-Space”, and partially by the Singapore Bioimaging Consortium Innovative Grant RP C-012/2006 “Improving Measurement Accuracy of Magnetic Resonance Brain Images to Support Change Detection in Large Cohort Studies”.

## References

1. Davies, R.A., John, N.W., MacDonald, J.N., Hughes, K.H.: Visualization of molecular quantum dynamics: a molecular visualization tool with integrated Web3D and haptics. In: Proc. 10th Int. Conf. on 3D Web Technology, pp. 143–150 (2005)
2. Eid, M., Alamri, A., El Saddik, A.: MPEG-7 description of haptic applications using HAML. In: Proc. IEEE Int. Workshop on Haptic Audio Visual Environments and their Applications, pp. 134–139 (2006)
3. Eid, M., Andrews, S., Alamri, A., El Saddik, A.: HAMLAT: A HAML-based authoring tool for haptic application development. In: Ferre, M. (ed.) EuroHaptics 2008. LNCS, vol. 5024, pp. 857–866. Springer, Heidelberg (2008)
4. El-Far, F.R., Eid, M., Orozco, M., El Saddik, A.: Haptic applications meta-language. In: Proc. 10th IEEE Int. Symp. on Distributed Simulation and Real-Time Applications, pp. 261–264 (2006)
5. Hamza-Lup, F.G., Sopin, I.: Haptics and extensible 3D in web-based environments for e-learning and simulation. In: Proc. 4th Int. Conf. on Web Information Systems and Technologies, pp. 309–315 (2008)
6. John, N.W.: The impact of Web3D technologies on medical education and training. Computers & Education 49(1), 19–31 (2007)
7. Jung, Y., Recker, R., Olbrich, M., Bockholt, U.: Using X3D for medical training simulations. In: Proc. 13th Int. Symp. on 3D Web Technology, pp. 43–51 (2008)
8. Levinski, K., Sourin, A., Zagorodnov, V.: Interactive Surface-guided Segmentation of Brain MRI Data. Computer in Biology and Medicine 39(12), 1153–1160 (2009)
9. Lundin, K., Persson, A., Evestedt, D., Ynnerman, A.: Enabling design and interactive selection of haptic modes. Virtual Reality 11(1), 1–13 (2006)
10. Magnusson, C., Tan, C., Yu, W.: Haptic access to 3D objects on the web. In: Proc. EuroHaptics (2006)
11. Wei, L., Sourin, A., Sourina, O.: Function-based visualization and haptic rendering in shared virtual spaces. Vis. Comput. 24(10), 871–880 (2008)

# Design and Costs Estimation of Electrical Substations Based on Three-Dimensional Building Blocks

Eduardo Islas Pérez, Jessica Bahena Rada,  
Jesus Romero Lima, and Mirna Molina Marín

Instituto de Investigaciones Eléctricas, Av. Reforma 113 Col Reforma,  
Cuernavaca, Morelos, México, 62490  
[{eislas,jesbahrad,jesus.romero,mmolina}@iie.org.mx](mailto:{eislas,jesbahrad,jesus.romero,mmolina}@iie.org.mx)

**Abstract.** Substations design is a fundamental engineering component in power network construction. The benefits obtained for having adequate tools and design systems are related mainly to cost savings, reduction of construction problems and faster throughput of projects. In this paper we propose an approach based on three dimensional building blocks to construct virtual substations. The building blocks can become 3D standards for advanced engineering, automated drawing, data extraction and reusability of designs. Therefore these substation designs can improve quality and reliability of the design process. With virtual substations we can use them to help on making decisions about construction site selection and community and government acceptance. Finally 3D visualization and walkthrough can be used to improve construction, commissioning, operations and maintenance of distribution and transmission electrical substations.

**Keywords:** Building Blocks, 3D Environments, Electrical Substations Design, CAD Tools.

## 1 Introduction

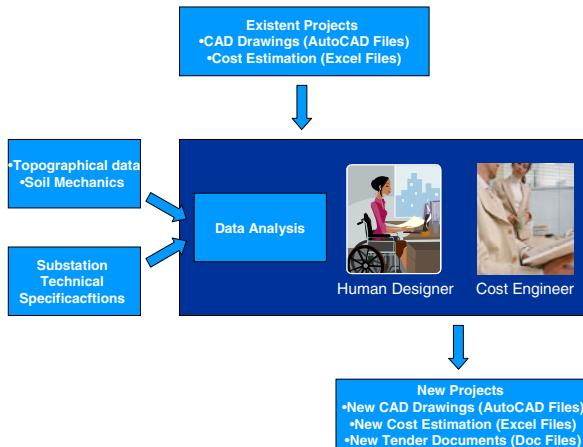
In this paper a system for designing electrical distribution substations (SIDSED) is described. The system uses different levels of building blocks to ease the design process and facilitate the estimation of costs of new electrical substations. The building blocks are based on three levels of abstraction; buildings blocks in the highest level are composed from building blocks in lower levels. Each building block has an associated cost obtained from a concepts catalogue with unit prices. The system was developed for CFE (Electricity Federal Commission) which is the main utility for generation and distribution of electricity in Mexico.

Using different levels of abstraction allow human designers the flexibility to devise a substation design using different levels of complexity and plasticity. For instance, if it is needed to change radically the design of a new substation an engineer can use basic elements from lower levels of abstraction (transformers, high-voltage circuit-breakers, lightning rods, structures, foundations, ducts banks, etc). Also, if a new design of a substation is required quickly, the designer may use modules, which is a group of elements in superior levels of abstraction (line bay, transformer bay, control room, edge wall, etc). Once the configuration of a substation has been completed,

human designers can estimate the cost for that substation and they are able to make decisions about the involved costs, type and size of equipments and future developments for that electrical substation.

## 2 Traditional Design for New Substations

Figure 1 depicts a flow diagram about the current process for designing a new electrical substation. Human designer bases new designs on information and data from developed projects such as: topographical data, old CAD drawings and budget data to generate new CAD drawings, costs estimation and tender documents for a new substation.



**Fig. 1.** Flow diagram for traditional substations design at CFE

## 3 System Requirements

CFE asked for several requirements to meet the needs of 13 distribution divisions. The requirements for SIDSED were established taking into account that it will be used in all CFE's Divisions across the country. The main solicited requirements were:

- R1. The system must consider the whole design process for 115-KV substations.
- R2. The system must be able to design 3D normalized arrangements (H, ring, main bus and main-bus transfer-bus).
- R3. The system must have interoperability with costs engineering software in order to obtain the costs for building substations.
- R4. The system must have 3D visualization and walkthrough to give information about equipments (transformers, high-voltage circuit-breakers, etc.), building elements (foundations, ducts banks, walls, etc.) and verify security distances between components.

- R5. The system must generate drawings, concepts catalogue and costs estimation documents.
- R6. The system must generate a visualization file to share the substation with personnel without the necessity of having the system or any software used for development.

## 4 Related Work

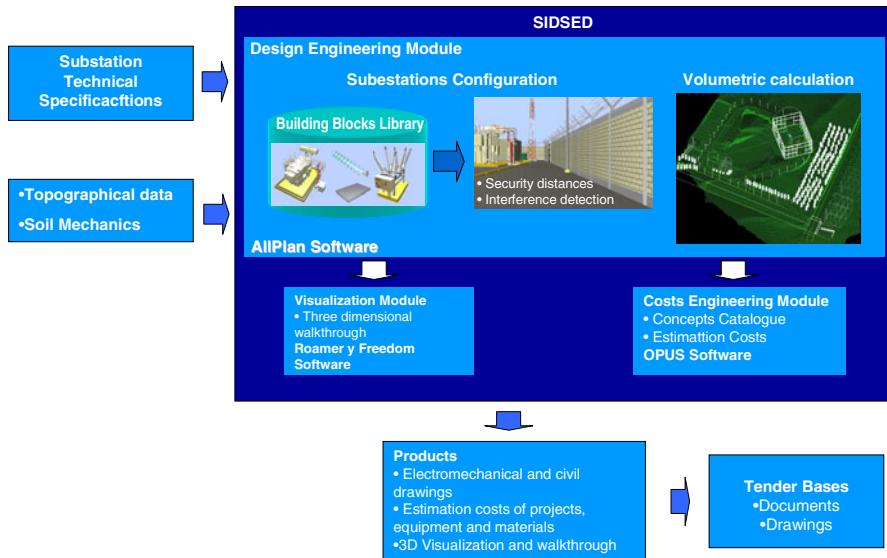
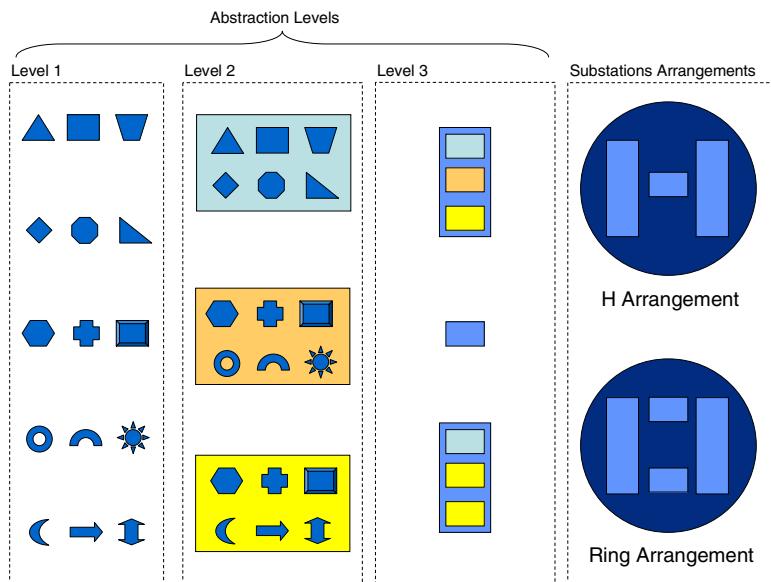
Substations design is a fundamental engineering design component in power network construction. Puget and Enriquez dealt with several concepts related with good practices for substations design [1] [2]. However, due to complexity in designing an electrical substation the process can be tedious and time consuming; therefore the design time can be prohibitively long. In recent times, with the aid of two dimensional CAD systems, substation's design time has been reduced considerably. During these years, other industries have adopted three dimensional modeling for example: process plant design, mechanical design, manufacturing, architecture, civil engineering, medicine, etc. There are a lot of software packages, hardware tools and information related to 3D and virtual environments [3]. The benefits of 3D design and modeling in these industries are well documented [4]. Reduction of change orders, concurrent engineering designs, elimination of interferences, increased quality of design, accurate material requirements, faster project throughput, visualization and construction sequencing have all shown quantifiable benefits [5].

Aberden [6] describes in a full study the benefits obtained with the migration from 2D to 3D, mainly reducing the change of orders and reusing existing parts and morphing an existing part into a new one. With respect to reusing parts in the design process, there are recent works which discuss problems involved in design reuse [7] [8]. However, most studies on the topic of reuse have predominately dealt with problems involving computer software and architectural reuse [9], [10], [11], [12]. Griss and Nazareth have shown that reusing design components is useful to reduce costs and shrink development time [13] [14]. In our approach, we reuse electrical and civil components as building blocks in order to ease the design process. Additionally the system estimates the costs for building new electrical substations.

## 5 System Description

In order to accomplish with the specified requirements, SiDSED was developed having three modules: engineering design module, costs engineering module and visualization module (see Figure 2).

In the engineering design module, human designers use the building blocks to design a new electrical substation, taking into account some topographical data. After that, the cost of an electrical substation is estimated with the costs engineering module. Finally, with 3D visualization and walkthrough module, designers can make decisions about aspects related with construction, operations, maintenance and training.

**Fig. 2.** SIDSED Configuration**Fig. 3.** Schematic diagram at different abstraction levels

## 5.1 Three-Dimensional Building Blocks Scheme

The 3D building blocks library was developed using a CAD software tool, which was selected from several software tools options to fulfill the established requirements for the system [15]. The 3D models were associated at different levels of abstraction and stored in a library embedded in the design engineering module.

The schematic diagram of building blocks at different abstraction levels is shown in Figure 3. It can be seen from the figure that each building block at level 1 is a basic element. In level 2 there are building blocks formed by elements from level 1, which are represented by horizontal rectangles. In the third level of abstraction, the horizontal rectangles are put together in order to have building blocks represented by vertical rectangles, which finally are used to design the physical arrangements of substations (H, ring, main bus and main bus-transfer bus). It is important to mention that each building block has associated its costs in such a way we can obtain the costs of the whole substation.

# 6 System Modules

## 6.1 Design Engineering Module

This module is divided in two processes: volumetric calculation and substations configuration. For volumetric calculation the soil movement is estimated in order to create terraces, levels and profiles in the construction site and estimate the associated costs depending on several factors (terrain type, soil volume, unit costs, etc.). The objective of the substations configuration process is to build normalized substations including bays, control rooms, edge wall, ducts banks, etc. With the final design of the substation, the civil and engineering CAD drawings can be generated; a list of all elements of the substation are generated and linked with their costs. With these information can be generated the documents related with tender bases. The updating of CAD drawings and costs estimation will be made automatically when the substation design is changed, avoiding the time consuming activities to maintain all information updated.

## 6.2 Costs Engineering Module

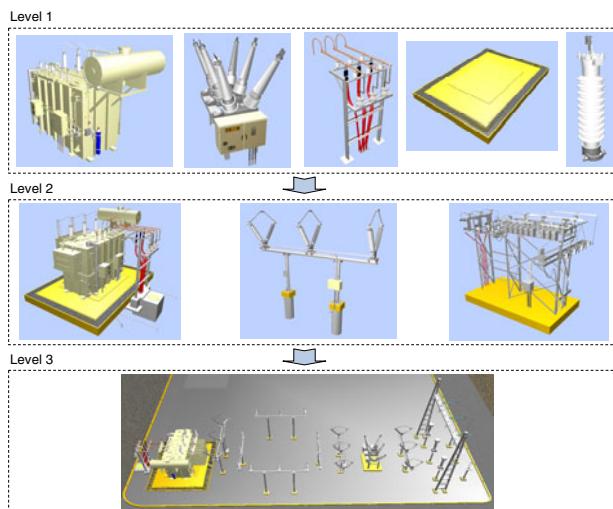
In this module the list of all elements generated in the design module are linked automatically to estimate the costs for building that substation. The associated costs for each building block are stored in a catalogue of concepts which describes the technical specifications and costs based on their unit prices.

## 6.3 Visualization Module

This module consists of visualization and walkthrough around the virtual electrical substation in order to make decisions about layout design, types and sizes of equipments and constructive elements. The main objectives of this module are: to detect and avoid interferences between elements and verify security distances between components. It is also helpful to show the design to be approved by other government sector.

#### 6.4 Three Dimensional Building Blocks Library

For creating the 3D library we build and grouped together 3D elements at different levels of abstraction. As we can see in Figure 4, the first level consists of building blocks of basic elements, such as transformers, high-voltage circuit-breakers, lightning rods, structures, foundations, duct banks, etc. (see Figure 4, level 1). Each of these elements has associated its unit price through a link with the concepts catalogue. In the second level are the building blocks that are formed by elements of the first level, for instance, the transformer-perch building block is composed by a transformer, a transition perch, foundations, and groundings. The H structure with disconnect blades is formed by an H structure, blades, a motoperator, foundations and grounding (see Figure 4, level 2). Finally, the third level is formed by building blocks in the most superior abstraction level, for example: line bay, transformer bay, control room, edge wall, etc (see Figure 4, level 3). This 3<sup>rd</sup> level is the most used abstraction level in the design process because it has all the necessary equipments for a specific function. For example, the control room has all equipments and materials inside of a standardized control room. If the designer needs to design a novel control room then he/she needs to use the 1<sup>st</sup> and 2<sup>nd</sup> abstraction level. It is very important to mention that each building block in the three abstraction levels has associated its unit price in such a way that at the end, the costs estimation for the whole substation is obtained.

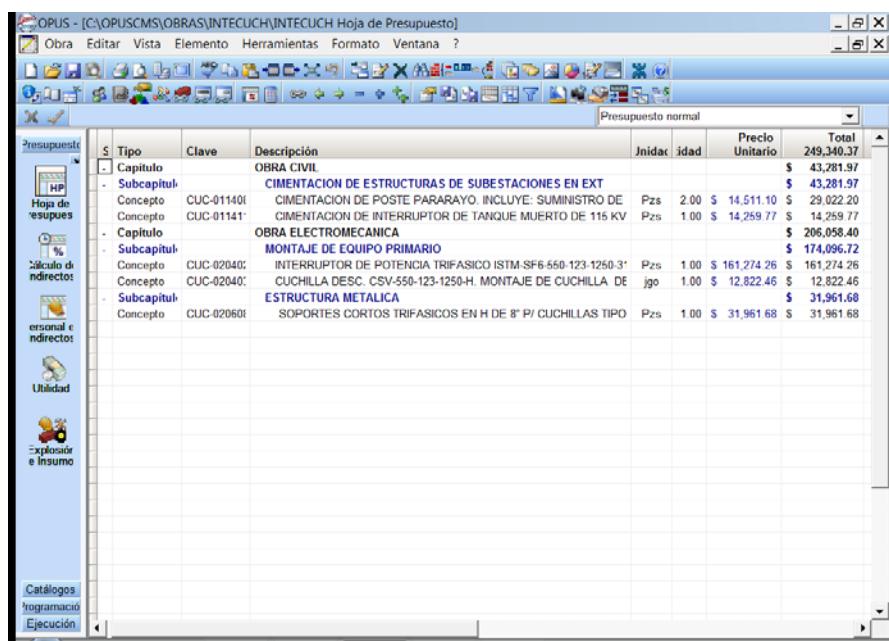


**Fig. 4.** Building Blocks at different levels of abstraction

Because building blocks were obtained from experienced human designers; these were developed in such a way that these can be used in different types of arrangements (ring, transfer bus, H, etc.). Therefore using these building blocks, the engineer can design a great variety of electrical substations very quickly.

## 7 Costs Estimation and Design of Electrical Substations

SIDSED is being used for development and costs estimation of new substations at CFE. To estimate the associated costs to build a new substation each building block has associated unit prices from a concepts catalogue. By designing a new substation based on those building blocks we will know a priori the total cost for that substation. As an example in Figure 5 is shown the costs estimation of a building block in the 3<sup>rd</sup> level integrated by 2 building blocks in the 2<sup>nd</sup> level of abstraction. In this example the cost involves an H structure, a high-voltage circuit-breaker, blades, motoperator, foundations and grounding. The cost of this building block is estimated in almost 250,000 pesos.

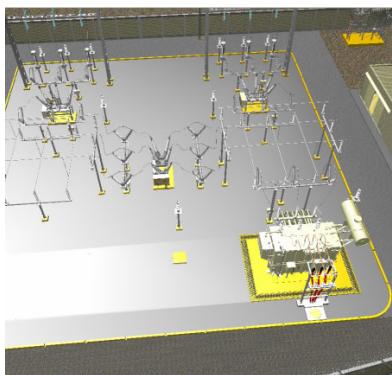


The screenshot shows a software application window titled 'OPUS - [C:\OPUSCMS\OBRA\INTECUCH\INTECUCH Hoja de Presupuesto]'. The menu bar includes 'Obra', 'Editar', 'Vista', 'Elemento', 'Herramientas', 'Formato', 'Ventana', and '?'. The toolbar contains various icons for file operations, selection, and data entry. On the left, a vertical sidebar lists categories: 'Presupuesto' (with 'HP' icon), 'Hoja de presupuesto' (with 'HP' icon), 'Álculo de indirectos' (with '%' icon), 'Personal e indirectos' (with person icon), 'Utilidad' (with wrench icon), and 'Catálogos programación Ejecución' (with gear and document icons). The main area displays a budget table titled 'Presupuesto normal' with the following data:

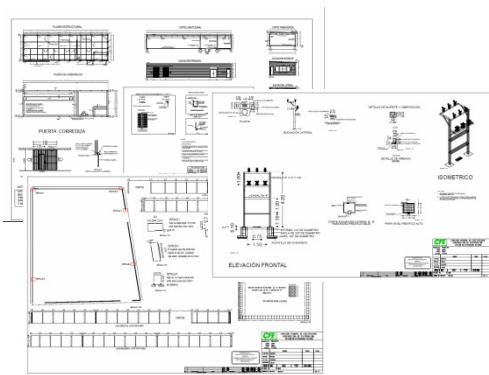
	Tipo	Clave	Descripción	Unidad	Cantidad	Precio Unitario	Total
- Capítulo			OBRA CIVIL				\$ 43,281.97
- Subcapítulo			CIMENTACION DE ESTRUCTURAS DE SUBESTACIONES EN EXT				\$ 43,281.97
Concepto	CUC-011401		CIMENTACION DE POSTE PARARAYO. INCLUYE: SUMINISTRO DE	Pzs	2.00	\$ 14,511.10	\$ 29,022.20
Concepto	CUC-011411		CIMENTACION DE INTERRUPTOR DE TANQUE MUERTO DE 115 KV	Pzs	1.00	\$ 14,259.77	\$ 14,259.77
- Capítulo			OBRA ELECTROMECANICA				\$ 206,058.40
- Subcapítulo			MONTAJE DE EQUIPO PRIMARIO				\$ 174,096.72
Concepto	CUC-020401		INTERRUPTOR DE POTENCIA TRIFASICO ISTM-SF6-550-123-1250-3'	Pzs	1.00	\$ 161,274.26	\$ 161,274.26
Concepto	CUC-020402		UCHILLA DESC. CSV-550-123-1250-H. MONTAJE DE UCHILLA DE	jgo	1.00	\$ 12,822.46	\$ 12,822.46
- Subcapítulo			ESTRUCTURA METALICA				\$ 31,961.68
Concepto	CUC-020601		SOPORTES CORTOS TRIFASICOS EN H DE 8' P/ UCHILLAS TIPO	Pzs	1.00	\$ 31,961.68	\$ 31,961.68

**Fig. 5.** Estimation Costs for a building block in the 3<sup>rd</sup> level

In Figure 6 a) and b) are shown images and drawings from the Valle Verde substation (i.e. top view, edge wall, control room and transition perch). Valle Verde substation is configured in an H arrangement. Figure 7 shows images from La Reina a main-bus substation arrangement. Figure 8 shows La Diana, an encapsulated GIS substation, this substation is being designed in three levels in a completely novel way because it is located in a very populated area in Mexico City and it has spatial restrictions requirements. The system will help to make decisions about the convenience of each substation and which is the best equipment distribution and configuration based mainly on their estimated costs among other criteria such as: 3D interferences, additional future bays, etc.

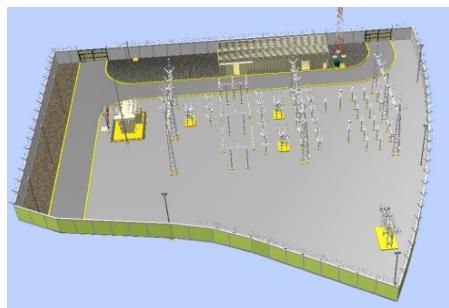


#### a) 3D Design

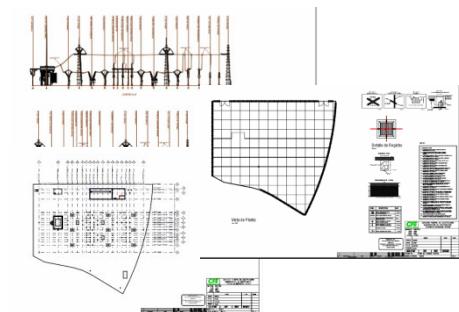


b) CAD Drawings

**Fig. 6.** Valle Verde Substation

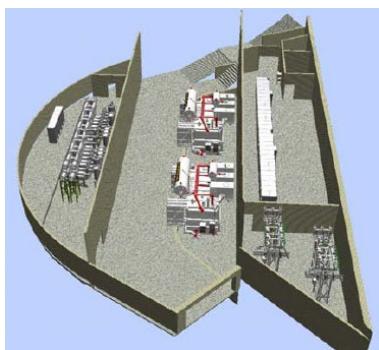


### a) 3D Design

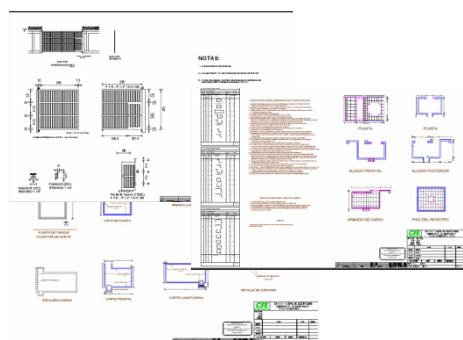


b) CAD Drawings

**Fig. 7.** La Reina Substation



### a) 3D Design



b) CAD Drawings

**Fig. 8.** La Diana Substation

## 8 Results

The results obtained with this approach is a library of 120 building blocks in the first level of abstraction, 30 elements in the second level and 15 in the most superior and complex level of abstraction. Additionally human designers in Distribution Department are working in the development of electrical distribution substations based on different arrangements (main bus, H, ring and main bus-transfer bus) used at CFE. With the building blocks library human designers will be able to design other normalized arrangements as well as generating fully new substations configuration.

Once the 3D virtual substation is designed the CFE's Divisions obtain the following benefits:

- Volumetric calculation for soil movement.
- The civil and electromechanical CAD drawings.
- The costs estimation of substations based on the concepts catalogue.
- A three-dimensional visualization to navigate around the substation and to take 3D measurements in order to verify security distances between components.
- Costs estimation documents to help in the elaboration of tender documents.
- Visualization files to review the completeness and exactness of substations.

Some of the benefits obtained with SIDSED are the following:

- **Reduction in time.** In a preliminary study the first group of designers is taking in average 75% less time to design a new substation.
- **Cost savings.** The cost savings for CFE is related with reducing the change of orders or modify a substation design when is it in the construction process.
- **Standardization.** Improving the standardization process of substations through the use of a standardized building blocks library besides using a unique concepts catalogue with standardized unit prices.
- **Design sharing to review.** Facilitate the process of reviewing and project acceptance.

## 9 Conclusions and Future Work

Some of the benefits obtained with the use of this type of approaches are related mainly with cost savings through design automation, reduction of construction problems and faster throughput of projects. The building blocks can become 3D standards for advanced engineering, automated drawing, data extraction and reusability of designs. With virtual substations we can use it for site selection, community and government acceptance and 3D visualization and walkthrough can be used to improve construction, commissioning, operations and maintenance.

Nowadays we are training personnel and implementing SIDSED in the first three Divisions at CFE and in a near future it will be implemented and used in all CFE's Divisions along the country.

## References

1. Puget, S.: Substation Work Practices Publications. Puget Sound Energy (759), Newport Beach, CA, USA (1999)
2. Enríquez, G.: Elementos de diseño de subestaciones eléctricas, segunda edición, Limusa, Mexico (2002)
3. Burdea, G., Coiffet, P.: Virtual Reality Technology, 2nd edn. Wiley-Interscience, Hoboken (2003)
4. Lambert, M.: Designing Substations in 3D. DistribuTECH, San Diego CA, USA (2005)
5. Romero, G., Maroto, J., Félez, J., Cabanellas, J.M., Martínez, M.L., Carretero, A.: Virtual Reality Applied to a Full Simulator of Electrical SubStations. Electric Power Systems Research 78(3), 409–417 (2008)
6. Aberdeen Group. The Transition from 2D Drafting to 3D Modeling Benchmark Report. Aberdeen Group, Inc. (2006)
7. Ball, L.J., Lambell, N.J., Ormerod, T.C., Slavin, S., Mariani, J.: Representing Design Rationale to Support Innovative Design Reuse: A Minimalist Approach. Automation in Construction, 663–674 (2001)
8. Busby, J.S.: The Problem with Design Reuse: An Investigation into Outcomes and Antecedents. Journal of Engineering Design, 277–296 (1999)
9. Frakes, W.B., Kang, K.: Software Reuse Research: Status and Future. IEEE Transactions of Software Engineering (2005)
10. Rothenberger, M.A.: Project-Level Reuse Factors: Drivers for Variation within Software Development Environments. Decision Sciences, 83–107 (2003)
11. Sherif, K., Appan, R., Lin, Z.: Resources and Incentives for the Adoption of Systematic Software Reuse. International Journal of Information Management (2006)
12. Van Ommering, R.: Software Reuse in Product Populations. IEEE Transactions of Software Engineering (2005)
13. Griss, M., Jacobson, I., Jonsson, P.: Software Reuse: Architecture, Process, and Organization for Business Success. ACM Press, New York (1997)
14. Nazareth, D.L., Rothenberger, M.A.: Assessing the Cost Effectiveness of Software Reuse: A Model for Planned Reuse. Journal of Systems and Software, 245–255 (2004)
15. Bahena, J., Zayas, B., Islas, E., Molina, M.: Evaluación de herramientas CAD para el diseño de Subestaciones Eléctricas de Distribución. Internal Report (2008)

# Generating Shaded Image with Lighting Using Image Fusion Space

Satoru Morita

Faculty of Engineering, Yamaguchi University

**Abstract.** We generate an image with many lighting equipments by fusing images with a few lighting equipments. In general, the principle of the superposition is discussed for the lighting. An image with two lighting equipments is generated by adding the pixel values of two images with a lighting equipment. But it is difficult to generate the image by adding the pixel values of two images as the internal parameter changes in the case of using a single-lens reflex camera with the sensitivity automatic control facilities. On the other hand, it is difficult to generate the image using traditional fusion methods. We introduce the image fusion space describing the relationship between traditional fusion methods. We propose the method selecting the fusion method for each pixel from the image fusion space. The fusion image is generated using the fusion methods with the parameter suited in the divided regions by searching image fusion space. We show the effectiveness by generating the fusion images with lighting equipments in the general environment.

## 1 Introduction

Recently, studies have investigated the deletion and addition of objects in a photo [1] [2] [3]. There are important techniques to generate a entertainment movie at a low cost. It is important that the lighting in an image derived by the composite images is realistic and natural. In this paper, we generate the fusion image including the shaded features on two images by fusing the images with the light that is different position from the light position of another image.

Recently image fusion methods are studied [4] [5] [6]. A multifocus image is generated by fusing pictures with different focal point lengths [7]. Zhang and Blum generated the high quality image by fusing low quality images [8]. Ranchin and Wald generated a image by fusing remote sensing images with different features [9]. The method deciding the Multi-resolution Signal Decomposition (MSD) coefficient of the fusion image is discussed [10] [11] [12]. The Coefficient Based Activity (CBA) method using MSD coefficient to decide the activity level, the Window Based Activity (WBA) method using MSD coefficient in the window, the Weighted Average - Window Based Activity (WA-WBA) method using the weighted average in the window, the Ranking - Window Based Activity (RA-WBA) method using the rank in the window are discussed [10]. CM method selecting the image that the activity level is big, the method using the average of MSD coefficient without relating to the activity level are discussed [10].



**Fig. 1.** The image set A is an image of figure (a) and an image of figure (b). The image set B is an image of figure (c) and an image of figure (d). The image set C is an image of figure (e) and an image of figure (f).

An image with all lighting equipments included in two photographs is generated from two photographs taken with a single-lens reflex camera with the sensitivity automatic control facilities. In general, the principle of the superposition is discussed for the lighting [13] [14] [15]. An image with two lighting equipments is generated by adding the pixel values of two images with a lighting equipment. But it is difficult to generate the image by adding the pixel values of two images as the internal parameter changes in the case of using a single-lens reflex camera with the sensitivity automatic control facilities. On the other hand, it is difficult to generate the image with two lighting equipments from two images with different lighting equipment using traditional image fusion methods [10]. In this paper, we propose the image fusion space describing the relation between neighbor image fusion methods. We generate the images with lighting equipments by deciding the fusion method for a pixel using the image fusion space. We identify whether the image with a front light or the image with a back light by defining two thresholds and classifying three regions from the histogram. The activity level is defined from the intensity image. We generate the shaded image with several lights by selecting the fusion methods according to regions for intensity, saturation and hue images using this activity level.

## 2 Image Fusion

### 2.1 Fusion Method and Activity Level Measurement

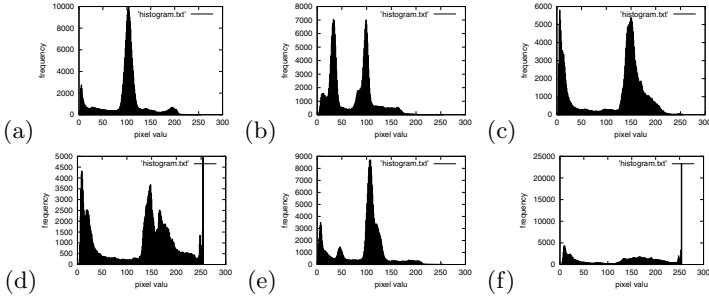
We use image intensity as the Coefficient-Based Activity (CBA) method in this paper. The image intensity is defined as equation (1) using the activity level in the following;

$$Iz(p) = \frac{Ax(p)Ix(p) + Ay(p)Iy(p)}{Ax(p) + Ay(p)}. \quad (1)$$

The equation (1) is extended to equation (2) using a fusion parameter  $q$  and  $K$  in the following;

$$Iz(p) = K \frac{Ax(p)^q Ix(p) + Ay(p)^q Iy(p)}{Ax(p)^q + Ay(p)^q}. \quad (2)$$

where  $K = 2^l$ .



**Fig. 2.** (a), (b), (c), (d), (e) and (f) is the histogram for the images of figure 1(a), (b), (c), (d), (e) and (f)

$Ii(p)$  is the pixel value of image  $i$  at the pixel position  $p$  and  $Ai(p)$  is the activity level of image  $i$  at the pixel position  $p$ .  $Ai(p) = |Di(p)|$ ,  $Di(p)$  is the wavelet coefficient, the wavelet level of 0 is required for the photograph texture. That is, we use  $Ai(p) = |Ii(p)|$ .  $p = (m, n)$  is the position of the image.  $m$  is  $x$  coordinate,  $n$  is  $y$  coordinate. The value  $q$  is a fusion parameter, and it is from  $-\infty$  to  $\infty$ . In the case of  $l = 0.0, q = 0.0$ , two values is added in a pixel. In the case of  $l = 1.0, q = 0.0$ , two values is averaged in a pixel. In the case of  $l = 1.0, q = 1.0$ , the rate of the activity level is used in a pixel. In the case of  $q = \infty$ , the maximum value pixel is chosen in a pixel. In the case of  $q = -\infty$ , the minimum value pixel is chosen in a pixel. We can put the methods to choose the maximum value pixel and the minimum value, the method to average pixel values, and the method to use the rate of the activity level on an fusion parameter  $q$  axis. We can know the methods with the various fusion parameter  $q$  except the observed fusion parameter  $q$  by observing the method with the limited fusion parameter  $q$ .

## 2.2 Region Partition by Histogram

As two original images are compared with the reference image, the shaded feature is strongly affected in a part, and the shaded feature is not affected in another part. The background and shadow can be simply classified in an image. At first, we classify the region using the threshold of the histogram from intensity image. Threshold  $t_1$  and threshold  $t_2$  divide into the region a, the region b and the region c. The region a is the region having the value that is smaller than threshold  $t_1$ .

**Table 1.** The method defining whether a front light or a back light using a histogram.  $h1$  is the maximum value of the frequency of region  $a$ .  $h2$  is the maximum value of the frequency of region  $b$ .  $h3$  is the maximum value of the frequency of region  $c$ .

	$h1 > h2$	$h1 < h2$
$h2 > h3$	back light	front light
$h2 < h3$	back light	back light

The region c is the region having the value that is bigger than threshold  $t_1$ . The region b is between the region a and the region c. where  $t_1 < t_2$ . If the number of the global mountains is three, the extrema points on two edges of a center mountain are  $t_1$  and  $t_2$ .

### 2.3 Estimating Front Light and Back Light Using Histogram

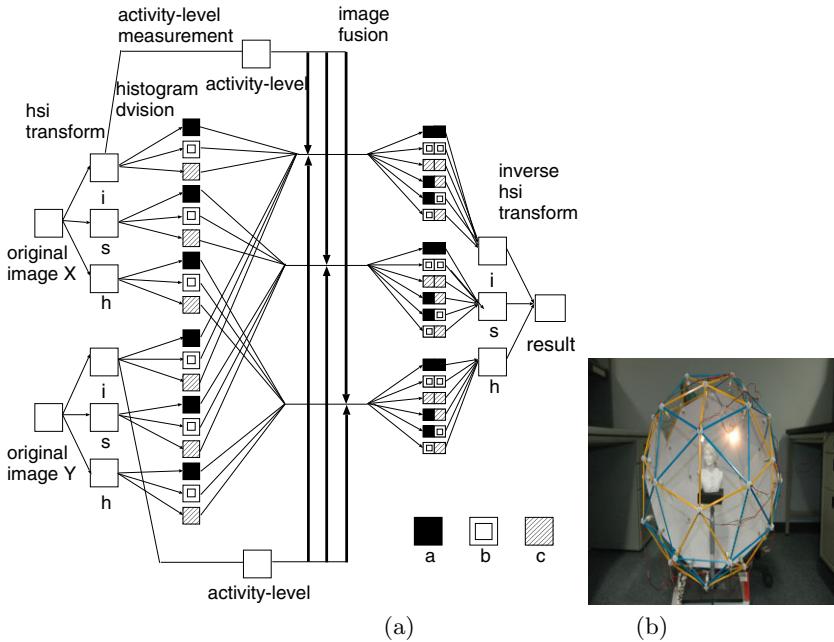
An image with a front light or an image with a back light are estimated using intensity of the original two images used for the image fusion. The maximum  $h1$ ,  $h2$  and  $h3$  of the frequency on the three region a, b and c in a histogram is calculated. If the maximum value  $h2$  of the region b is bigger than the maximum of the other region, or it is comparatively near the value, we define that it is the front light. If the maximum value  $h1$  of the region a or the maximum value  $h3$  of the region c is bigger than the other region, we define that it is the back light. If  $h2$  is smaller than the 0.665 time of  $h1$ , we regard that  $h1$  is enough big to  $h2$  in this paper. Whether the image with a front light or the image with a back light are defined using the methods corresponding to the combination of  $h1$  and  $h2$  empirically. Table 1 shows that the method defining the front light and the back light. The rate of definition rate whether the front light of the back light is 96.29% for the 54 individual photos.

## 3 Generating Shaded Image Using Image Fusion

Figure 3 shows the flow of the image fusion. The two original images are translated into intensity image, saturation image and hue image using HSI transformation of a hexagonal pyramid model.

We divide the histogram of an intensity image into three regions using two thresholds. The saturation image and the hue image are divided into regions corresponding to the partition of the intensity image. The six regions of aa, bb, cc, ab, ac and bc are classified by the combination selecting two region from a, b and c regions. We show the flow of the image fusion in figure 3(a).

Figure 3(b) shows the experimental equipment. The polygons such as the dome are structured using sixteen equilateral triangles and sixty isosceles triangles. Forty two miniature bulbs of 4.5 voltage are fixed at polygon vertex . An object is put in the center of the doom, and a photograph is taken from the outside of the dome. We use a single-lens reflex camera with the sensitivity automatic control facilities which is D70 produced by NICON corporation. The



**Fig. 3.** (a) The flow of the image fusion. (b)Experiment equipment.

image size is 512 times 512, the aperture is F29, and the shutter speed is 2''. When an object is viewed from the camera position, we regard it as the front light if an object is in front of the miniature bulb, and we regard it as the back light if an object is behind the miniature bulb. The images by fusing two photographs taken using the front light, the images derived by fusing two photographs taken with the back light and the images derived by fusing two photographs taken with the back light and the front light are used in the experiment. There are the set of two original images and the reference image. The reference image is taken using the lights used for the two original images. Three light positions and three objects are used for a combination of two front lights, a combination of a front light and a back light and a combination of two back lights in the experiment. We use photographs of eighty-one calculated using  $3 \times 3 \times 3 \times 3$ .

Figure 1 and figure 2 show the original photograph and the histogram respectively. In this paper, figure 1(a) and (b) is called the image set A, figure 1(c) and (d) is called the image set B, and figure 1(e) and (f) is called the image set C. From the experimental equipment, figure 1(a), (b) and (e) are the images with a front light and figure 1(c), (d) and (f) are the images with a back light. The thresholds of the histogram for figure 2(a) are 63 and 128 and the thresholds of the histogram for figure 2(b) are 79 and 125. The thresholds of the histogram for figure 2(c) are 121 and 219 and the thresholds of the histogram for the image of figure 2(d) are 120 and 239. The thresholds of the histogram for figure 2(e) are 87 and 140 and the thresholds of the histogram for figure 2(f) are 105 and 232.

The right result defined from the experimental equipment is equal to the result by the proposed algorithm based on histogram.

### 3.1 Image Fusion Space

To evaluate the intensity values of the fusion image, we evaluate the sum of minimum error between the two image and the fusion image in the following;

$$E1 = \frac{1}{N(S) \cdot L} \sum_{i=1}^L \sum_{(x,y) \in S} |Ipr(i,x,y) - Ifd(i,x,y)|. \quad (3)$$

When the attention region is  $S$ , the pixel number contained in  $S$  is  $N(S)$ . We use the spatial difference to evaluate the clear image for the density of the fusion image in the following equation;

$$\begin{aligned} E2 = \frac{1}{N(S) \cdot L} \sum_{i=1}^L & \sum_{(x,y) \in S} [|dif(Ipr(i,x,y)) \\ & - dif(Ifd(i,x,y))|], \end{aligned}$$

where

$$\begin{aligned} dif(Ii(i,x,y)) = & \sqrt{[Ii(i,x,y) - Ii(i,x,y-1)]^2 \\ & + [Ii(i,x,y) - Ii(i,x-1,y)]^2}. \end{aligned} \quad (4)$$

In the case of evaluation values  $E1$  and  $E2$  for the intensity image,  $L = 1$ . In the case of evaluation values  $E1$  and  $E2$  for hue and saturation images,  $L = 2$ . An image is evaluated using both  $E1$  and  $E2$  in the following equation;

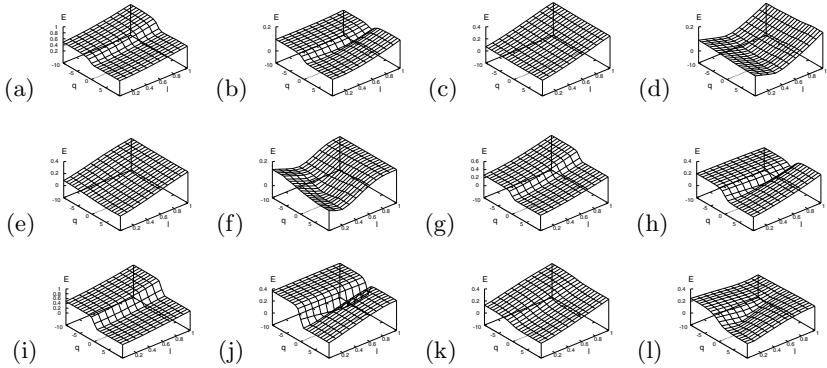
$$E3(\alpha) = \alpha * E1 + (1 - \alpha) * E2. \quad (5)$$

$Ipr(i,x,y)$  is a pixel value of a reference image, and  $Ifd(i,x,y)$  is a pixel value of a fusion image achieved as result. The point of  $(q,l)$  is a method of the image fusion. The fusion image is evaluated in the two dimensional space with  $q$  axis and  $l$  axis.

The fusion image is evaluated using the two step algorithm. In the first step, an original image is partitioned into regions  $a$ , region  $b$  and region  $c$  using histogram. As an image fusion result is generated from the two images, an fusion image is partitioned into region  $aa$ , region  $bb$ , region  $cc$ , region  $ab$ , region  $ac$  and region  $bc$ . Region  $ac$  of an image fusion result is generated from region  $a$  of a image and region  $c$  of another image by pixel. Region  $aa$  of an image fusion result is generated from region  $a$  of a image and region  $a$  of another image by pixel. We evaluate the six regions using  $E(\alpha = 0.6)$ . We defined the parameter  $\alpha = 0.6$  derived good quality image empirically. The reason why we evaluate using  $E(\alpha = 0.6)$  is to evaluate the general error using both the differential error and the pixel value error. The fusion parameters  $q$  and  $l$  with the minimum value of  $E(\alpha = 0.6)$  for each region is defined independently. The method by the fusion parameters  $q$  and  $l$  derived for the saturation and hue image in this method is suited to generate the shaded image.

The fusion parameters  $q$  and  $l$  for the each region of the intensity image derived in this method are used in the next step as an initial value. The parameter of a region is varied while the parameters of the other regions are fixed. The six regions are evaluated using  $E(\alpha = 0.0)$  in turn. The reason why we evaluate using  $E(\alpha = 0.0)$  for the intensity image again is to decrease the intensity difference for boundary of each region. After the fusion parameters  $q$  and  $l$  of the six regions was updated, we evaluate the whole image using  $E(\alpha = 0.6)$ . At first, the process starts from a region where the evaluation value  $E(\alpha = 0.6)$  is the maximum in six regions. The reason why we evaluate the whole image using the  $E(\alpha = 0.6)$  after the fusion parameters of the six regions are updated using  $E(\alpha = 0.0)$  is to prevent the differential error become more than the pixel value error. If the evaluation value  $E(\alpha = 0.6)$  calculated for the whole image after the six regions parameter is updated using  $E(\alpha = 0.0)$  is more than the initial value  $E(\alpha = 0.6)$  for the whole image derived in the previous step, the step algorithm is finished. If not, the fusion parameter is updated using this algorithm repeatedly.

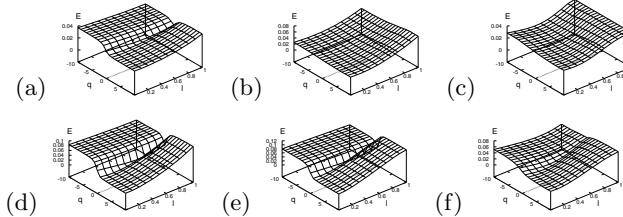
Figure 4(a)(b)(c)(d)(e) and (f) are the fusion space about intensity of region  $aa$ , region  $bb$ , region  $cc$ , region  $ab$ , region  $ac$  and region  $bc$ . Figure 4 (b)(d) and (f) show image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the intensity element. Figure 4 (a)(c) and (e) show image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the hue and saturation element. Figure 4 (a) and (b) are evaluated for the region  $aa$  of the image set A. Figure 4 (c) and (d) are evaluated for the region  $bb$  of the image set A. Figure 4 (e) and (f) are evaluated for the region  $cc$  of the image set A. Figure 4 (h)(j) and (l) show image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the intensity element. Figure 4 (g)(i) and (k) show image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the hue and saturation element. Figure 4 (g) and (h) are evaluated for the region  $ab$  of the image set A. Figure 4 (i) and (j) are evaluated for the region  $ac$  of the image set A. Figure 4 (k) and (l) are evaluated for the region  $bc$  of the image set A. We define the fusion parameters  $q$  and  $l$  with the minimum value for the shaded intensity image and the hue and saturation image of  $aa$ ,  $bb$ ,  $cc$ ,  $ab$ ,  $ac$  and  $bc$  regions. Table 2 shows the best evaluation value  $E3(\alpha = 0.6)$  and the fusion parameters  $q$  and  $l$  derived for the intensity image and the saturation and hue image of the image set A. Figure 5 show image fusion spaces evaluated using the value  $E(\alpha = 0.0)$  for the intensity element using a fusion parameters derived by the previous process as a initial value. Figure 5 (b)(d) and (f) show image fusion spaces evaluated using the value  $E(\alpha = 0.0)$  for the hue and saturation element. Figure 5 (a)(b)(c)(d)(e) and (f) are evaluated for the region  $aa$ ,  $bb$ ,  $cc$ ,  $ab$ ,  $ac$  and  $bc$  of the image set A. We define the fusion parameters  $q$  and  $l$  with the minimum value for  $aa$ ,  $bb$ ,  $cc$ ,  $ab$ ,  $ac$  and  $bc$  regions. The best evaluation value  $E3(\alpha = 0.0)$  is calculated using the fusion parameters  $q$  and  $l$  derived for the shaded intensity image of the image set A by the evaluation  $E(\alpha = 0.0)$ . The evaluation value  $E3(\alpha = 0.0)$  was 0.019448.



**Fig. 4.** Image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the intensity element (b)(d)(f) and the hue and saturation element (a)(c)(e) for (a)(b)region aa, (c)(d) region bb, (e)(f) region cc of the image set A. Image fusion spaces evaluated using the value  $E(\alpha = 0.6)$  for the intensity element (h)(j)(l) and the hue and saturation element (g)(i)(k) for (g)(h) region ab, (i)(j) region ac and (k)(l)region bc of the image set A.

**Table 2.** The best evaluation value  $E3(\alpha = 0.6)$  and the fusion parameters  $q$  and  $l$  derived for the intensity image and the saturation and hue image of the image set A. The evaluation value  $E3(\alpha = 0.0)$  is 0.019448.

area	1 i	q i	$E3(\alpha = 0.6)$ i	1 hs	q hs	$E3(\alpha = 0.6)$ hs
aa	0.300000	1.000000	0.035653	0.000000	2.000000	0.298037
bb	0.400000	0.000000	0.017439	0.000000	5.000000	0.060434
cc	0.300000	1.000000	0.024500	0.000000	10.000000	0.042229
ab	0.000000	10.000000	0.028027	0.000000	2.000000	0.081633
ac	0.000000	10.000000	0.049987	0.000000	2.000000	0.059423
bc	0.300000	1.000000	0.026066	0.000000	5.000000	0.044507



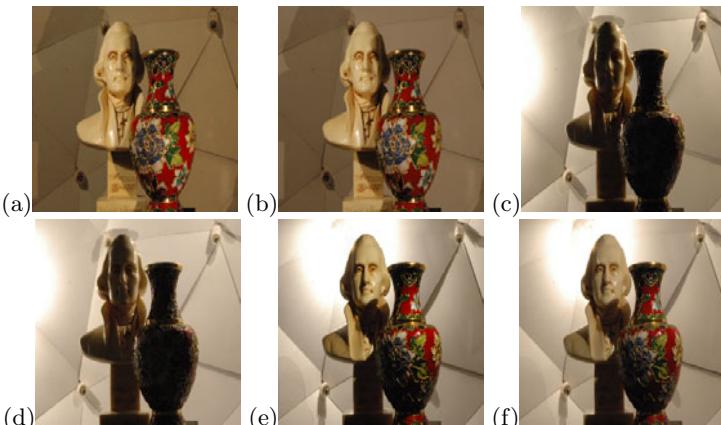
**Fig. 5.** Image fusion spaces evaluated using the value  $E(\alpha = 0.0)$  for intensity element for (a)region aa, (b) region bb, (c) region cc, (d) region ab, (e) region ac and (f)region bc of the image set A.

### 3.2 Generating Shaded Image With Lighting

Figure 6 (a) shows the result derived by fusing two images with a front light for the image set A of figure 1. Figure 6(b) shows a reference image for the image set A. Figure 6 (c) shows the result derived by fusing two images with a front

light for the image set B of figure 11. Figure 6(d) shows a reference image for the image set B. Figure 6(e) shows the result derived by fusing two images with a front light for the image set C of figure 11. Figure 6(f) shows a reference image for the image set C. The image set A is the combination of two images with a front light. The image set B is the combination of two images with a back light. The image set C is the combination of an image with a front light and an image with a back light. It is found that the fusion image is similar to the reference image. It can be applied for the combination of two images with a front light, the combination of two images with a back light and the combination of an image with a front light and an image with a back light.

As the number of training pattern changes, the performance changes in the case of defining the parameter.



**Fig. 6.** (a) The result derived by fusing two images with a front light for the image set A. (b) A Reference image for the image set A. (c) The result derived by fusing two images with a front light for the image set B. (d) A Reference image for the image set B. (e) The result derived by fusing two images with a front light for the image set C. (f) A Reference image for the image set C.

## 4 Conclusions

An image with all lighting equipments included in two photographs was generated from two photographs taken with a single-lens reflex camera with the sensitivity automatic control facilities. We find the image fusion method for regions partitioned using a histogram in the combinations of a front and a back light. We can find the best method by searching the image fusion space describing the neighbor relation between the traditional fusion methods. It was confirmed that the fusion image including two shaded features is generated. This method is effectiveness by applying for the general image except the experiment equipment.

## References

1. Levin, A., Zomet, A., Weiss, Y.: Learning how to inpaint from global image statistics. In: Proc. of Ninth IEEE International Conference on Computer Vision, pp. 305–312 (2003)
2. Bertalmio, M., Sapiro, G., Caselles, V., Ballester, C.: Image inpainting. In: SIGGRAPH 2000, pp. 417–424 (2000)
3. Bertalmio, M., Bertozzi, A.L., Sapiro, G.: Navie-stokes fluid dynamics and image and video inpainting. In: Proc. IEEE Computer Vision and Pattern Recognition, vol. 1, pp. 355–362 (2001)
4. Toet, A.: Multiscale contrast enhancement with application to image fusion. Optical Engineering 31, 1026–1031 (1992)
5. De, I., Chanda, B.: A simple and efficient algorith for multifocus image fusion using morphological wavelets. Signal Processing, elsevira (2005)
6. Tapiador, F.J., Casanova, J.L.: An algorithm for the fusion of images based on jaynes maximum entropy method. Int. J. Remote Sensing 23, 777–785 (2002)
7. Li, H., Manjunath, B.S., Mitra, S.K.: Multisensor image fusion using the wavelet transform. Graphical Models Image Process. 57, 235–245 (1995)
8. Zhang, Z., Blum, R.S.: A categorization of multiscale decomposition-based image fusion schimes with a performance study for a digital camera application. Proc. IEEE 87, 1315–1326 (1999)
9. Rancin, T., Wald, L.: Fusion of high spatial and spectral resolution images: The arsis concept and its implementation. Photogrammetric Engineering and Remote Sensing 66, 19–61 (2000)
10. Pajares, G., del la Crus, J.M.: A wavelet-based image fusion tutorial. Pattern Recognition, 1855–1871 (2004)
11. Wang, A.H.: A new multiwavelet-based approach to image fusion. Journal of Mathematical Imaging and Vision 21, 177–192 (2004)
12. Pu, T.: Contrast-based image fusion using the discrete wavelet transform. Int. J. Remote Sensing 39, 2075–2082 (2000)
13. Shashua, A.: Geometry and photometry in 3d visual recognition. MIT, Cambridge (1992)
14. Behumeur, P.N., Kriegman, D.J.: What is the set of image of and object under all possible lighting conditions? In: Proc. IEEE Computer Vision and Pattern Recognition 1996, pp. 270–277 (1996)
15. Debevec, P., Hawkins, T., Tchou, C., Duiker, H., Sarokin, W., Sagar, M.: Acquiring the Reflectance Field of a Human Face. In: SIGGRAPH 2000, pp. 145–157 (2000)

# Automatic Detection of Morphologically Distinct Objects in Biomedical Images Using Second Generation Wavelets and Multiple Marked Point Process

Hiroshi Hatsuda

Department of Computational Biology, The University of Tokyo, Japan

**Abstract.** Automatically analyzing morphology of biological objects such as cells, nuclei, and vessels is important for medicine and biology. However, detecting individual biological objects is challenging because biomedical images tend to have a complex structure composed of many morphologically distinct objects and unclear object boundaries. In this paper, we present a novel approach to automatically detect individual objects in biomedical images using a multiple marked point process, in which points are the positions of the objects and marks are their geometric attributes. With this model, we can consider both prior knowledge of the structure of the objects and observed data of an image in object detection. Our proposed method also uses the second generation wavelets-based edge-preserving image smoothing technique to cope with unclear boundaries of biological objects. The experimental results show the effectiveness of our method.

## 1 Introduction

Analyzing morphology of biological objects such as cells, nuclei, and vessels in biomedical images has been becoming more important than ever in the fields of current medicine and biology. Morphologically inspected anatomical images can allow the diagnoses of diseases and injuries at early stages in a less invasive way [1]. In addition, the success of genome sequencing projects has enabled us to develop genetic technologies to perturb individual genes systematically in sequenced genomes. Observing the effects of perturbed genes in the cells by using the systematic genome-wide techniques is reliable and has become a standard strategy for identifying gene functions [2]. For this purpose, the morphology of the gene-perturbed cells is one of the most significant features to be examined.

Although morphological analysis has important roles in medicine and biology, it is still challenging because of the following two problems. The first problem is that biomedical images tend to contain many objects such as cells, nuclei, and vessels, and the boundary of each object is apt to be blurred. The second problem is that biomedical images tend to have a complex structure composed of morphologically distinct objects, and thus it is difficult to incorporate geometric constraints of individual objects and an overall complex structure of objects into image analysis algorithms.

Several studies have been performed to detect objects in biomedical images. Yoo *et al.* [3] implemented various algorithms such as region growing [4], watersheds [5],

level set [6] to extract biological objects from images. Carpenter *et al.* [7] also developed cell identification software using many fundamental techniques including thresholding, mathematical morphology [8], and watersheds [5]. Although the method by McCullough *et al.* [9] can accurately locate cell borders by using dynamic programming, it requires user manipulation. Al-Kofashi *et al.* [10] proposed a novel technique based on multiscale Laplacian-of-Gaussian filtering [11] and graph cut image segmentation [12] to detect cell nuclei. Moreover, retinal vessels can be extracted in [13] by using an active contour model [14]. Although some of these methods can perform well in a specific type of image, they do not cope with the two problems described above.

To deal with the first problem, we use the second generation wavelets for edge-preserving image smoothing [15]. Edge-preserving image smoothing is an image processing technique to smooth images and yet preserve edges. Traditionally, bilateral filter [16] is one of the most widely used edge-preserving smoothing methods. This is a non-linear filter, where each pixel in the filtered image is a weighted mean of its neighbors; the weights decrease both with spatial distance and with difference in pixel value. Although bilateral filter performs well to some extent, it is of limited effectiveness for edge-preserving smoothing. To overcome the limitations of bilateral filter, Fattal proposed an edge-preserving image smoothing technique based on edge-avoiding wavelets [15], which uses the second generation wavelet transform [17] and achieves nonlinear data-dependent multi-scale edge-preserving image smoothing. The second generation wavelets-based smoothing approach outperforms or competes with other state-of-the-art methods in its ability of edge-preserving image smoothing regardless of its simplicity, and surpasses all of them in the computational cost. This technique can emphasize object boundaries in images by decreasing the differences of image intensity in all parts except for edges, and we therefore employ it to address the first problem in this study.

To deal with the second problem, several algorithms have been proposed for detecting objects such as cells, nuclei, and vessels in biomedical images. Active contour [14] is one of the most widely-used and well-studied image segmentation techniques in biomedical imaging. Active contour model uses prior knowledge of the shape of an object by confining the smoothness or the symmetry of an object boundary within a certain range. Although it is a powerful approach, it has two difficult issues related to topological change and initialization. In other words, contours must be initialized close to the boundaries for all objects in order that it successfully extracts unknown number of objects. Therefore, active contour model cannot be used in this study. Even though we solve the initialization problem by using better methods such as level set image segmentation technique [6], we still cannot overcome our second problem. Although the level set method can cope with unknown number of objects, it cannot use prior knowledge of the shape of an overall complex structure of objects and thus has difficulty in detecting individual objects in images containing numerous objects. To understand the necessity of using prior knowledge of an overall structure, the well-known image of “Dalmatian dog” proposed by R. Gregory [18] is highly effective. If we are unfamiliar with this picture, it is likely that it will not make sense. Once the dog silhouette has been seen, it becomes clear that there is a dog in the original image. This example illustrates the amazing ability of the human visual system to find meaningful objects in images by using prior knowledge of what we see in them [19]. The

important point here is that this is not a particular case; this is our basic visual capability. We can identify individual objects (e.g. a cell) only by knowing that they consist in an overall structure (e.g. a tissue composed of many cells). Without the prior knowledge of the overall structure, it is difficult to distinguish between a local meaningful object and a noise. Therefore, computer vision algorithms should also use prior knowledge of what they see; in other words, they should employ geometric features of an overall structure in an image to detect individual objects. In addition, primitive methods such as thresholding, region growing [4], and watersheds [5] image segmentation cannot extract individual objects successfully because they do not consider the geometric constraints of objects of interest and thus are vulnerable to local noise in images.

Therefore, we employ a multiple marked point process (MMPP) to automatically detect individual biological objects. Unlike the methods described above, MMPP can incorporate the geometric constraints of objects of interest and their interactions into an object detection algorithm. In MMPP, points are positions of the objects and marks are their geometric attributes such as a disk [20], line segments [21], and rectangles [22]. MMPP can use both the information of image intensity to fit the objects in an image and prior information about geometric constraints of the overlaps of the objects. MMPP is formulated by using a Gibbs energy, and the minimum energy, corresponding to the optimal configuration of objects, is estimated by a Reversible Jump Markov Chain Monte Carlo (RJMCMC) [23] or a birth-and-death dynamics [20] coupled with the traditional simulated annealing [24]. We also propose a multiple birth-and-death dynamics and a non-uniform birth process of multiple marks in the optimization step.

## 2 Method

### 2.1 Edge-Preserving Image Smoothing Using the Second Generation Wavelets

The first step of our method is edge-preserving image smoothing, which is important for biological object detection because of their blurred boundary. For this purpose, we use a technique based on edge-avoiding wavelets (EAW) [15] using the second generation wavelet transform (SGWT) [17]. Unlike traditional wavelets, SGWT consists of the lifting scheme in which base functions are not designed explicitly and depend on input data to cope with local particularities of the data. With SGWT, EAW realizes nonlinear data-dependent multi-scale image decomposition, which is represented by

$$I(x, y) \mapsto a^J, \left\{ d^j \right\}_{j=1}^J \quad (1)$$

where  $I(x, y)$  is the gray-scale intensity or the logarithm of Y component of the YUV color representation in monochrome or color image, respectively,  $a^J$  is the approximation coefficient at the coarsest scale  $J$ , and  $\left\{ d^j \right\}_{j=1}^J$  denotes detail coefficients from the finest scale to the coarsest scale. We refer to [15] for the details of EAW-based image decomposition. The edge-preserving image smoothing is conducted by

the reconstruction of the decomposed data. We reconstruct the decomposed image according to

$$a^J, \left\{ \gamma^j d^j \right\}_{j=1}^J \mapsto I'(x, y) \quad (2)$$

where  $\gamma^j \leq 1$  is a parameter to determine the extent of smoothing at scale  $j$ . Small  $\gamma$  values at fine scales smooth the details of images, whereas large values at fine scales enhance the details.

## 2.2 Object Detection Using a Multiple Marked Point Process

The second step is object detection using a multiple marked point process. We model a biomedical image as a collection of objects whose positions and features are determined by a marked point process [20 – 22] in  $X = P \times M$ .  $X$  is a marked point process in a position space  $P$  where each point is associated with a mark from a mark space  $M$ . The Poisson process defines complete random configurations of points in  $P$ , and  $M$  is composed of a set of parameters of multiple marks. In this study, we use a set of ellipses as the marks. A marked point process extends a point process by adding specific marks associated with objects to each point in order to model images in terms of geometrical features. We therefore can consider an image as the configuration of ellipses.

To model measurements consistent with data and interactions between objects realized from a marked point process, we define a Gibbs density  $d(x)$  of a configuration

$x$  of the objects:  $d(x) = \frac{1}{Z} \exp(-U(x))$ , where  $Z$  is a normalizing constant  $Z = \int \exp(-U(x)) dx$  and a Gibbs energy  $U(x)$  defined by

$$U(x) = w U_d(x) + U_p(x). \quad (3)$$

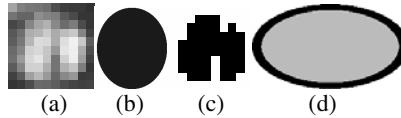
In (3), the data energy  $U_d(x)$  considers how the objects fit to the image intensity, the prior energy  $U_p(x)$  considers interactions of the objects, and  $w$  is a weighting factor between the two energy terms. The density  $d(x)$  will be maximized on the optimum configuration  $\hat{x}$ . In other words, the optimum configuration can be determined by minimizing the Gibbs energy  $U(x)$ :  $\hat{x} = \arg \min_{x \in X} U(x)$ .

### 2.2.1 Data Energy

We consider the data energy to fit the objects to the image intensity (luminance). Because the image intensity of an object is usually different from that of background, we measure the difference between the intensity distribution inside an object and that of the object's background. In this study, we use a set of ellipses to represent biological objects. We need three parameters to define an ellipse: semimajor axis  $a$ , semiminor axis  $b$ , and orientation  $\theta$ . Thus, the parameter space  $M$  is  $M = [a_{\min}, a_{\max}] \times [b_{\min}, b_{\max}] \times [0, \pi]$ , where  $(a_{\min}, a_{\max})$  and  $(b_{\min}, b_{\max})$  are the minimum and the maximum semimajor axis and semiminor axis, respectively. Let

$x_i = (p, k)$  be an ellipse, where  $p$  is a position and  $k = (a, b, \theta)$  is a mark. We define the background of an ellipse as the region between an ellipse  $x_i$  and a concentric ellipse  $x'_i = (p, k')$ , where  $k' = (a + \eta, b + \eta, \theta)$ . We denote  $R_f(i)$  the foreground region of an ellipse and  $R_b(i)$  the region of its background.

In addition, we modify the foreground region by using graph cut image segmentation [12]. We model the region in an ellipse as a mixture of two Gaussian distributions and estimate the parameters of the two Gaussians by using divided histograms determined by entropy-based image segmentation [25]. Instead of user manipulation, we use the two Gaussians to automatically define the regional energies in graph cut method [12]. We thus exclude background pixels decided by the graph cut from the foreground of an ellipse. We need this modification because geometric figures do not always represent the foreground of a biological object. For example, a nucleus may be different from cell cytoplasm and similar to background in image intensity. In this case, ellipses have difficulty in representing the foreground of a cell. Figure 1 illustrates the definitions of foreground and background of an ellipse.



**Fig. 1.** Foreground and background of an ellipse. (a): An object in an original image, (b): An ellipse to represent (a), (c): Foreground determined by graph cut method. We use this region to compute the Bhattacharya distance of the ellipse (b), (d): Background of an ellipse. Black region is the background of a gray ellipse.

We employ the Bhattacharya distance measure  $D(i)$  between the distribution of the image intensity in  $R_f(i)$  and that in  $R_b(i)$  [20]. The data energy  $U_d(x_i)$  associated with a mark  $x_i$  is defined as follows:

$$U_d(x_i) = \begin{cases} 1 - \frac{D(i)}{h} & \text{if } D(i) < h \\ \exp\left(-\frac{D(i)-h}{D(i)}\right) - 1 & \text{if } D(i) \geq h \end{cases}, \quad (4)$$

where  $h$  is a threshold parameter.  $U_d(x_i) \in [-1, 1]$  provides positive values to weakly contrasted marks and negative values to well located marks.

### 2.2.2 Prior Energy

We consider the prior energy to integrate prior knowledge of the structure of the objects into object detection. In this study, we focus on the objects' overlap. Each mark in the final configuration denotes an object such as cell, nucleus, and a fraction of a vessel. Because objects do not overlap in a 2D image, we need to minimize the overlap between the marks in an estimated configuration. However, it is not desirable to

completely forbid the overlapping marks because they are only approximations of the shapes of objects and optimum configuration may contain some slightly overlapping marks. We therefore define the prior energy  $U_p(x)$ :

$$U_p(x) = \sum_{x_i, x_j \in x} \left( e^{\kappa C(x_i, x_j)} - 1 \right) \quad (5)$$

where  $x$  represents a configuration of objects,  $C(x_i, x_j)$  is a function to quantify the relative mutual overlap [0,1] between a mark  $x_i$  and  $x_j$ , and  $\kappa$  is a parameter to determine the degree of the penalization. This prior energy takes account of all interactions between neighboring objects and it approximates the geometric constraint of an overall structure of objects in an image.

### 2.2.3 Energy Minimization

To achieve the optimum configuration of the objects, we need to minimize the Gibbs energy  $U(x)$  (3). Reversible Jump Markov Chain Monte Carlo (RJMCMC) [21 – 23] and birth-and-death dynamics [20] are used to minimize the energy  $U(x)$  (3) in marked point process models. We need RJMCMC instead of MCMC because we need to cope with the different number of objects. In RJMCMC, a move from a configuration  $x$  to  $y$  is executed according to a kernel  $Q_m(x \rightarrow y)$ , and the move is accepted with the probability

$$\min \left( 1, \frac{Q_m(y \rightarrow x)}{Q_m(x \rightarrow y)} e^{-\frac{U(y)-U(x)}{T}} \right). \quad (6)$$

Reversible jumps between the different dimensions (i.e. the number of objects) are performed by a birth-and-death kernel  $Q_{BD}$ . We select to add or remove an object by following the Poisson process. The acceptance ratio of the added object is defined by

$$\frac{Q_{BD}(y \rightarrow x)}{Q_{BD}(x \rightarrow y)} = \frac{p_d}{p_b} \frac{\nu(P)}{n(x)+1} = \frac{\nu(P)}{n(x)+1} \quad (7)$$

where  $P_b$  is the probability of selecting a birth,  $P_d$  is the probability of selecting a death, and  $\nu(P)$  is the intensity of the reference Poisson process. A randomly selected object is  $x_{n(x)+1}$ , and the configuration is changed from  $x$  to  $y = x \cup \{x_{n(x)+1}\}$ . In this study, the probabilities of birth and death are the same ( $p_b = p_d$ ). If an object is added, its mark and parameters are randomly selected. Furthermore, the acceptance ratio of the removed object is defined by

$$\frac{Q_{BD}(y \rightarrow x)}{Q_{BD}(x \rightarrow y)} = \frac{p_b}{p_d} \frac{n(x)}{\nu(P)} = \frac{n(x)}{\nu(P)} \quad (8)$$

where a randomly selected object is  $x = x_i, i \in [1, n(x)]$ , and the configuration is changed from  $x$  to  $y = x - \{x_i\}$ .

Changing the mark of an object is performed by a switching kernel  $Q_s$ . Unlike the birth-and-death kernel, this move changes neither the number of objects nor the number of parameters of a mark. We thus use a switching kernel  $Q_s = 1$  in this process.

In addition, we incorporate a data-dependent term into the birth-and-death kernel to accelerate this process. It is preferable to select well-positioned objects more often than weakly-positioned ones in the birth process. For this purpose, we use the data energy  $U_d(x_i)$  to determine a data-dependent birth probability  $B(x_i)$  at every pixel:

$$B(x_i) = \frac{\min U_d(x_i) + 1}{\sum_j (\min U_d(x_j) + 1)}. \quad (9)$$

where  $\min U_d(x_i)$  represents the minimum data energy among all marks at  $x_i$ . The new birth-and-death kernels are given by

$$\frac{Q_{BD}(y \rightarrow x)}{Q_{BD}(x \rightarrow y)} = \frac{p_d}{p_b} \frac{\nu(P)}{(n(x)+1)B(x_i)}, \quad \frac{Q_{BD}(y \rightarrow x)}{Q_{BD}(x \rightarrow y)} = \frac{p_b}{p_d} \frac{n(x)B(x_i)}{\nu(P)} \quad (10)$$

where  $x_i$  is an added or removed object. We use these birth-and-death kernels in this study. We decrease  $T$  according to  $T_t = \alpha T_0$  ( $\alpha = 0.99999$ ) in each iteration of this process.

Although RJMCMC is effective to minimize the Gibbs energy, the computational cost of RJMCMC is too expensive to analyze many images. A birth and death dynamics is proposed in [20] to search the optimum configuration in a marked point process, and is much more effective than RJMCMC in terms of computational cost. We extend it and present a multiple birth and death dynamics to cope with multiple marks. We define non-uniform birth rates based on the data energy for each mark and death rates based on both a present configuration and the data energy according to the detailed balance conditions. The algorithm of a multiple birth and death dynamics is presented in Algorithm 1. Additionally, we use Mersenne twister [26] to generate a pseudo-random number to compare the probabilities in RJMCMC and multiple birth-and-death processes.

**Algorithm 1.** Multiple birth and death dynamics

1. The data energy  $U_d^m(x_i)$  (3) is computed for each pixel  $i \in I$  on an image  $I$ . We compute it for each mark, and the superscript  $m$  represents a mark  $m \in M$ .
2. The inverse temperature parameter  $\beta$  and the discretization step  $\delta$  are initialized.
3. For each pixel  $i \in I$  if there is no object whose center is at  $i$ , we randomly select a mark with probability

$$P(U_d(x_i) = U_d^m(x_i)) = \frac{U_d^m(x_i)}{\sum_{j \in M} U_d^j(x_i)}. \quad (11)$$

4. The birth rate for each  $i$  is computed according to

$$B(i) = \frac{zb(i)}{\sum_{j \in I} b(j)}, \quad (12)$$

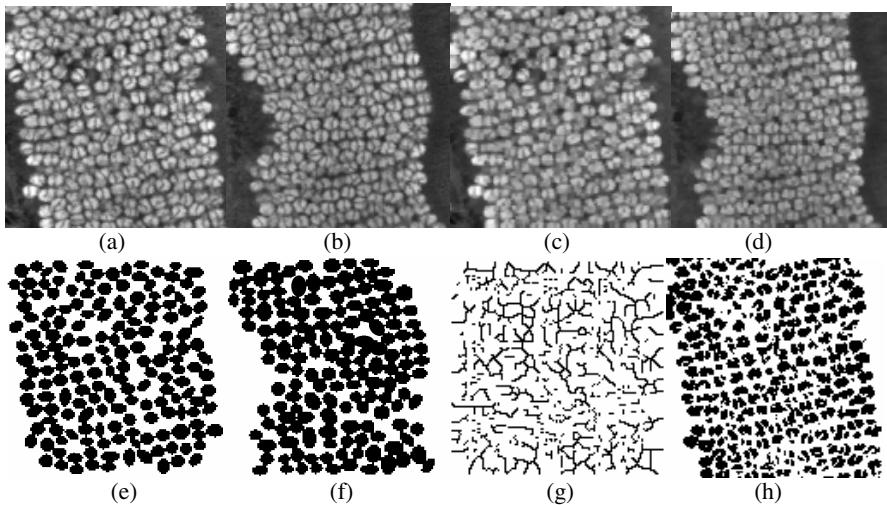
where  $z$  is a parameter of this step and

$$b(i) = 1 + 9 \frac{\max_{j \in I} U_d(x_j) - U_d(x_i)}{\max_{j \in I} U_d(x_j) - \min_{j \in I} U_d(x_j)}. \quad (13)$$

5. Birth step: For each  $i$  if there is no object with center  $i$ , we add a marked point with probability  $\delta B(i)$ .
6. The prior energy  $U_p(x_i)$  of the current configuration  $x$  is computed for each pixel  $i$ .
7. Death step: We sort the current configuration  $x$  from the highest to the lowest data energy. Then, the death rate  $d(x_i)$  is computed according to  $d(i) = \frac{1}{1 + \delta a(i)}$ , where  $a(i) = \exp(-\beta U(x_i))$ . We delete a mark with probability  $d(i)$ .
8. Convergence test: The convergence is achieved if the following conditions are met: all the objects added in the birth step have been deleted in the death step, and all the objects deleted in the death step have been added in the birth step. If the process has not converged, the temperature and the discretization step are decreased by a given factor. Then, the process goes back to step 3.

### 3 Experimental Results and Discussion

We evaluated the proposed method on a set of images of cat retinal photoreceptor nuclei [27]. Detecting and counting the photoreceptors is important for the diagnosis of retina degeneration. Figure 2 shows the results of object detection. Figure 2(a) and (b) are original images, (c) and (d) are smoothed images of (a) and (b) respectively by using SGWT-based edge-preserving image smoothing, and (e) and (f) are detection results of (a) and (b) respectively with the proposed method. We also apply level set [6] and entropy-based [25] image segmentation algorithms to analyze Figure 2(a), and show their results in (g) and (h), respectively. Figure 2(g) depicts border lines determined by the level set method, which did not converge because of unclear boundaries of objects. Entropy-based method roughly extracted objects; however, it did not accurately detect object boundaries and cannot identify individual objects unlike our method. These results demonstrate the effectiveness of our method, and multiple objects are accurately detected. It takes approximately 15 seconds to analyze a  $128 \times 128$  image with the multiple birth-and-death dynamics by using a 2.0 GHz CPU computer. Previous approaches including level set, entropy-based, and other methods described in Introduction have difficulty detecting individual objects in these images because of the lack of geometric constraints of an overall structure in images. Finally, we chose  $\{\gamma\}$ ,  $w$ ,  $a_{\min}$ ,  $a_{\max}$ ,  $b_{\min}$ , and  $b_{\max}$ ,  $\{\theta\}$ ,  $\eta$ ,  $h$ ,  $\kappa$ ,  $z$ , initial  $\beta$ , initial  $\delta$ , step size of  $\beta$ , and step size of  $\delta$  to be  $\{0.0, 0.8\}$ , 1, 1, 4, 10, 4, 10,  $\{0, \pi/4, \pi/2, 3\pi/4\}$ , 1, 5000, 100, 20, 50, 0.1, and 5, respectively.



**Fig. 2.** Experimental results. (a), (b): Original images, (c), (d): Smoothed images using SGWT-based image smoothing, (e), (f): Result images using the proposed method, (g): Result image of (a) using level set method, (h): Result image of (a) using entropy-based method.

## References

1. Doi, K.: Computer-aided diagnosis in medical imaging: Historical review, current status and future potential. *Comput. Med. Imaging Graph.* 31(4-5), 198–211 (2007)
2. Carpenter, A.E., Sabatini, D.M.: Systematic genome-wide screens of gene function. *Nature Review Genetics* 5, 11–22 (2004)
3. Yoo, T., Metaxas, D.: Open science – combining open data and open source software: medical image analysis with the Insight Toolkit. *Medical Image Analysis* 9, 503–506 (2005)
4. Chang, Y.L., Li, X.: Adaptive image region-growing. *IEEE Trans. Image Processing* 3(6), 868–872 (1994)
5. Vincent, L., Soille, P.J.: Watersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans. Pattern Anal. Machine Intell.* 13(6), 583–598 (1991)
6. Malladi, R., Sethian, J.A., Vemuri, B.C.: Shape modeling with front propagation: a level set approach. *IEEE Trans. Pattern Anal. Machine Intelligence* 17(2), 158–175 (1995)
7. Carpenter, A.E., et al.: CellProfiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biology* 7, R100 (2006)
8. Dougherty, E.R.: Mathematical morphology in image processing. CRC press, Boca Raton (1992)
9. McCullough, D.P., Gudla, P.R., Harris, B.S., Collins, J.A., Meaburn, K.J., Nakaya, M., Yamaguchi, T.P., Misteli, T., Lockett, S.J.: Segmentation of whole cells and cell nuclei from 3D optical microscope images using dynamic programming. *IEEE Trans. Medical Imaging* 27(5), 723–734 (2008)
10. Al-Kofahi, Y., Lassoud, W., Lee, W., Roysam, B.: Improved automatic detection and segmentation of cell nuclei in histopathology images. *IEEE Trans. Biomedical Engineering* 57(4), 841–852 (2010)

11. Lindeberg, T.: Feature detection with automatic scale selection. *Int. J. Comp. Vis.* 30(2), 79–116 (1998)
12. Boykov, Y., Funka-Lea, G.: Graph cuts and efficient N-D image segmentation. *Int. J. Comp. Vis.* 70(2), 109–131 (2006)
13. Al-Diri, B., Hunter, A., Steel, D.: An active contour model for segmenting and measuring retinal vessels. *IEEE Trans. Medical Imaging* 28(9), 1488–1497 (2009)
14. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: active contour models. *Intl. J. Comput. Vis.* 1(4), 321–331 (1988)
15. Fattal, R.: Edge-avoiding wavelets and their applications. *ACM Trans. Graphics* 28(3) (2009)
16. Tomasi, C., Manduchi, R.: Bilateral filtering for gray and color images. In: Proc. ICCV, pp. 839–846 (1998)
17. Sweldens, W.: The lifting scheme: a construction of second generation wavelets. *SIAM J. Math. Anal.* 29(2), 511–546 (1997)
18. [http://www.michaelbach.de/ot/cog\\_dalmatian/](http://www.michaelbach.de/ot/cog_dalmatian/)
19. Gilbert, C.D., Sigman, M.: Brain states: top-down influences in sensory processing. *Neuron* 54(5), 677–696 (2007)
20. Descombes, X., Minlos, R., Ehzhina, E.: Object extraction using a stochastic birth-and-death dynamics in continuum. *J. Math. Imaging Vis.* 33, 347–359 (2009)
21. Lacoste, C., Descombes, X., Zerubia, J.: Point processes for unsupervised line network extraction in remote sensing. *IEEE Trans. Pattern Anal. Machine Intelligence* 27(10), 1568–1579 (2005)
22. Ortner, M., Descombes, X., Zerubia, J.: Building outline extraction from digital elevation models using marked point processes. *Int. J. Comp. Vis.* 72(2), 107–132 (2007)
23. Green, P.J.: Reversible jump Markov chain Monte Carlo computation and Bayesian model determination. *Biometrika* 82, 711–732 (1995)
24. Metropolis, M., Rosenbluth, A., Teller, A., Teller, E.: Equation of state calculations by fast computing machines. *J. Chemical Physics* 21, 1087–1092 (1953)
25. Abutaleb, A.S.: Automatic thresholding of gray-level pictures using two-dimensional entropy. In: Computer Vision, Graphics, and Image Processing, vol. 47, pp. 22–32 (1989)
26. Matsumoto, M., Nishimura, T.: Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator. *ACM Trans. Modeling and Computer Simulation* 8(1), 3–30 (1998)
27. Gelasca, E.D., Obara, B., Fedorov, D., Kvilekval, K., Manjunath, B.S.: A biosegmentation benchmark for evaluation of bioimage analysis methods. *BMC Bioinformatics* 10, 368 (2009)

# Imaging-Based Computation of the Dynamics of Pelvic Floor Deformation and Strain Visualization Analysis

Christos E. Constantinou<sup>1</sup>, Linda McLean<sup>2</sup>, Ellen Kuhl<sup>3</sup>, and Bertha Chen<sup>4</sup>

<sup>1</sup> Stanford University and Spinal Cord Injury VA (640/128) Palo Alto, CA, USA

<sup>2</sup> School of Rehabilitation Therapy, Queen's University, Ontario, Canada

<sup>3</sup> Department of Mechanical Engineering, Stanford University, Stanford, CA

<sup>4</sup> Department of Obstetrics and Gynecology,

Stanford University School of Medicine, Stanford, CA

**Abstract.** Understanding of the visco-elastic and contractile properties of pelvic floor muscles and their characteristics in deforming soft tissues are important factors in predicting their functionality. Ultrasound imaging provides a noninvasive way to visualize the temporal sequence of the displacement and deformation of abdominal structures and this paper provides a conceptual framework for dynamic bio-imaging analysis. The objective of this study is thus to measure the effect of displacement and deformation and strain generated by the reflex activation pelvic floor muscles. Computation of the deformation analysis is presented using the temporal and spatial components of urethral profile reflecting the deformation and strain generated by voluntary and reflex activity.

## 1 Introduction

Computation of pelvic floor muscle function constitutes an important part of the clinical evaluation and management many of clinical problems, ranging in early life from conception, birth and in the later stages of life to continence. Such an evaluation contributes not only in the diagnosis of current conditions, but provides important information that is invaluable in the prediction of a variety of dysfunctional states. Advances in ultrasound imaging enabled the visualization of the major structures contained within the pelvic cavity and is widely utilized to assess primarily the anatomical status of many organs. As such it has become one of the most widely used clinical tools to view the position and dimensions of most of the echogenic structures. While such imaging provides important anatomical information, it utilizes a static and somewhat simplistic approach devoid of functional or dynamic information. Furthermore, the complex geometry of the orientation of the fibers and the muscle group bio-mechanics require dynamic considerations to be made incorporating circumferential, radial and longitudinal directions.[1-3]

Dysfunction may be present even in the setting of what may appear normal, especially early on, in a potentially damaging process at a time when intervention might prove most helpful. Clearly such interpretation of function is usually subjective with semi-quantitative visual assessment. The problem of such interpretation is magnified when observations are made of episodes that are fast such as the cough

reflex. The importance of documenting these events is critical because it is during these reflexes that incontinence may occur in the vast number of patients. In previous observations we demonstrated using computational techniques applied to sequences of video segments the detection of subtle changes in contractile function and the generation of new parameters.[4]

Using this technique, displacement, velocity and acceleration data relative to well defined structures that were actively contracting were obtained. Clearly, strain, a representation of deformation or change in length, is an important parameter that can be computed from bio-imaging. In this presentation the potential clinical utility of strain and Strain Rate (SR) in the noninvasive assessment of pelvic floor muscles will be presented. It is expected that a case will be made as to how these parameters yields enhanced insight into dynamic as well as contractile mechanics of the pelvic floor.

Strain or elastic stiffness, a dimensionless index was first described in 1973, using the isolated heart muscle and in intact hearts, as the deformation that occurs after the application of stress.[5] Strain reflects total deformation of the myocardium during the cardiac cycle relative to its initial length. The definition and formula for this definition of strain (Lagrangian strain) is:

$$\epsilon = (L - L_0)/L_0 = \Delta L/L_0$$

$L_0$  is the original length;  $L$  new length;  $\Delta L$ , and the change in length.

Thus, as used in this presentation, strain represents deformation (change in length) of the levator muscles from their resting position to their shortening consequent to the cough reflex. Normally, the terms of longitudinal and circumferential strain demonstrate negative curves, representing a shortening of the muscles. Conversely, radial strain is a positive curve, reflecting a lengthening relative to the initial dimension. In ultrasound imaging of the heart, strain and SR can be determined by either tissue Doppler or 2D speckle tracking. The validity of this method of strain and SR calculation has been tested against sonomicrometry and three-dimensional (3D) tagged MRI strain, the invasive and clinical gold standards respectively.

## 2 Tools and Methods

Data from asymptomatic controls and were obtained using transperineal ultrasound imaging methods and stored on disk for offline analysis. Segments of video images were imported into Mimics 13.1 subroutines. The segmentations routine was used to track and measure the movements of the bladder neck with respect to both the bony pubis and anorectal angle.

A mask was first created for each anatomical structure. Subsequently, polylines were used to create outlines of the pertinent anatomical organs in each frame. Vertical and horizontal displacements were taken from the bladder neck to the symphysis and anorectal angle.

Displacement, velocity strain and acceleration were then measured. A 2-5 serial averaging smoothing function was used to filter the individual data from presentation. Cough events were recorded using the audio channel of the ultrasound image scanner and used as a timing reference point.

Analysis was done by identifying and segmenting the outline of the anatomical landmarks of bladder, urethra, symphysis pubis and AnoRectal Junction (ARJ). Displacement curves were subsequently computed between ARJ and urethra. These curves measure the change in the distance between two anatomical structures relative to the resting state. Each curve, representing an anatomical structure, can be compared to another curve along each point of its length where the opposing structure exists on a direct line of contact, allowing the transmission of mechanical forces between the two structures. Each line connecting the two structures defines a pair of points, one on each structure. The set of all pairs of points is used in constructing the displacement curves.

To identify these pairs of opposing points along each curve, we defined a central axis dividing the ARJ and the urethra. The first step in computing this axis was to convert the ultrasound image into a mask using shareholding and morphological image processing operations. The right-edge of this mask defined the input to the fitting function, which computed a 2<sup>nd</sup>-order polynomial fit. The displacements between the ARJ and the urethra were computed along the points of this central axis for which there were opposing pairs of points on the segmented ARJ and urethral lines joined by a line perpendicular to this axis. For this patient, we were able to use a single parameter to define this curve; in general, we will need two parameters for more complex anatomies.

Segmentations of the anterior and posterior surface of the urethra were also done and the Urethral Diameter Profile UDP was computed. Analogously the Vaginal Diameter Profile VDP was constructed and represented so that compression or expansion can be illustrated with respect to time.

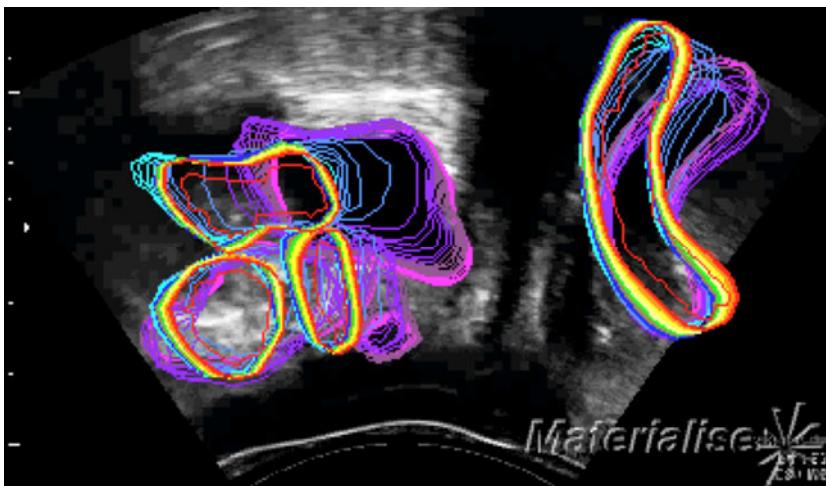
For this patient's anatomy, a one-dimensional function  $y(x)$  was sufficient to define the urethral axis, using a straight-line fit of the segmentation at each frame. The distances between the structures along the axis were computed. We ignored points on the segmentations that did not occur as pairs, connecting the two structures by a line perpendicular to the axis. In general, we will need two dimensional, parameterized curves  $(x(s), y(s))$  to define the central axes for more complex anatomies.

Data from the UDP and VDP was subsequently computed to derive the strain variation within these two structures relative to the stimulus. Animated illustrations of the biomechanical parameters can be viewed in the authors Web page. <http://danielkorenblum.com/cconst>

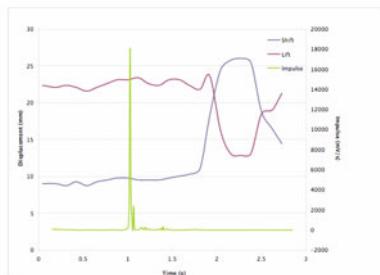
Using the above methods it is possible to characterize (a) direct biomechanical aspects of displacement, velocity and acceleration of the contraction generated as well as derive (b) visualizations that are time referenced of urethral and vaginal dimensions and strain generated in these structures.

### 3 Results

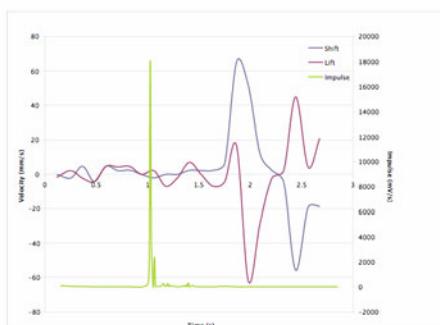
A typical segmentation of a single cough reflex is presented by Figure 1 showing the features of bladder urethral, symphysis and the anorectal region. It shows that during contraction, deformation takes place in all structures.



**Fig. 1.** Sequence of displacements of bladder urethral and rectum taking place over a time period of approximately 2 seconds. Digitized video segment was segmented and color coded over the complete coughing cycle.

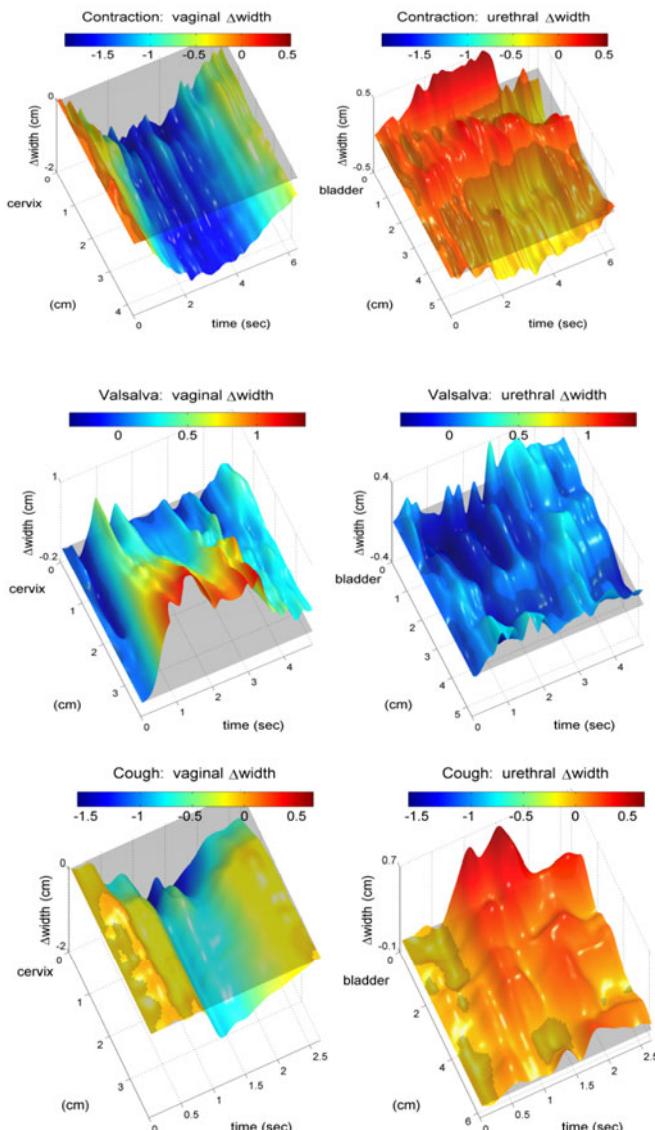


**Fig. 2.** Urethral displacement consequent to a cough. The effect of the cough stimulus to shift the urethra towards the pubis and also to lift the structure upwards is given by the two lines. The impulse line given at the 1.0 sec marking denotes the rectified audio signal of the cough.

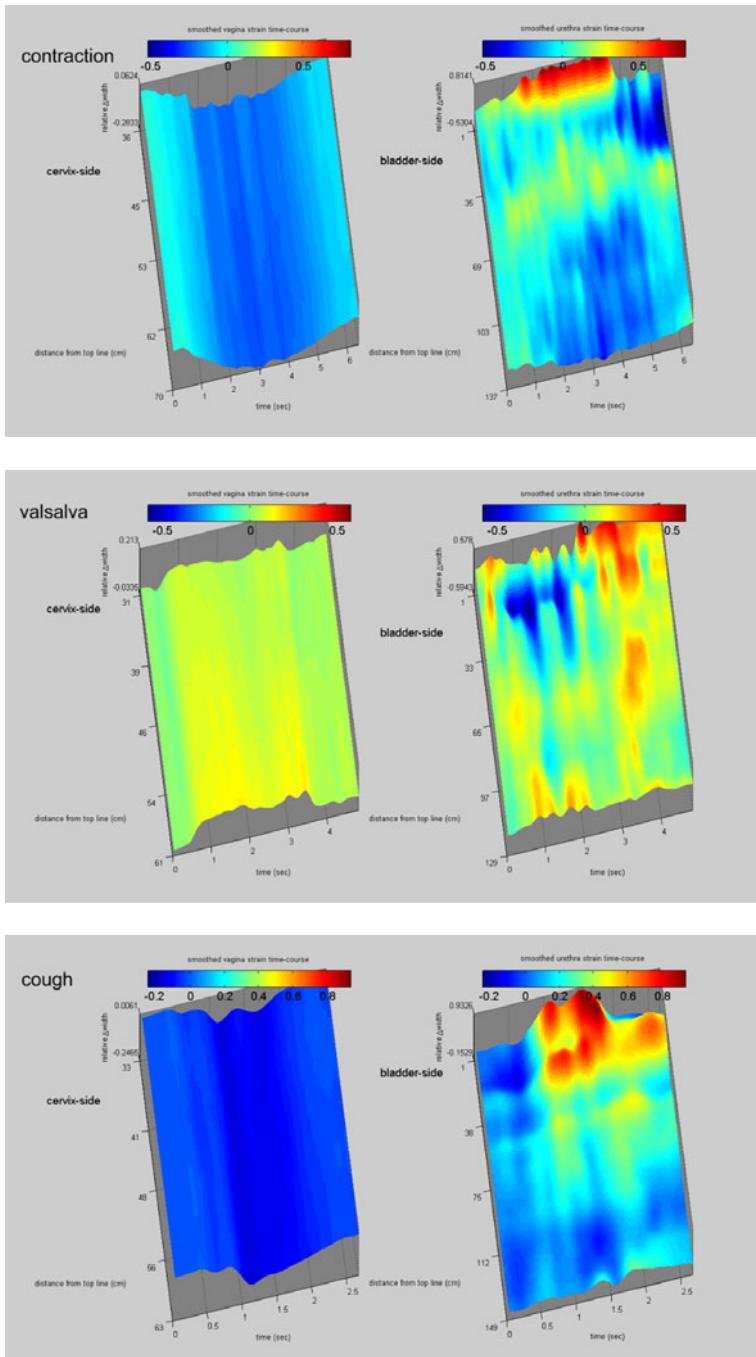


**Fig. 3.** Velocity of contraction consequent to the cough analogous to Figure 2 above using the same notation

On the basis of measurements made the displacement of the urethra at its junction with the bladder is given below by Figure 2 and the associated measure of velocity is given by Figure 3.



**Fig. 4.** Visualization time dependent of vaginal and urethral displacement profiles, VDP, UDP, as a consequent to [A] voluntary active contractions, [B] passive Valsalva maneuvers and [C] reflex response to a cough.

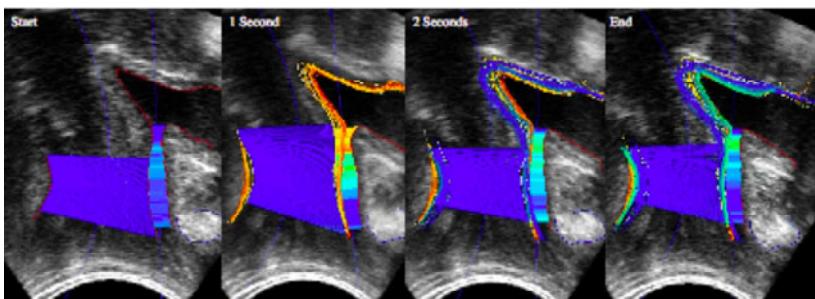


**Fig. 5.** Visualization of time dependant vaginal and urethral profile showing the strain generated by the active displacements shown above

The time dependant urethral diameter change produced by the activation of the pelvic floor muscles by voluntary contraction and resulting in compression is illustrated by Figure 4, which represents the defined UDP. As indicated by Figure 4, the UDP is given in three dimensions representing the sequence of width changes.

The visualization schema shows that the maximum compression during a contraction occurs in the region closest to the bladder. During Valsalva both the urethra and vaginal width decreases.

The corresponding time dependent profiles for the voluntary and reflex responses is given by Figure 5 which shows that the relative magnitude of urethral compression is considerably high at the region of the bladder neck and extends along the entire length of the urethra.



**Fig. 6.** Sequence of images showing the strain values within the pelvic floor and urethra. Blue indicates extension and red indicates compression. Upper plane shows the sequence of strain changes over a 3 second time segment and the lower plane the time dependent profile of stain along the urethra.

The displacements are colored to show relative stretching or compression compared to the resting state. Displacements show that the vagina was compressed during the contraction, while the urethra was stretched near the bladder and compressed at the meatus. The displacements show that the vagina was stretched. Figure 4, which illustrate the measured displacement of the segmentations defining the structures, was plotted as a surface with changes in width along the central axis shown in centimeters on the vertical scale.

The distance along the central axis where the displacement was measured is shown on the ordinate and time is shown on the abscissa. Blue corresponds to compression and red corresponds to stretching. Zero change in width is shown as a transparent gray plane. The surfaces show that the vagina was compressed while the urethra was stretched along its bladder side and slightly compressed along its meatus side during the contraction.

Compared to the process of actively recruiting pelvic floor contraction, or reflexively generating a response during a cough, pelvic floor organs also respond passively to voluntary straining. To capture the differences between these two actions, Figure 4 illustrates a complete sequence the response to voluntary PFM contraction and straining.

## 4 Discussion and Conclusion

Using this approach it was possible to identify the nature of the, urethral displacement profile and vaginal displacement profile and in terms of strain reflecting the biomechanical influence of PFM in the kinematic response of the urethra, bladder using ultrasound imaging. Visualizations of the temporal sequence of urethral closure were generated on the basis of the active reflex reaction of the anatomical displacements such as coughing, as well as the passive response to voluntarily initiated actions such as straining and contractions. Currently the results apply on the normal response of asymptomatic subjects. Consequently their response varies according to the purpose demanded and can be voluntary or triggered by reflex reactions.[6] Clearly the response in pathological cases of incontinence remains to be evaluated.[7]

Identification was made establishing some of the biomechanical factors involved in the kinematic response of some of the major contained structures, bladder, urethra and rectum using ultrasound imaging. The advantage of this visualization approach is to examine in slow motion the effect of active reflex reactions on the anatomical displacements such as coughing as well as the passive response to voluntarily initiated actions such as straining contractions.

The application of such visualizations, done under control conditions afforded by the use of asymptomatic volunteers, can be used to lay the groundwork for more extensive modeling and virtual reality constructs where parameter such as tissue properties can be used. Furthermore in such a model, the impact of virtual surgery using new material or configurations of supporting mechanisms such as meshes can be explored.

Practically an important component of the mechanism of continence is attributed to the activation of the PFM, generating zonal compression of closure pressures on the urethra and vagina. Simulation of the distribution of the forces involved can lead to a better clinical understanding of the relative contribution each individual muscle component along the length of the urethra.[8]. To develop a realistic simulation model it is essential to include, in addition to the anatomical configuration, the elastic constituents of the relevant tissue characteristics. Clearly information regarding the constitutive properties of these tissues would render the simulation model accurate and may explain the extent to which the variations observed in these simulations. Furthermore, the extent to which the vagina fits around the probe can be evaluated by implementing the collision detection model.

Clearly computational simulations of the pelvic floor are needed to better understand the cause of many of the clinical disorders encountered and it is important to combine anatomical information from image analyses with unique *in vivo* characterized force profiles for different physiologically relevant loading scenarios to the *in vitro* acquired anisotropic material property values. Although we expect the pelvic muscle to have different material property values compared to other muscles, the characteristic features of the underlying material model, having large deformations, incompressibility, non-linear anisotropic passive elasticity, permanent inelastic deformation upon non-physiological overstretch, and superposed anisotropic active muscle contraction, need to be considered from a biomechanical point of view.

Future analysis using specifically written finite element approach will require the complex interplay between active and passive muscle forces and the long-term adaptation in response changes, for example caused by surgical removal of tissue or implantation of tissue graft. Furthermore the incorporation of the urethra and sphincter should be included since it is through the urethra that leakage takes place in subjects with stress urinary incontinence. What is ultimately needed is to integrate geometry, forces, and material property values, to provide a unique virtual test-bed for probing different treatment strategies for pelvic floor disorders and for optimizing surgical process parameters before testing them in humans. In such a model, simulations can be carried out incorporating the influence of posture as well as active forces produced reflexively and voluntarily.

The transmission characteristics of closure forces to the bladder and urethra can be a significant contributory factor in the prevention of incontinence. Finally the outcome of virtual operating procedures using different materials can be simulated and their suitability considered within an objective framework. To achieve this goal it is essential to also incorporate the remainder of the anatomical structures within the pelvic cavity within the context of finite element analysis protocols.

## Acknowledgements

We are grateful to Daniel Korenblum and Rashmi Sridhara for their assistance in the analysis. Study was supported by NIH grant R01 EB006170 to CEC and the Canadian Institute for Health Research MOP-97848 to LMc.

## References

1. D'Aulignac, D., Martins, J.A.C., Pires, E.B., Mascarenhas, T., Natal Jorge, R.M.: A shell finite element model of the pelvic floor muscles. *Computer Methods in Biomechanics and Biomedical Engineering* 8(5), 339–347 (2005)
2. Parente, M.P.L.R., Natal Jorge, M., Mascarenhas, T., Fernandes, A.A., Martins, J.A.C.: Deformation of the pelvic floor muscles during a vaginal delivery. *Int. Urogynecol. J.* 19, 65–71 (2008)
3. Chen, L., Ashton-Miller, J.A., DeLancey, J.O.: A 3D finite element model of anterior vaginal wall support to evaluate mechanisms underlying cystocele formation. *Journal of Biomechanics* 42, 1371–1377 (2009)
4. Constantinou, C.E.: Dynamics of the Female Pelvic Floor. *Int. Journal Computational Vision and Biomechanics* (1), 69–81 (2007)
5. Mirsky, I., Parmley, W.W.: Assessment of passive elastic stiffness for isolated heart muscle and the intact heart. *Circ. Res.* 33, 233–243 (1973)
6. Constantinou, C.E., Govan, D.E.: Spatial distribution and timing of transmitted and reflexly generated urethral pressures in the healthy female. *J. Urol.* 127, 964–969 (1982)
7. Madill, S., McLean, L.: Quantification of abdominal and pelvic floor muscle synergies in response to voluntary pelvic floor muscle contractions. *Journal of Electromyography and Kinesiology* (2007)
8. Madill, S.J., McLean, L.: Relationship between abdominal and pelvic floor muscle activation and intravaginal pressure during pelvic floor muscle contractions in healthy continent women. *Neurourology and Urodynamics* 25, 722 (2006)

# Exploiting Multiple Cameras for Environmental Pathlets

Kevin Streib and James W. Davis

Dept. of Computer Science and Engineering  
Ohio State University, Columbus, OH, 43210  
`{streib, jwdavis}@cse.ohio-state.edu`

**Abstract.** We present a novel multi-camera framework to extract reliable pathlets [1] from tracking data. The proposed approach weights tracks based on their spatial and orientation similarity to simultaneous tracks observed in other camera views. The weighted tracks are used to build a Markovian state space of the environment and Spectral Clustering is employed to extract pathlets from a state-wise similarity matrix. We present experimental results on five multi-camera datasets collected under varying weather conditions and compare with pathlets extracted from individual camera views and three other multi-camera algorithms.

## 1 Introduction

An important task in video surveillance is to observe/model an environment and extract behavioral trends. This is typically done by collecting data for extended periods of time and extracting pathway regions (trajectory clusters [2-4], semantic regions [5, 6], pathlets [1]) using either trajectory or feature-based approaches. Generally these algorithms exploit data from single cameras and the extracted regions typically correspond to locations in the image where motion occurs. If these regions were projected to an orthophoto (i.e., aerial top-down image as seen in Google maps), locations corresponding to non-ground plane motion (e.g., head of a pedestrian) would project to locations other than the ground plane area where the object was actually moving. Thus, many of the extracted regions describe only 2D sensor-view patterns. What is really desired are pathway regions corresponding to ground-plane areas of traffic.

One solution to the erroneous region projection could be to use tracks solely from the area of objects in contact with the ground. However, finding such locations is generally not straightforward, nor would it be feasible to perform in real-time with environments containing large amounts of moving objects. As many surveillance systems contain multiple cameras viewing the same area from different vantage points, we exploit the data collected from multiple cameras to generate refined pathway regions to better capture the actual pathway regions. We choose to map the cameras to an orthophoto, instead of to a single camera view, because they do not suffer from projective distortions which causes image pixels to represent varying amounts of spatial data.

The camera network we utilize encompasses a crowded urban environment where receiving a single long track per object for its duration through the scene is infeasible in real-time. Consequently, we employ the Kanade-Lucas-Tomasi (KLT) tracker

[7], which results in multiple fragmented tracks per object but is capable of tracking hundreds of features simultaneously in real-time. A pathway analysis method suited to this type of tracking data is the approach of [1], which extracts “pathlets” (coherent motion regions containing tracks with similar origin and destination) of a scene from KLT tracks. In this paper, we present a novel extension to [1] which combines information from multiple cameras to extract the pathlets. The approach is based on using tracks from other cameras to vote for or remove tracks from a given camera view. We evaluate our proposed approach on multiple datasets, and compare the results with pathlets extracted using the tracking data from the individual camera views and three other multi-camera approaches suited to the task.

## 2 Related Work

The majority of surveillance-related research on modeling scene behavior has focused on using a single camera. These works often use either trajectory [1-4] or feature-based [5, 6] approaches to extract trajectory clusters or semantic motion regions of a scene. Trajectory methods have included envelope approaches to determine if tracks should be assigned to existing routes or formed into new routes [3], vector quantization to reduce trajectories to a set of prototypes [2], and Spectral Clustering on pairwise trajectory similarity matrices [4]. Alternatively, optical flow [6] and combinations of optical flow and appearance metrics [5] can be used to analyze scene activity without relying on tracking.

Recently, research has begun focusing on utilizing entire camera networks rather than single cameras. Data is fused from 2D image and 3D world coordinates in [8] to track objects between cameras. In [9] a camera network topology is estimated and targets are tracked across blind areas within the camera network by associating activities across camera views. In [10] objects are tracked in partially overlapping cameras, and the extracted features are mapped to a ground plane and associated across views to track targets through the camera network. Information is combined across all cameras in [11] and the ground plane location corresponding to feet are found to track people in crowded scenes. In [12, 13] correlation of activity regions from different cameras are modeled and used to detect global activity anomalies. Features are extracted from objects tracked independently in each camera view in [14] and the distribution of activities across camera view feature spaces are learned to group trajectories belonging to the same activity and model paths taken across camera views.

While several methods have been proposed to utilize multiple cameras to either track objects for extended durations or correlate activities across camera views, to the best of our knowledge the work presented in this paper is the first approach which exploits data from multiple cameras to extract meaningful pathway regions that more accurately describe where objects are moving than regions derived using a single camera.

## 3 Multi-camera Pathlet Extraction

We based our approach on the pathlet method of [1], that is specifically designed to handle weak tracking data. The algorithm in [1] overlays an  $L \times L$  grid onto a scene and

quantizes tracks into states, where each state  $s_i = [(x, y), \theta]$  is defined as a grid cell location and the quantized angle of the track through the cell. Once quantized, the tracks are used to count the transitions from each state to states in their 8-connected neighboring cells. These counts are then normalized to generate a Markovian state transition model for each state. Next, the ratio of number of tracks that enter/leave a state versus start/stop at the state are used to find the entry/exit probabilities for each state close to the border of regions where tracks exist. The result is a probabilistic scene model of where tracks enter the scene, how they transition throughout the scene, and where they exit the scene. This model is employed to sample tracks (longer than the original weak tracks) which are used to determine where tracks through each state typically originate and terminate. Finally, the original weak tracks are used to produce a trace of the count of tracks in each state across time. A state-wise similarity matrix is constructed based on the origin/destination and temporal cross-correlation of tracks through the states, and the Spectral Clustering algorithm of [15] is employed to extract pathlets from the scene. See [1] for additional details.

We extended the algorithm in three ways to enable multi-camera fusion. First, the Markovian transitions between states are adapted to handle confidence weights assigned to the tracks to enable scene modeling using tracks with varying believability. Second, we removed the constraint that entry and exit states be close to the border of regions where tracks exist to eliminate the necessity of finding the border around motion regions, and instead use the sum of track weights entering a state ( $N_{in}$ ) and exiting a state ( $N_{out}$ ) to calculate the entry and exit weights ( $\mathcal{W}_E$  and  $\mathcal{W}_X$ ) for *any* state as

$$\mathcal{W}_E = \frac{N_{out}}{\left(1 + \exp\left(-\frac{\frac{N_{out}}{N_{out}+N_{in}} - \mu}{\sigma}\right)\right)}, \quad \mathcal{W}_X = \frac{N_{in}}{\left(1 + \exp\left(-\frac{\frac{N_{in}}{N_{out}+N_{in}} - \mu}{\sigma}\right)\right)}, \quad (1)$$

where  $\mu$  and  $\sigma$  are the mean and standard deviation of a CDF of a logistic distribution (set to 0.75 and 0.05 in our experiments), respectively. We use the above formulation as it is a more flexible method than presented in [1]. Finally, since the temporal cross-correlation used in [1] was empirically found to have little affect on the extracted pathlets, we define the state similarity matrix using solely the origin and destination similarity of tracks through the states.

Our proposed multi-camera method projects the tracks from all cameras to an orthophoto (via homography) and computes a weight for each track based on its spatial and orientation similarity to temporally overlapping tracks from the remaining cameras. The intuition for this approach can be explained by considering the ideal case of tracking an individual with two cameras that are  $180^\circ$  apart and registered to the orthophoto with a planar homography. Intuitively, KLT tracks from the two cameras corresponding to features close to the feet of the person will result in projected tracks on the ground plane location [11]. Feature points near the head of the person will project incorrectly (they violate the planar assumption). Thus, if projected tracks in one camera view are close in proximity and travel in the same direction as temporally overlapping tracks from other cameras, they are more likely to correspond to tracks closer to the ground, thereby more accurately describing the true paths in the environment.

The procedure for determining  $w(t_i^1)$ , the weight of track  $i$  in Camera-1 from all other cameras, is as follows. First,  $\alpha(t_i^1, t_j^2)$ , the pairwise track weight for each temporally overlapping track  $t_j^2$  in Camera-2, is computed as

$$\alpha(t_i^1, t_j^2) = \frac{1}{\tau_2 - \tau_1 + 1} \sum_{k=\tau_1}^{\tau_2} v(t_i^1[k], t_j^2[k]) \cdot \exp\left(-\frac{\|t_i^1[k] - t_j^2[k]\|}{\sigma}\right), \quad (2)$$

within the temporal overlap  $\tau_1$  to  $\tau_2$ , and where the binary orientation similarity

$$v(a, b) = \begin{cases} 1 & |\angle(a) - \angle(b)| \leq 15^\circ \\ 0 & \text{else} \end{cases} \quad (3)$$

considers two tracks to be traveling in the same direction at time  $k$  if the direction of their instantaneous velocities are within  $15^\circ$ . We use a value of  $\sigma = 4$  for the exponential weighting in Eq. 2 in all of our experiments.

Next, we use a greedy algorithm to determine  $\beta_2(t_i^1)$ , the weight of  $t_i^1$  from *all* tracks in Camera-2. First, all elements in an indicator vector  $x$  used to keep track of matched observations in  $t_i^1$  are initialized to zero. Then, while there are unmatched observations (to  $t_i^1$ ) and unmatched tracks from Camera-2 that have temporally co-occurring observations,  $\beta_2(t_i^1)$  is incremented by  $\frac{m}{M} \cdot \alpha(t_i^1, t_{jj}^2)$ , where  $t_{jj}^2$  is the unmatched track from Camera-2 resulting in the maximum track weight  $\alpha(t_i^1, t_{jj}^2)$ ,  $m$  is the number of matched observations between the two tracks (i.e., temporally co-occurring and previously unmatched in  $x$ ), and  $M$  is number of observations in  $t_i^1$ . The indicator vector  $x$  is then updated to reflect the observations that were just matched. Thus,  $\beta_2(t_i^1)$  is a weighted average of the tracks from Camera-2 that get matched to  $t_i^1$  based on the number of elements they match to in  $x$ .

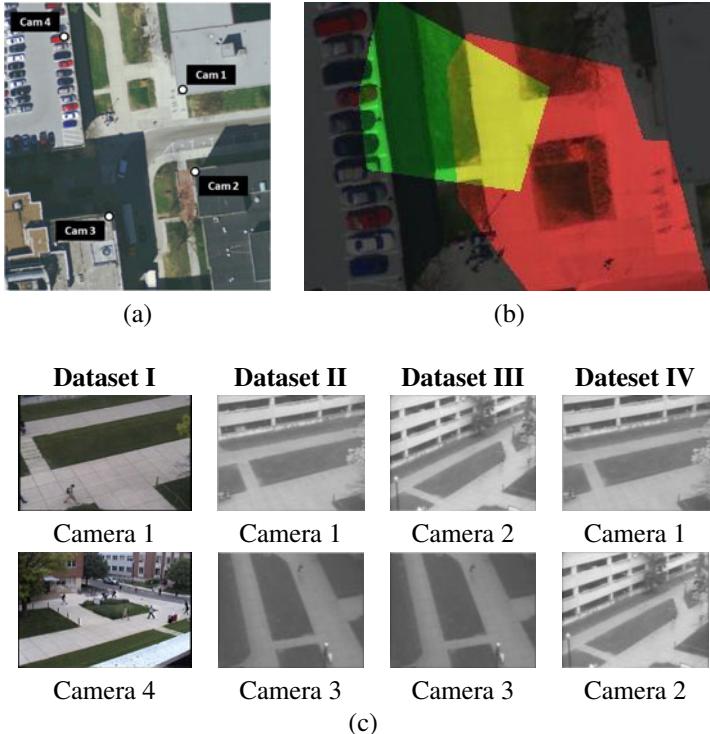
The final weight of track  $i$  in Camera-1,  $w(t_i^1)$ , is calculated as the product of the camera weights for  $t_i^1$  across all cameras

$$w(t_i^1) = \prod_{z \neq 1} \beta_z(t_i^1). \quad (4)$$

Thus, the weight of a track will increase as the support from tracks in other cameras increases.

## 4 Experimental Results

We tested the proposed multi-camera pathlet extraction method on four two-camera datasets and one three-camera dataset from the camera locations shown in Fig. II(a), whose corresponding camera views are shown in Fig. II(c). The three-camera dataset, Dataset V, is composed of track data from the three camera views used in Datasets II-IV. Track information was collected on different days with varying weather conditions. Dataset I consists of tracks collected from Cameras 1 and 4 for approximately fifteen minutes on a sunny day. Datasets II-V consist of tracks collected from Cameras 1, 2, and 3 for six hours on a rainy day.



**Fig. 1.** (a) Camera locations of four cameras used for the datasets. (b) Individual and overlapping coverage of Cameras 1 and 4. (c) Camera views for Datasets I-IV. (Best viewed in color.)

#### 4.1 Alternate Approaches

We present three alternate methods in addition to our proposed approach to extract pathlets using data from multiple cameras and compare the results. In all of the methods, projections between the camera and ortho-space are performed using a planar homography [16] (i.e., a plane-to-plane mapping) from manual correspondence, which results in a projection matrix to transfer between coordinate systems. Using the homography, the camera views are mapped to a common orthophoto for each dataset to determine the overlapping ground region.

The first method (Method A) is a naive approach which projects the tracks from all cameras to the orthophoto. Tracks are given equal unitary weight and pathlets are extracted from a state space model built in the ortho-space. This method is the same as the proposed method but with no track weighting scheme.

The second method (Method B) projects the tracks from each camera separately to the orthophoto (again assigning each track with a unitary weight), and extracts pathlets using the tracking data from each individual camera alone. The pathlets extracted from each individual camera are then intersected together, resulting in a set of combined pathlets. The intersection algorithm generates a label matrix (number of states  $\times$  number of cameras) where each element  $[i, j]$  corresponds to the pathlet ID that state

$i$  belongs to from the pathlets extracted using the tracking data from camera  $j$ . Each combined pathlet consists of all states having the same label combination across the individual camera pathlets, where a state must be labeled in a pathlet from each camera to persist. Thus, the number of combined pathlets is equivalent to the number of unique rows of the label matrix containing no zeros.

The third method (Method C) extracts the pathlets for the tracking data from each camera (assuming unitary track weight) in the camera-space and projects the resulting pathlets from each individual view to the orthophoto. The pathlet projection algorithm maps each state pixel/angle combination in the camera-space to a pixel/angle combination in the ortho-space. The corresponding ortho-space pathlets consist of all states whose respective cells are at least 50% populated. The ortho-space pathlets are computed using an intersection algorithm similar to the algorithm used in Method B. To account for slight projection deformations, the intersection algorithm is relaxed by allowing combinations of states from ortho-space pathlets from different cameras if their orientations are within  $45^\circ$ .

Our proposed approach (Sect. 3) will be referred to as Method D.

## 4.2 Optimal Pathlet Selection

To provide the maximal results for the different approaches, we first define the region-of-interest (ROI) in the orthophoto. For our datasets (Fig. 1(c)) we define the sidewalks as the ROIs as these should be the only locations with moving targets. We then overlay the grid used to build the state space model (Sect. 3) and define grid cells as positive if more than 50% of its pixels are within the ROI in the overlapping camera region and negative otherwise. For Method C the boundaries of the overlapping camera region and ROI are mapped back to the individual camera views, where positive and negative cells are defined based on the respective grid sizes used for each camera view.

Since it is feasible for states to contain insufficient information from training to build an accurate Markovian transition model, we remove states with low weight (where a state's weight is equivalent to the sum of the track weights for the tracks that are mapped to it) from the state space model. First, we sort the states in descending order of weight and generate a sorted cumulative distribution function (CDF). Again, our goal is to generate pathlets that describe the environment (i.e., walkways), rather than the scene from the individual camera view. Thus, we want to include states from positive cells and remove states from negative cells. After state removal, let  $T_p$ ,  $F_p$ , and  $F_n$  be the number of positive cells with states, the number of negative cells with states, and the number of positive cells without states, respectively. Then, precision ( $P = T_p / (T_p + F_p)$ ) measures the percentage of kept cells that belong to the ROI, while recall ( $R = T_p / (T_p + F_n)$ ) measures the percentage of cells belonging to the ROI which are kept. Since precision can be high with a low recall and vice-versa, we use the F-measure ( $F = 2 \cdot \frac{P \cdot R}{P + R}$ ), which combines precision and recall into a single metric that is maximized when precision and recall are jointly maximized, to determine the quality of the kept states. To optimize the pathlets (for each method), we find the F-measure resulting from the highest weighted states kept if the sorted CDF is thresholded at  $X\%$ , and keep the states based

on the threshold resulting in the maximum F-measure. Thus, the threshold is designed to maximize the states kept from positive cells while minimizing states kept from negative cells.

The output of each multi-camera fusion algorithm is a set of pathlets describing the environment in the ortho-space. We then project the ortho-space pathlets back to the original camera views. We overlay the individual camera view grid on each projected pathlet and determine its spatial extent in the camera view grid space by only keeping cells which are at least 50% covered by the projected cluster. The camera view pathlets are run through a connected components algorithm, separating disjoint pathlets into a set of connected pathlets, and small pathlets (those containing less than 5 cells) are removed, yielding the final set of camera view pathlets. We use a  $10 \times 10$  pixel grid for all cameras and datasets except for Camera-4 in Dataset I (where we use a  $20 \times 20$  pixel grid because of the zoom factor). A  $5 \times 5$  pixel grid is used in the ortho view for all datasets.

### 4.3 Results

We use precision, recall, F-measure, and  $N_p$  (the number of pathlets containing at least 50% of their cells within the overlapping camera region) as the primary means to quantitatively compare the pathlets extracted from the different approaches. Intuitively,  $N_p$  provides a rough comparison to determine if an environment is relatively under- or over-segmented. Tables 1 and 2 show the four quantitative measurements corresponding to the threshold which optimizes the F-measure of the pathlet extraction methods (as described in Sect. 4.2) for each dataset.

**Table 1.** Quantitative measurements of pathlets extracted using tracking data from individual camera views on five datasets. (Cam-x / Cam-y / Cam-z)

Dataset	$N_p$	Precision	Recall	F-Measure
I	2 / 6	0.82 / 0.82	0.96 / 0.93	0.88 / 0.87
II	9 / 7	0.64 / 0.75	0.97 / 0.85	0.77 / 0.80
III	8 / 14	0.69 / 0.76	0.85 / 0.92	0.77 / 0.83
IV	9 / 4	0.63 / 0.65	0.96 / 0.86	0.76 / 0.74
V	10 / 6 / 6	0.64 / 0.66 / 0.75	0.96 / 0.85 / 0.91	0.77 / 0.75 / 0.82

Table 1 shows that recall is typically much higher than precision for pathlets extracted from each individual camera. This suggests that the pathlets extracted from a single camera cover a majority of the ROI, but also bleed outside the ROI, resulting in non-walkway pathlets.

Based on Table 2 Method A generally has the least precision of the multi-camera fusion methods, which is expected given the naive nature of the algorithm. Furthermore, the method typically results in a higher recall than precision, meaning it tends to generate pathlets which cover, yet bleed outside the ROI.

Methods B and C generally result in the highest precision of the multi-camera fusion algorithms. This result is expected since both methods use the intersection algorithm

**Table 2.** Quantitative measurements of pathlets extracted using four multi-camera fusion algorithms on five datasets. Bold font represents the highest F-measure received in each camera for the dataset. (Cam-x / Cam-y / Cam-z)

Dataset	Method	$N_p$	Precision	Recall	F-Measure
I	A	8 / 12	0.80 / 0.87	0.91 / 0.90	0.85 / 0.89
	B	6 / 7	0.95 / 0.97	0.75 / 0.72	0.83 / 0.82
	C	8 / 8	0.88 / 0.94	0.87 / 0.85	<b>0.88</b> / 0.89
	D	8 / 8	0.82 / 0.90	0.93 / 0.91	0.87 / <b>0.91</b>
II	A	4 / 4	0.84 / 0.74	0.72 / 0.70	0.77 / 0.72
	B	6 / 5	0.93 / 0.84	0.73 / 0.71	0.82 / 0.77
	C	8 / 7	0.92 / 0.85	0.77 / 0.76	0.84 / 0.80
	D	4 / 5	0.89 / 0.77	0.85 / 0.86	<b>0.87</b> / <b>0.81</b>
III	A	5 / 6	0.78 / 0.70	0.83 / 0.84	0.80 / 0.76
	B	8 / 8	0.89 / 0.85	0.69 / 0.70	0.78 / 0.77
	C	12 / 13	0.93 / 0.86	0.75 / 0.73	0.83 / 0.79
	D	8 / 9	0.86 / 0.79	0.91 / 0.91	<b>0.89</b> / <b>0.85</b>
IV	A	5 / 4	0.64 / 0.59	0.94 / 0.92	0.77 / 0.72
	B	4 / 4	0.70 / 0.65	0.74 / 0.72	0.72 / 0.68
	C	10 / 7	0.73 / 0.65	0.84 / 0.80	<b>0.78</b> / 0.72
	D	5 / 4	0.74 / 0.69	0.81 / 0.79	0.77 / <b>0.74</b>
V	A	5 / 5 / 5	0.67 / 0.62 / 0.58	0.95 / 0.92 / 0.94	0.78 / 0.74 / 0.71
	B	5 / 4 / 6	0.93 / 0.90 / 0.83	0.70 / 0.67 / 0.73	0.80 / 0.77 / 0.77
	C	18 / 11 / 14	0.97 / 0.96 / 0.87	0.73 / 0.71 / 0.69	0.83 / 0.81 / 0.77
	D	6 / 5 / 5	0.85 / 0.82 / 0.76	0.89 / 0.87 / 0.86	<b>0.87</b> / <b>0.84</b> / <b>0.81</b>

described in Sect. 4.1. Consider an ideal case where two camera views are  $180^\circ$  apart focusing on a single sidewalk which lies between the cameras (e.g., similar to Dataset I in Fig. 1(b)). In this scenario tracks from one camera will cover the sidewalk and overlap to the right of the sidewalk, while tracks from the other camera will cover the sidewalk and overlap to the left of the sidewalk. Thus, the intersection of the corresponding pathlets will lie directly on the sidewalk resulting in a very high precision. However, since states must have been kept in all camera views to be present in the final combined pathlets, the intersection algorithm used in Methods B and C also generally causes the recall to be lower than in Methods A and D.

Method D, our proposed method, tends to balance precision and recall more than the other multi-camera fusion methods. Furthermore, it results in the highest average F-measure for all five datasets and the highest F-measure in all the datasets' individual camera views except for two cases, where it is only 0.01 less than best the F-measure received from Method C. However, in most cases Method D requires less pathlets to model the environment than Method C (see  $N_p$  in Table 2), which suggests that Method C is prone to generating over-segmented pathlets.

## Qualitative Analysis

Figure 2 shows the pathlets extracted for Dataset I using the KLT tracking data from each individual camera and from the multi-camera fusion methods using the highest

weighted states resulting in the optimized F-measure (as described above). In each image the white area, black lines, and colored blobs correspond to the overlapping camera region, ROI outline, and extracted pathlets, respectively.

The pathlets extracted using the tracking data from both individual cameras capture the bi-directionality of the main sidewalk inside the overlapping camera region. However, tracks on the upper body of pedestrians cause the pathlets to extend beyond the sidewalk and outside the ROI. Furthermore, there are additional pathlets in both camera views outside the overlapping camera region which extend beyond the camera's ROI due to the projection of far-field data.

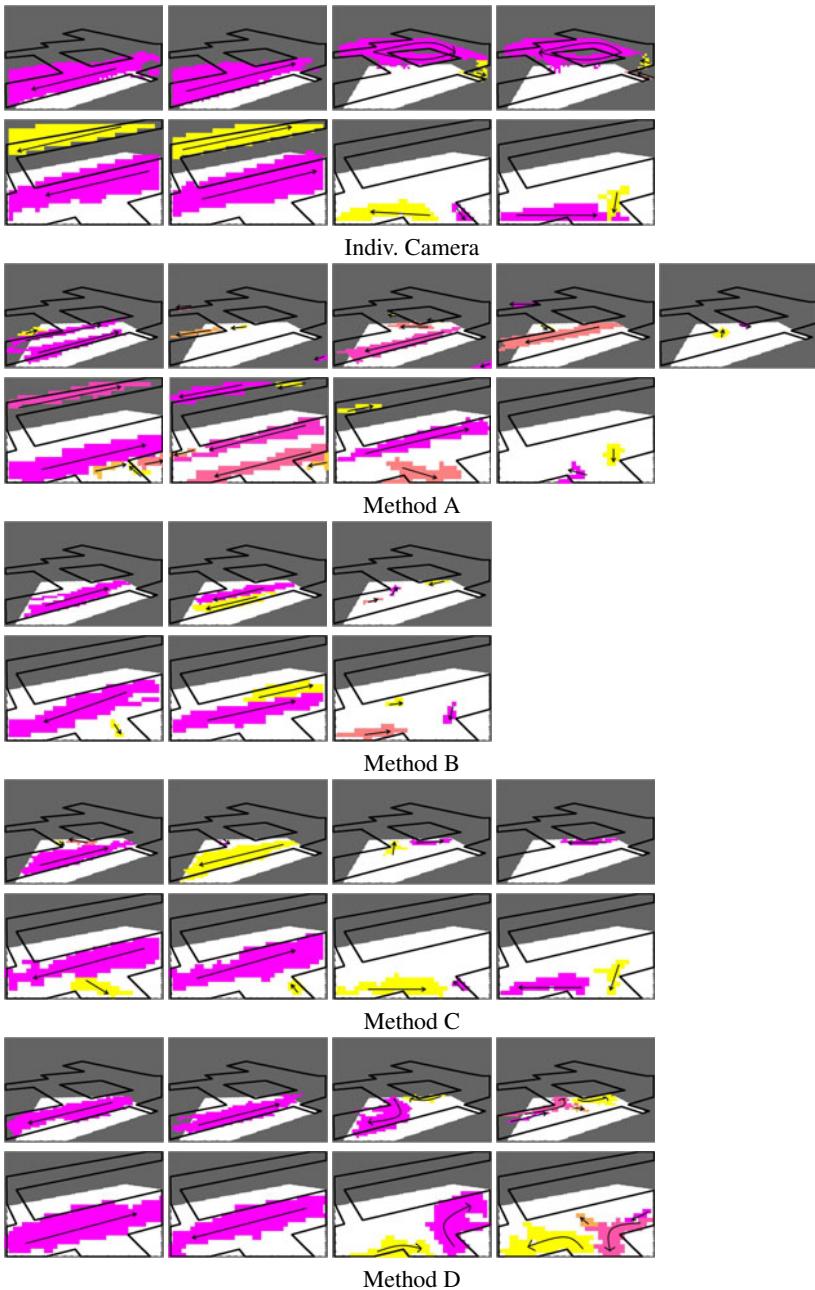
Method A combines the track data from the two individual cameras after they are projected to a common ortho-space. As shown in Fig. 2(b), a majority of the overlap from Cameras 1 and 4 occurs within the ROI. Consequently, the states in these areas will contain more tracks, resulting in a higher contribution to the total state weight. For this particular scenario this artifact of the camera overlap removes a majority of the false positive cells when optimizing the F-measure, resulting in pathlets that are primarily inside the ROI for the overlapping camera region. However, Fig. 2 clearly shows that Method A over-segments the scene, using multiple pathlets to describe traffic traveling in the same direction.

The intersection algorithm used in Method B causes the primary sidewalk to be over-segmented in both camera views. Interestingly, this over-segmentation does not occur in Method C which also uses an intersection algorithm. This is likely a result of the relaxation described in Sect. 4.1 to deal with projection deformations. Method C also produces pathlets for traffic entering/exiting the main sidewalk from/to an adjoining sidewalk, which were only extracted from Camera-4 but kept because of the intersection algorithms.

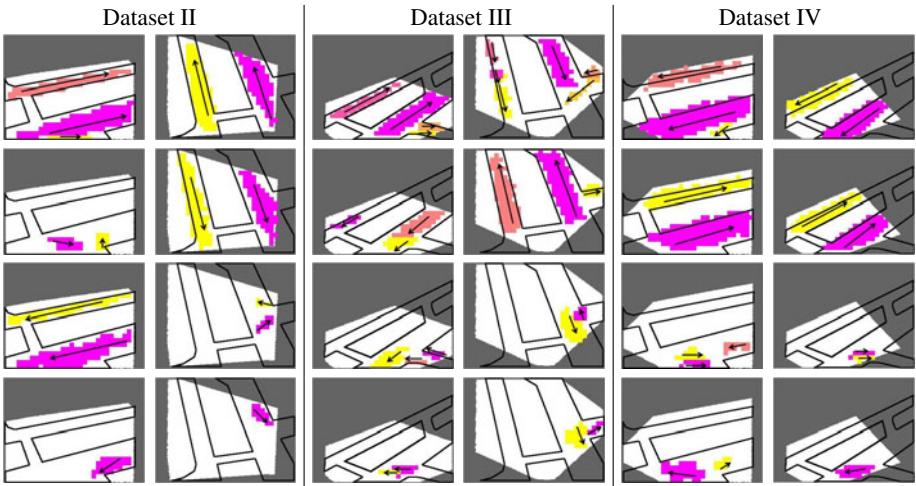
Method D (our proposed approach) correctly yields one distinct pathlet to define each direction of the main sidewalk, and also extracts pathlets for traffic entering/exiting from/to the adjoining sidewalk.

We focus on Methods C and D from now on since they are both quantitatively and qualitatively superior to the other approaches. Figures 3 and 4 show the pathlets extracted from Datasets II-IV for Methods C and D (our proposed method), respectively.

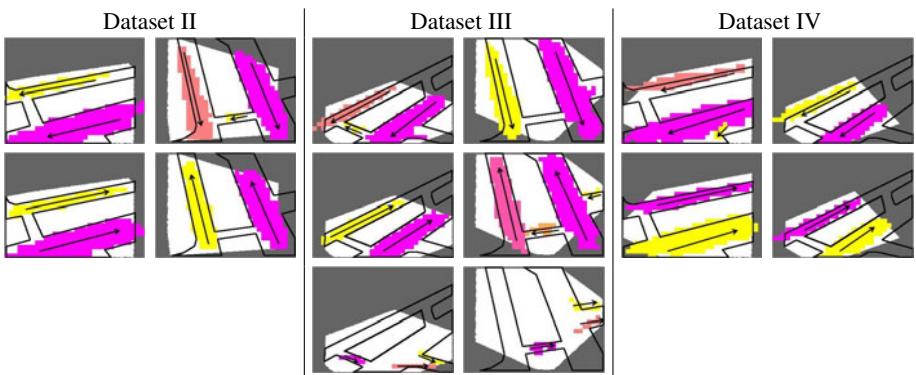
In Dataset II both algorithms extract pathlets to capture the bi-directionality of the two primary sidewalks. Furthermore, Method D captures movement on the less frequently traveled secondary sidewalk which connects the two primary sidewalks, which Method C fails to do. In Dataset III, Method C severely over-segments the scene while Method D again captures the bi-directionality of the primary sidewalks with single pathlets in each direction. Furthermore, Method D also captures a pathlet describing the connecting sidewalk and pathlets describing traffic entering and exiting from the 'T' junction. In Dataset IV both methods capture the bi-directionality of the primary walkways with Method C generating more pathlets describing tracks entering and exiting the 'T' junction. In Dataset V (not shown due to space constraints) Method D outperforms Method C which drastically over-segments the environment (see  $N_p$  in Table 2). Combining the quantitative measurements in Table 2 with the tendency Method C has for over-segmenting an environment, it is apparent that our proposed approach (Method D) performs the best at extracting environmental pathlets for the datasets.



**Fig. 2.** Extracted pathlets using tracking data from the individual cameras and the four multi-camera information fusion algorithms for Dataset I. (Best viewed in color.)



**Fig. 3.** Extracted pathlets using Method C for Datasets II-IV. (Best viewed in color.)



**Fig. 4.** Extracted pathlets using our proposed method (Method D) for Datasets II-IV. (Best viewed in color.)

## 5 Summary

We presented a novel approach to employ data from multiple cameras to generate pathlets. Our proposed approach weights tracks based on their spatial and orientation similarity to tracks collected simultaneously in other cameras. The weighted tracks are used to build a Markovian state space model and Spectral Clustering is utilized to extract pathlets from a state-wise similarity matrix based on the origin/destination of tracks through the states. We compared our approach with pathlets extracted from the individual camera views and from three other multi-camera algorithms on five multi-camera datasets collected under varying conditions. Finally, we showed quantitatively

and qualitatively that our proposed method outperforms the other methods. This research was supported in part by the US Air Force Research Laboratory Human Effectiveness Directorate (WPAFB) under contract No. FA8650-07-D-1220.

## References

1. Streib, K., Davis, J.W.: Extracting pathlets from weak tracking data. In: Proc. AVSS (2010)
2. Stauffer, C., Grimson, W.E.L.: Learning patterns of activity using real-time tracking. *IEEE TPAMI* 22, 747–767 (2000)
3. Makris, D., Ellis, T.: Automatic learning of an activity-based semantic scene model. In: Proc. AVSS (2003)
4. Wang, X., Tieu, K., Grimson, W.E.L.: Learning semantic scene models by trajectory analysis. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *ECCV 2006*. LNCS, vol. 3953, pp. 110–123. Springer, Heidelberg (2006)
5. Li, J., Gong, S., Xiang, T.: Scene segmentation for behaviour correlation. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008*, Part IV. LNCS, vol. 5305, pp. 383–395. Springer, Heidelberg (2008)
6. Wang, X., Ma, X., Grimson, E.: Unsupervised activity perception in crowded and complicated scenes using hierarchical bayesian models. *IEEE TPAMI* 31, 539–555 (2009)
7. Shi, J., Tomasi, C.: Good features to track. In: Proc. CVPR (1994)
8. Black, J., Ellis, T., Rosin, P.: Multi view image surveillance and tracking. In: IEEE Workshop on Motion and Video Computing (2002)
9. Makris, D., Ellis, T., Black, J.: Bridging the gaps between cameras. In: Proc. CVPR (2004)
10. Anjum, N., Cavallaro, A.: Trajectory association and fusion across partially overlapping cameras. In: Proc. AVSS (2009)
11. Khan, S.M., Shah, M.: A multiview approach to tracking people in crowded scenes using a planar homography constraint. In: Leonardis, A., Bischof, H., Pinz, A. (eds.) *ECCV 2006*. LNCS, vol. 3954, pp. 133–146. Springer, Heidelberg (2006)
12. Loy, C.C., Xiang, T., Gong, S.: Time-delayed correlation analysis for multi-camera activity understanding. *Intl. Journal of Computer Vision* (2010)
13. Li, J., Gong, S., Xiang, T.: Discovering multi-camera behaviour correlations for on-the-fly global activity prediction and anomaly detection. In: IEEE Intl. Workshp on Visual Surveillance (2009)
14. Wang, X., Tieu, K., Grimson, E.: Correspondence-free activity analysis and scene modeling in multiple camera views. *IEEE TPAMI* 32, 56–71 (2010)
15. Zelnik-Manor, L., Perona, P.: Self-tuning spectral clustering. In: NIPS (2004)
16. Criminisi, A., Reid, I., Zisserman, A.: A plane measuring device. *Image and Vision Computing* 17, 625–634 (1999)

# On Supervised Human Activity Analysis for Structured Environments

Banafshe Arbab-Zavar, Imed Bouchrika, John N. Carter, and Mark S. Nixon

School of Electronics and Computer Science, University of Southampton,  
Southampton, SO17 1BJ, UK

**Abstract.** We consider the problem of developing an automated visual solution for detecting human activities within industrial environments. This has been performed using an overhead view. This view was chosen over more conventional oblique views as it does not suffer from occlusion, but still retains powerful cues about the activity of individuals. A simple blob tracker has been used to track the most significant moving parts i.e. human beings. The output of the tracking stage was manually labelled into 4 distinct categories: walking; carrying; handling and standing still which are taken together from the basic building blocks of a higher work flow description. These were used to train a decision tree using one subset of the data. A separate training set is used to learn the patterns in the activity sequences by Hidden Markov Models (HMM). On independent testing, the HMM models are applied to analyse and modify the sequence of activities predicted by the decision tree.

## 1 Introduction

Automated detection and tracking human activities within video sequences is a challenging problem which finds application in monitoring and surveillance systems as well as human-machine interactions. Recently, parallel to advances in video camera technologies as well as storage and computation capabilities, there has been an increase of research interest in the area of human action recognition in the computer vision community.

Various types of features have been proposed for this task. Parameswaran et al. [1] detects a number of body joints and analyses their trajectories in 2D invariance space. Detecting and tracking body parts have also been used to infer the higher level activities [2][3]. In this, state space methods have been employed to analyse a sequence of lower level events. Rather than tracking various body parts or joints, other methods have used holistic features [4], and local spatio-temporal interest points [5][6]. Sun et al. [7] experimented with both holistic features and local interest points and showed that the effectiveness of these features depends on the characteristics of the dataset. Apart from the approach to recognize the actions, various proposed methods differ significantly in terms of: the activities which they aim to recognize; camera angle; background properties and image quality.

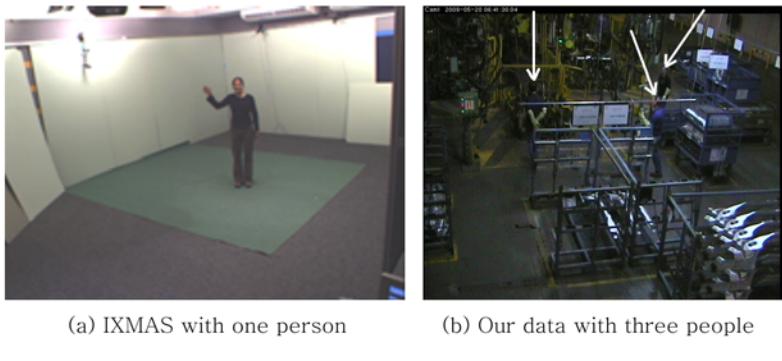
Despite various approaches to human action recognition, the datasets which are used are mainly well-constrained and occlusion-free, which are far from what may be observed by a surveillance camera. The side and frontal views appear to be the dominant view angles for these analyses. In this paper, we will consider the problem of human action recognition from a continuous feed of video capturing from a top view panoramic camera monitoring an industrial plant. In this, the conventional view angles are subject to unworkable levels of occlusion. Multiple subjects may appear on each frame while the background is also changing. We analyse four action categories: walking; carrying; handling and standing still which are taken together from the basic building blocks of a higher level work flow analysis. We use a simple blob tracker to detect the main moving parts i.e. human beings. Various shape-based and motion-based features are then extracted for the action recognition. These features are extracted from a 10 frames long window. A binary decision tree which uses the features selected via the ASFFS feature selection algorithm provides initial prediction for the activity which is being performed. Exploiting our continuous video data, we can then analyse the validity of the predicted sequence of activities and their stability over time. Note that given the nature of the data, which captures a stage in an industrial work flow, there are patterns in the sequences of activities, and these activities are also spatially constrained. The sequence of predicted activities is analysed by HMM models which have been trained on a separate training data.

## 2 Human Activity Analysis

### 2.1 On Viewpoint Selection

There has been very little work in recognition of human activities for the top view. Parameswaran et al. [1] model actions in terms of view-invariant canonical body poses and trajectories in 2D invariance space. On a small dataset they obtained 12/18 true classifications for top view, which is similar to what they achieve for frontal view, while side view obtains a better classification rate. It has been repeatedly mentioned that the top view obtains the lowest classification rates as compared to the other views. The recognition rates of 33.6% [8] and 66.1% [9] have been reported on the IXMAS dataset [8], while the recognition rates from the other views average around 63.9% and 74.1% respectively. These methods are mainly concerned with achieving a viewpoint invariance, which could handle images from the top view as well as the frontal and side views. Lv et al. [10] offer better results for single camera recognition, with a 78.4% recognition rate for top view and an average rate of 81.3% for the other views. In this, they search for the best match to the input sequence among synthetic 2D human pose models for different actions rendered from a wide range of viewpoints. For comparison purposes, note that the IXMAS dataset is a well-constrained dataset with a single moving subject at each frame. Figure 1 shows the front/side view images from IXMAS and our dataset.

Our data is from video cameras monitoring an industrial plant. Note the severely cluttered scene and the level of occlusion for the side/front view



**Fig. 1.** Compare the frontal view from the IXMAS data with a side/front view of our data

camera. In fact, our dataset is characterized by the severe levels of occlusion which affects all the camera views except for the overhead camera (see Figure 2(a)). We propose that this scenario is likely to arise in many surveillance systems specially within similar industrial environments. Thus, for detecting human activities, we have chosen to use the overhead view, which is not affected by occlusion. Unlike the methods mentioned above, we propose to design methods which primarily capture the information from the top view.

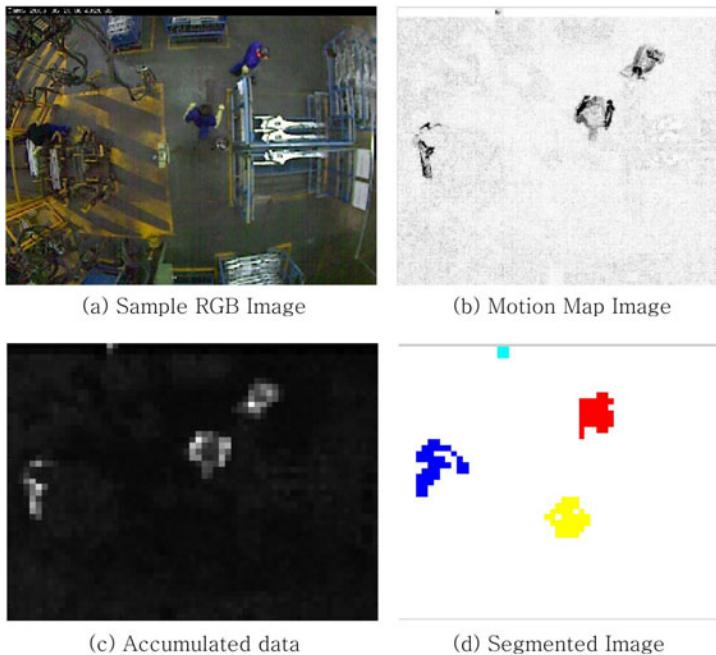
## 2.2 Human Detection and Tracking

In order to derive a set of features for the classification of human behaviour, first we need to determine the number of individual workers and their bounding boxes at each frame. Considering that the humans are the main moving objects in these videos, we apply frame differencing to compute the motion map image based on the change detection for the inter-frame difference. The motion map  $M_t$  at frame  $t$  is computed as the absolute difference of two consecutive frames  $I_t$  and  $I_{t+1}$  as:

$$M_t = ||I_t - I_{t+1}||. \quad (1)$$

An accumulation process is thereafter applied on the motion map by dividing the map into a grid with a bin of size  $10 \times 10$  pixels. Summing the values in each bin, a threshold is then applied to the accumulated image. Finally, Connected Component Analysis is applied to derive the larger blobs which correspond to the human workers. Figure 2 shows the various stages of detection.

In order to track multiple objects across consecutive frames, we propose to model the moving objects as temporal templates characterized by a combination of three basic features: the size, the centroid position, and the aspect ratio of height to width of the bounding box. Shape-based features are considered because they involve low-complexity computation and yet they enjoy robust characteristics. A number of constraints are imposed on these features to handle complex cases of split and merge of moving regions as well as exit and entry into the scene.



**Fig. 2.** Four stages of the human detection

### 2.3 Feature Extraction

A label — walking; carrying; handling and standing — will be assigned to each detected blob at each frame determining its activity. However, to arrive at this label, we consider a period of ten consecutive frames in which the individual is detected. Schindler et al. [11] have also asked the question: how many frames is required for human action recognition? They showed that for the set of actions which they were aiming to recognize a short sequence of 5-7 frames can obtain a performance similar to the analysis of the entire sequence. However, an analysis of recognition from top view has not been considered in this work.

Since both the temporal features and the shape of the moving blob include cues as to the activity which is being performed, we extract both shape-based and motion-based features for the detected blobs. These features are:

- Hu Invariant Moments [12], which are seven moments providing a global description of the shape. These are translation, scale and rotation invariant.
- Region-based properties: area, diameter, etc.
- Motion-based: speed and the direction of speed.

The mean value, within the 10-frame window, for each of these features is considered. However, as well as the mean, the changes in the value of these parameters can provide discriminant cues. Therefore, the sequence of values for each feature

is analysed for the frequency of changes via discrete Fourier transform. Magnitude and phase in different frequencies are then added to the feature vector. Let  $\phi$  be the set of all shape and motion based features which have been listed above. Let  $f_i(n)$  be the feature  $f_i$ , where  $f_i \in \phi$ , detected on the  $n^{th}$  frame of the 10-frame period analysed for each sample.  $F_i$  is the set of features  $f_i$  across the 10 frames interval;

$$F_i = \{f_i(n)\}, n = 1..10. \quad (2)$$

Let  $\mathcal{F}$  denote discrete Fourier transform.

$$X_i = \mathcal{F}(F_i)$$

$$A_i(n) = |X_i(n)|, \varphi_i(n) = \arg(X_i(n)) \quad (3)$$

where  $A_i$  and  $\varphi_i$  denote the magnitude and phase in different frequencies. Thereby the feature vector  $V$  is generated for each sample as:

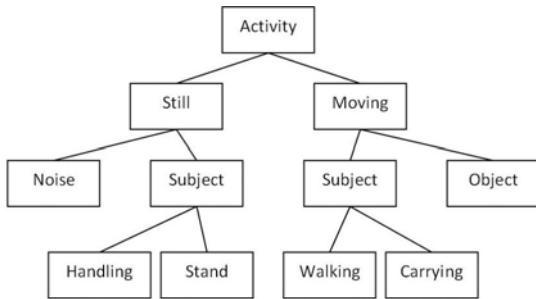
$$V = \{A_i(n), \varphi_i(n), \mu_i, \sigma_i\}, i = 1..|\phi|, n = 1..10 \quad (4)$$

where  $\mu$  and  $\sigma$  denote the mean and the standard deviation of the feature values. Thereby, a large and variant feature vector with 345 features is created.

As discussed in section 2.1 our industrial framework introduces extra complications in terms of limitations in quality and control over the acquired samples. The occlusion in the conventional oblique views have been discussed and a solution was offered through the use of the top view. However, other difficulties include poor image quality, noisy environment, camera shakes, changes in lighting and, in the case of our dataset, a practical issue with random phases of temporal inconsistency resulted from dropped frames. Thereby robustness to noise and outliers appears a desirable feature. Due to the composite nature of our 345-dimensional feature space and that various feature types are susceptible to different levels of corruption in noise, a feature subset selection method is employed to derive the discriminative cues whilst removing the corrupted and irrelevant features. This is explained in more detail in the next section.

## 2.4 Supervised Binary Tree Classification

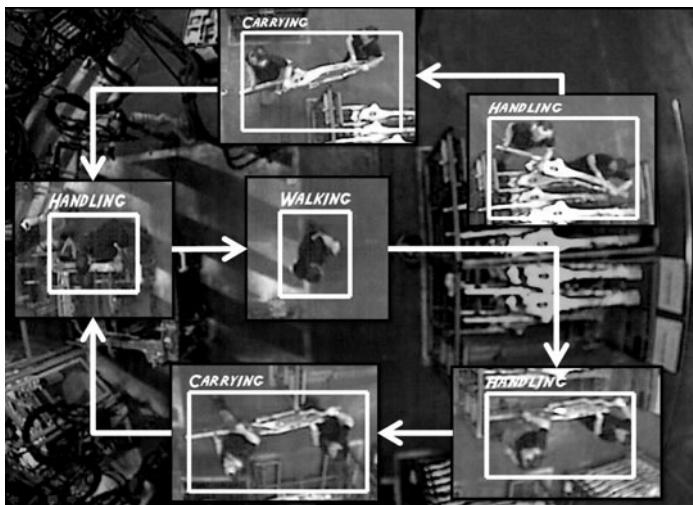
A binary decision tree approach has been adopted for the classification. The taxonomy is being structured for the different types of activities as shown in Figure 3. The output of the tracking stage was manually labelled into four distinct categories: walking; carrying; handling and standing still. Note that the two categories: object and noise (see Figure 3) have not yet been considered and only the detected humans are considered for activity recognition. A feature subset selection is being applied at each node of the tree to derive the best features at the selected node. For this, we use the Adaptive Sequential Forward Floating Selection (ASFFS) [13] algorithm. This is an improved version of the SFFS method which was shown by Jain et al. [14] to outperform the other tested suboptimal methods. Using the gallery of manually labelled activities and the selected subset of features, a k-nearest neighbour is applied at each node to obtain a classification.



**Fig. 3.** The binary tree structure for initial classification of activities

## 2.5 Spatially Specific HMMs for Sequence Analysis

The classification of activities based on visual characteristics and motion features has limitations. For example, carrying might appear as walking if the part being carried is too small. However, there are logical and structural patterns within a sequence of activities, which can be exploited to evaluate the validity of a sequence of predictions. Figure 4 shows some correctly classified activities in individual frames and how they relate to form a work flow within our dataset. The main pattern being displayed here is picking up a part from a rack and placing it on the welding cell. About half of the activities detected fall within this pattern while the rest of activities include walking and standing at arbitrary directions and locations as well as occasional handling of objects.



**Fig. 4.** Human activities detected on individual frames and superimposed on a still image of the plant showing the patterns in the work flow

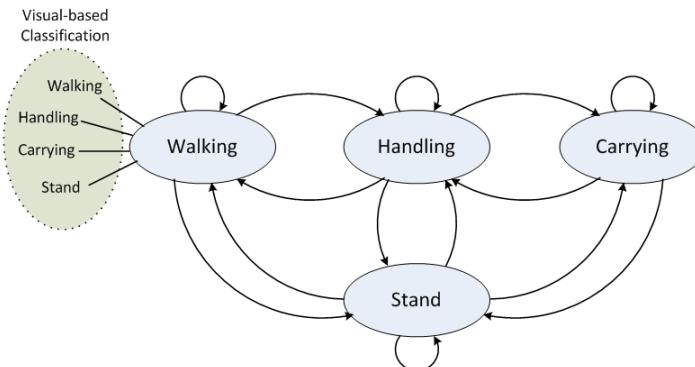
Hidden Markov Models (HMM) can model underlying dependencies within a sequence of unobserved states. As such, they appear an attractive method to analyse the patterns of activities within our data. A HMM with the structure shown in Figure 5 is used to learn the probabilities. In this, the hidden states are the activities — walking; carrying; handling and standing — and the visible states or observations are the predictions obtained by our binary tree classifier. Let the set of predictions,  $A$ , by the decision tree be:

$$A = \{a_t\}, t = 1..T \quad (5)$$

where  $a_t$  is the predicted action at time  $t$ , and  $T$  is the duration of the sequence. Let  $H$  be the set of hidden states for our HMM models. Given the set of predictions, the probability of being in state  $\alpha$  at time  $t$ , denoted by  $S_t = \alpha$ , is recursively calculated by:

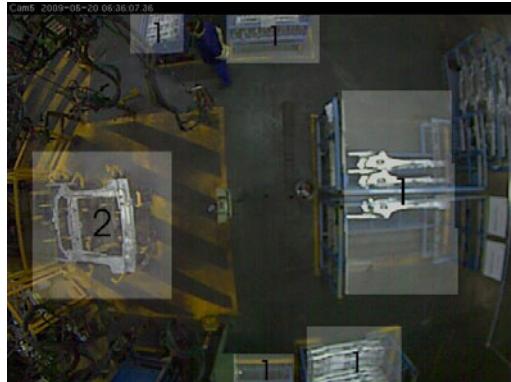
$$\begin{aligned} P(S_t = \alpha | A) = \\ P(A_t = a_t | S_t = \alpha) \cdot \max_{\beta \in H} [P(S_t = \alpha | S_{t-1} = \beta) \cdot P(S_{t-1} = \beta | A)] \end{aligned} \quad (6)$$

In this, the probabilities  $P(A_t = a_t | S_t = \alpha)$  and  $P(S_t = \alpha | S_{t-1} = \beta)$  are given by the HMM model.



**Fig. 5.** HMM structure; the hidden states are the activities and the observations at each state are the initial classifications obtained by the binary tree classification

Our data also imposes that there is a spatial dependency regarding the expectation of various activities. Given a low-level knowledge of the work flows, we have identified three main areas wherein the expectation of occurrence and the sequential order of activities differs significantly: i) the racks (pick up area); ii) the welding cell (put down area); iii) walk ways. Figure 6 highlights these three areas. A hysteresis thresholding improves the stability in determining the area of each sample at each frame. An HMM model has been trained for each of these areas. A separate, manually labelled training set is used for training the HMMs.



**Fig. 6.** The three areas for which different HMM models are generated are highlighted. Area 1 is the racks; area 2 is the welding cell; and the remaining are the walk ways.

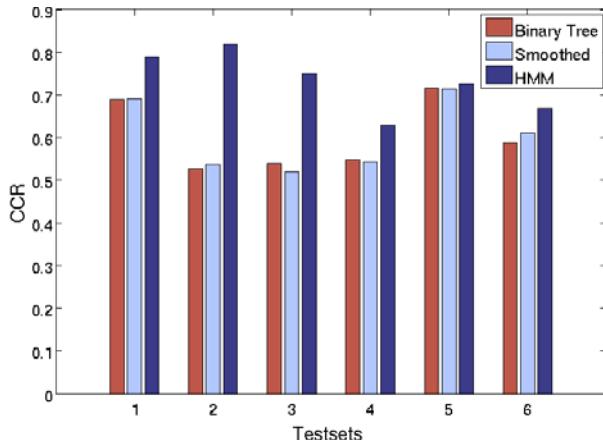
### 3 Experimental Analysis

A total of 170,000 frames have been used in our experiments. The frames are of form shown in Figure 2(a), which is the overhead view of the industrial plant. Multiple moving blobs might be detected at each frame. In average, there are 1,613 samples in each 10,000 frames; a sample being a detected blob in a frame which has been also detected in five frames prior to and in five frames after the current frame. From this, 50,000 frames have been used for feature subset selection. These frames also constitute the gallery to which a sample is compared. 60,000 frames are used in training of the HMM models. The remaining 60,000 frames are used for testing. The output from the tracking is manually labelled into: walking; carrying; handling and standing for all the test and training data.

Figure 7 shows the correct classification rates (CCR) on six separate test sets. Each test set consists of 10,000 consecutive frames. The CCRs for three approaches are shown:

- Binary tree classification: as described in section 2.4
- Binary tree classification with smoothing: In this, each activity which does not persist for more than 5 frames is set to the previous stable activity.
- Binary tree classification with HMM : The sequence of predictions from the binary tree is examined and is set to the most probable underlying sequence using the HMMs.

Clearly, HMM improves the performance in all the test sets. Table 1 gives the details of the recognition performance. Note that these CCRs, which are determined by comparing the auto-classifications to the manual labels at each frame and counting the miss-matches, are the lower-bounds for classification, since there is an ambiguity in labelling the activities in a frame by frame basis. Also, there is an uncertainty in determining when one activity ends and the next one starts. For example, we have manually evaluated the classification labels obtained by the binary tree classifier on testset 4. This manual evaluation shows



**Fig. 7.** The CCRs of activity detection on six testsets, each including 10,000 frames

that the assigned class for each sample is correct in 67% of the times, while the auto-evaluation shows a 55% CCR. A more credible evaluation of performance would be via evaluating the accuracy in detecting the higher level work patterns using these activities. The higher level work flows are deterministic in nature and are easier to label manually. Detecting the work flow patterns is the main avenue for our future research.

**Table 1.** Correct classification rates (CCR) on various testsets

	Testset 1	Testset 2	Testset 3	Testset 4	Testset 5	Testset 6
Binary tree	1617/2345	350/665	694/1286	1316/2403	138/193	1638/2784
	68.96%	52.63%	53.97%	54.76%	71.50%	58.84%
Smoothed	1538/2226	321/597	652/1252	1242/2288	132/185	1625/2658
	69.09%	53.77%	52.08%	54.28%	71.35%	61.14%
HMM	1851/2345	544/665	965/1286	1508/2403	140/193	1863/2784
	78.93%	81.80%	75.04%	62.75%	72.54%	66.92%

## 4 Conclusions

In this paper we have considered the problem of automatically detecting human activities in industrial environments. The top panoramic view have been chosen for the analysis since this view is less likely to be affected by occlusion. At present there is a dearth of analysis of imagery derived from overhead views. This is well suited to industrial environments, and might extend to indoor surveillance scenarios. Shape-based and motion-based features have been used to derive a classification based on a binary-tree structure of activities which are taken from

a higher level work flow. Classifying the activities based on the visual cues has limitations were the activities appear similar. A large improvement is observed when we employ Hidden Markov Models to analyse the sequence of detected activities. Having learned the patterns in activity sequences, these models offer a more viable and stable sequence of predictions based on the initial classification and their spatial properties. Considering the origin of our data which shows a period in a manufacturing cycle, the main avenue for our future research is detecting these higher level work flows.

## References

1. Parameswaran, V., Chellappa, R.: View invariance for human action recognition. *IJCV* 66, 83–101 (2006)
2. Ryoo, M.S., Aggarwal, J.K.: Semantic representation and recognition of continued and recursive human activities. *IJCV* 82, 1–24 (2009)
3. İkizler, N., Forsyth, D.A.: Searching for complex human activities with no visual examples. *IJCV* 80, 337–357 (2008)
4. Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. *TPAMI* 23, 257–267 (2001)
5. Niebles, J.C., Wang, H., Fei-Fei, L.: Unsupervised learning of human action categories using spatial-temporal words. *IJCV* 79, 299–318 (2008)
6. Laptev, I., Caputo, B., Schüldt, C., Lindeberg, T.: Local velocity-adapted motion events for spatio-temporal recognition. *CVIU* 108, 207–229 (2007)
7. Sun, X., Chen, M., Hauptmann, A.: Action recognition via local descriptors and holistic features. In: *CVPR*, Miami, USA (2009)
8. Weinland, D., Boyer, E., Ronfard, R.: Action recognition from arbitrary views using 3d exemplars. In: *ICCV*, Rio de Janeiro, Brazil (2007)
9. Junejo, I.N., Dexter, E., Laptev, I., Pérez, P.: Cross-view action recognition from temporal self-similarities. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV* 2008, Part II. LNCS, vol. 5303, pp. 293–306. Springer, Heidelberg (2008)
10. Lv, F., Nevatia, R.: Single view human action recognition using key pose matching and viterbi path searching. In: *CVPR*, Minneapolis, MN, USA (2007)
11. Schindler, K., van Gool, L.: Action snippets: How many frames does human action recognition require? In: *CVPR*, Anchorage, AK (2008)
12. Hu, M.: Visual pattern recognition by moment invariants. *IEEE Transactions on Information Theory* 8, 179–187 (1962)
13. Somol, P., Pudil, P., Novovičová, J., Paclík, P.: Adaptive floating search methods in feature selection. *Pattern Recognition Letters* 20, 1157–1163 (1999)
14. Jain, A., Zongker, D.: Feature selection: Evaluation, application, and small sample performance. *TPAMI* 19, 153–158 (1997)

# Human Behavior Analysis at a Point of Sale

R. Sicre and H. Nicolas

LaBRI, University of bordeaux - 351 Cours de la libération, 33405 Talence Cedex, France  
Mirane S.A.S - 16 rue du 18 mai 1945 33150 Cenon, France

**Abstract.** This paper presents a method that analyzes human behavior in a shopping setting. Several actions are detected and we are especially interested in detecting interactions between customers and products. This paper first presents our application context, the advantages and constraint of a shopping setting. Then we present and evaluate several methods for human behavior understanding. Human actions are represented with Motion History Image (MHI), Accumulated Motion Image (AMI), Local Motion Context (LMC), and Interaction Context (IC). Then we use Support Vector Machines (SVM) to classify actions. Finally, we combine LMC and IC descriptors in a real-time system that recognizes human behaviors while shopping to enhance digital media impact at the point of sale.

## 1 Introduction

Behavior understanding is a growing field of computer vision. Several applications are developed in order to detect human behaviors in various contexts, such as content-based video analysis and indexing, video-surveillance, interactive applications, etc.

Marketing is a new field of applications that uses computer vision systems to measure media, and display, efficiency. The marketing field has evolved lately. The use of digital media, or digital signage, at point of sale becomes more and more popular. This media offers new forms of communication with customers that bring along new issues. For example, media playing advertising clips one after another does not have significant impact on customers. It is then of primary concern to identify ideal content and location for these media, in order to maximize its impact on customers. Nowadays several software systems help solving these problems. A few systems track customers, in a video-surveillance context, to obtain statistical information regarding customers' habits and displacement inside shopping malls. Various systems calculate directly the media audience and opportunity to see the media, using face detection.

The study introduced in this paper is along the same lines and aims at improving the impact of digital media by maximizing interaction between media and customers. Furthermore, we want to produce statistical data on customers' interaction with products. More specifically, we detect customers picking up products from known areas in real-time using a fixed camera. The detection of such an event results, for example, in playing a clip related to the product.

After a short review, we present the system: first the video analysis part, then the behavior description and recognition. Finally we show some results and conclude.

## 2 Previous Work

Human behavior can be identified in many different contexts [5] [18] [12]. The goal of behavior analysis is to recognize motion samples in order to draw high-level conclusion. There are several issues due to the fact that we match real-world activities to outputs perceived by a video processing module. We have to select relevant properties computed with video processing tasks and handle the incompleteness and uncertainty of these properties.

Behavior analysis is generally composed of two steps: description and recognition of actions. Action description selects relevant measurements that characterize various actions in a specific context. Recognition is usually composed of two processes. First labelled data is generated and used as training. Then these training samples are used to recognize actions using learning methods, such as Hidden Markov Model, Neural Networks [17], Support Vector Machine (SVM) [6] [14], etc. However, recognition can be accomplished using a logical model based on the description measurements, such as Finite State Machine (FSM) [7]. Although this method is not very flexible, it can be applied without the training phase.

In our study, we combine FSM and SVM. FSM are used to detect the simple actions and SVM classify interactions between customers and products.

## 3 Shopping Setting

This section presents the shopping setting with more details. As we see in the previous work, behavior analysis is used in various contexts. Several datasets were used as a baseline for many researchers. We categorize four sorts of datasets used for different applications.

First datasets, like [14] [1] [19], aim at detecting specific motion behavior like people waving, jumping, walking, running, boxing, etc. Videos are mainly taken without camera motion and focus essentially on the actor.

The second type of datasets [9] [16] are directly extracted from movies. These datasets are used to detect people shaking hands, hugging, answering the phone, etc.

Different kind of behavior can be detected in a video-surveillance setting [11] [16], like meetings, language drop, crowd analysis, etc.

The last type of datasets concern sports videos [13]. The configuration varies and can be focused on the actors, extracted from a TV-show, or surveillance like.

Nowadays, only a few papers used dataset coming from point of sale [15] [6]. The shopping setting is between behavior analysis like [14] that observes a person to detect its moving behavior and video-surveillance that detect interactions between people, luggage, specific areas of the scene, etc. Thus we want to detect customer moving behavior as well as interaction with specific areas of the scene, i.e. products areas.

## 4 Behavior Model

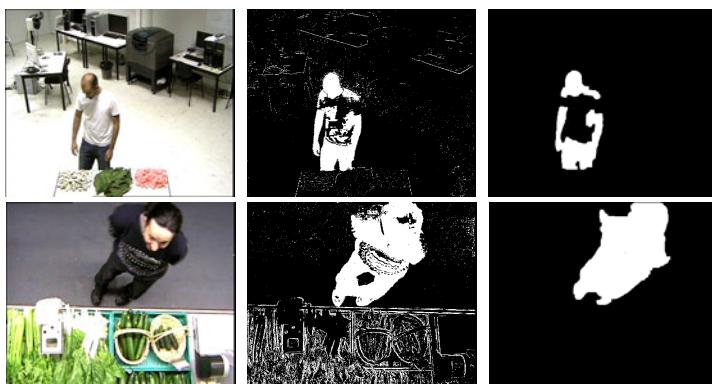
This section presents the model used to define human behavior while shopping. At a point of sale, customers walk around products, look at prices, pick up products, etc.

We create six states that correspond to the current behavior of a person. The chain of states describes the scenario played by the person. The same model is used for higher-level scenario detection and semantic interpretation [15].

- **Enter:** A new person appears in the scene.
- **Exit:** The person leaves the scene.
- **Interested:** The person is close to products, i.e. possibly interested.
- **Interacting:** The person interacts with products, is grabbing products.
- **Stand by:** The person is in the scene but not close to any product area or image boundary. The person can be walking or stopped.
- **Inactive:** The person has left the scene.

## 5 Video Analysis

In order to detect behaviors, we require information concerning every person in the scene. For every frame, we need people's location, contours, etc. Therefore we use a motion detection and object tracking process. Motion detection finds moving regions that do not belong to the background. Then, these regions are tracked over the frame sequence. Fast methods were selected in order to cope with the real-time constraints of our final application. [20] presents a clear overview of the object tracking methods.



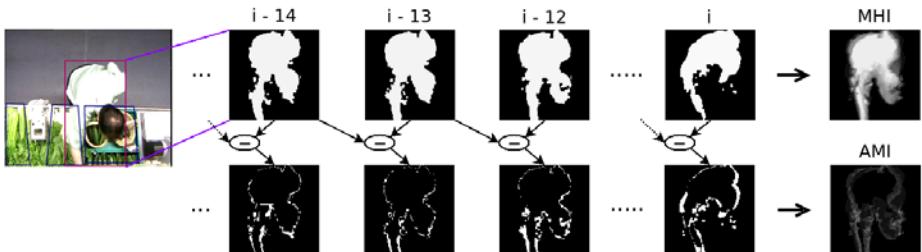
**Fig. 1.** Motion detection on videos from LAB1 (top) and MALL1 (bottom) datasets. Frames are on the first columns, rough detection on the second and filtered results on the third.

**Motion detection** uses a pixel based model of the background to generate precise contour of the detected regions. A mixture of Gaussians is associated with each pixel, in order to characterize the background [21]. This model is updated on-line. A Gaussian distribution is matched to the current value of each pixel. If this Gaussian belongs to the background, the pixel is classified as such. Otherwise the pixel is considered as foreground. Morphological filters are finally applied on this result, see figure 1.

**Object Tracking** is composed of two main processes. First, we calculate, for each region, a descriptor based on its position, size, surface area, first and second order color moments. Then, these descriptors are matched from one frame to the next, using a voting process. Secondly, we use matched regions to build and update the object list. An object is a region that was tracked for several frames. Matched regions are used to update information about objects: location, size, etc. Then unmatched regions are compared to inactive unmatched object to solve miss-detections. We also detect regions split and merge to detect occlusions.

## 6 Behavior Description

We focus on tracked person's interactions with product areas, i.e. people grabbing products. While grabbing a product, a person first reaches out with its arm, then grasps a product, and finally take the product. These different phases in the "product grabbing" event correspond to observable local motion of the person. Following the idea that similarity between various motions can be identified through spatio-temporal motion description, a corresponding descriptor has to be composed of sets of features sampled in space and time [17] [4]. This section presents various description methods.



**Fig. 2.** Diagram showing how MHI and AMI are generate from the frame sequence

### 6.1 Motion History Image

MHI is a temporal template used as model for actions [2]. MHI offers information concerning a person shape and the way it varies along a local period of time. We aggregate a sequence of foreground object masks, scaled to a standard size of 120x120 pixels, see figure 2. MHI is computed as follows:

$$MHI(x, y) = \frac{1}{T} \sum_{t=1}^T I(x, y, t) \quad (1)$$

Where  $I(x, y, t)$  is the pixel value of the Image  $I$  at position  $(x, y)$  at time  $t$ .  $T$  is the time interval used to calculate the MHI, we choose  $T = 15$ .

We define two energy histograms by projecting MHI values along horizontal and vertical axis [8]. These energy histograms are calculated as follows:

$$\begin{aligned} EH_h(i) &= \sum_{j=0}^{W-1} MHI(i, j), i = 0, \dots, H - 1 \\ EH_v(j) &= \sum_{i=0}^{H-1} MHI(i, j), j = 0, \dots, W - 1 \end{aligned} \quad (2)$$

$H$  and  $W$  are relatively the height and width of our scaled image. We have  $H = W = 120$ . These two energy histograms are used as a 240 (120x2) dimensional descriptor to recognize Interactions.

## 6.2 Accumulated Motion Image

AMI [8] was inspired from MHI and Motion Energy Image (MEI) [2]. As we see in the previous section, MHI and MEI use the entire silhouette. However, only areas including changes are used to generate the AMI that is defined as follows:

$$AMI(x, y) = \frac{1}{T} \sum_{t=1}^T |D(x, y, t)| \quad (3)$$

Where  $D(x, y, t) = I(x, y, t) - I(x, y, t-1)$ . We note that the image difference is calculated between two scaled masks and we keep  $T = 15$ , see figure 2.

We calculate the same energy histograms presented in the previous section that are used as descriptor (240 dimensions) to recognize interactions.

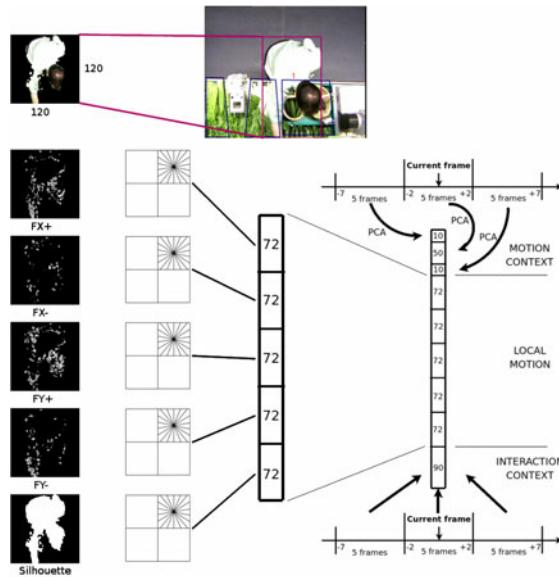
## 6.3 Local Motion Context

We then choose to describe motion using pixel-wise optical flow [4]. Since optical flow is not very accurate, we use histograms of features over image regions. Such a representation is tolerant to some level of noise, according to [17].

**Local motion:** First, each person's mask is scaled to a standard size of 120x120 pixels, while keeping aspect ratio. Then, the optical flow is computed using Lucas Kanade algorithm [10]. The result of this process is two matrixes with values of motion vectors along x and y axis. We separate negative from positive values in the two matrixes, and end out with 4 matrixes before applying a Gaussian blur to reduce the effects of noises.

**Silhouette:** A fifth matrix, representing the person's silhouette, is computed from the scaled mask.

**Data quantization:** We reduce the dimensionality of these matrixes to filter noises and save computation time. Each matrix is divided into a 2x2 grid. Each grid cell gets its values integrated over an 18-bin radial histogram (20 degrees per bin). Matrixes are now represented by a 72 (2x2x18) dimensional vector.



**Fig. 3.** Diagram representing the local motion and behavior context descriptor same summed

**Temporal context:** to take into account temporal information, we use 15 frames around the current frame and split them in three sets of 5 frames: past, current, and future. After applying Principal Component Analysis (PCA) on each set's descriptors, we keep the first 50 components for the current set, while we only keep the first 10 components for the past and future sets. The temporal context descriptor possesses then 70 (10+50+10) dimensions.

The final descriptor is composed of 430 (72x5+70) dimensions, see figure 3.

#### 6.4 Interaction Context

This last descriptor is based on interaction with product areas. These areas are assumed to be known. We use six measurements calculated as follow:

- The person's surface covering a product area.
- A Boolean that is true when this covering surface is bigger than a theoretical hand size or when a person is connected to a product area and there is motion detected on this area.
- The surface of the person.
- The height of its bounding box.
- The position of the bottom of the bounding box along y axis.
- The position of the top of the bounding box along y axis.

The first measurement increases when a customer is reaching out before taking a product. The second measurement detects motion in products area. In fact, when a product is taken motion is detected where the product is missing. Furthermore, the measurements, related to the height and position of the bounding box, have meaningful

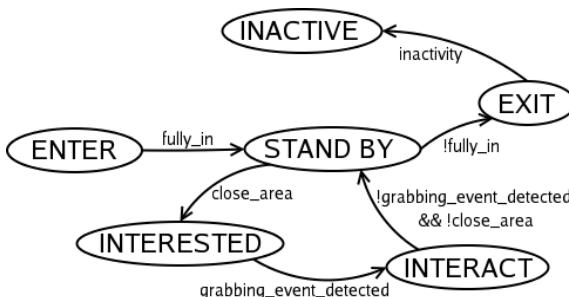
variations as a person reaches out for products. The surface tends to increase as a person grasps a product, when products are big enough. These measurements fill the interaction context descriptor that possess 90 (6x15) dimensions, because we keep each measurement of the 15 last frames.

After running some tests, see results section 8, we decide to combine the local motion context and the interaction context description into one descriptor of 520 (430+90) dimensions, see figure 3.

## 7 Interaction Recognition

This section presents the behavior recognition process. Based on the behavior model, we detect the six states and build a Finite State Machine (FSM). Using video analysis and behavior description information, for each frame, the state of each object has to be identified among the six pre-defined states:

- **Enter** is detected when a person appears and is connected to an image boundary.
- **Exit** is detected when a previously tracked person is connected to an image boundary.
- **Interested** is detected when a person's contour connects a product area.
- **Stand by** is detected when a person is in the scene and not connected to a product area or an image boundary.
- **Inactive** is detected when the system loses track of a person. This event happens when a person has left the scene or is occluded by something in the scene, or another person.
- **Interaction** is detected using SVM [3], with a radial basis function kernel, on various descriptors presented above.



**Fig. 4.** Part of the Finite State Machine. All transitions are not written to make it clear.

**Finite State Machine** is used in order to organize and prioritize the six states [7]. The state machine is synchronous and deterministic. Synchronous means that the machine iterates over each new frame. Based on the previous state, the system calculates a new one by testing each transition condition. If a condition is satisfied, the system moves to the new state. Otherwise, the system stays in the same state. The machine is deterministic because for each state, there can not be more than one transition for each possible input. One FSM model the behavior of one person, see figure 4.

## 8 Results

**Datasets description:** We use different datasets taken with the same camera, with 15 frames per second. A part of the datasets was taken in our laboratory (LAB1 and LAB3). The others were taken in a real shopping mall (MALL1 and MALL2). The two first datasets (LAB1 and MALL1) possess five and six sequences respectively and contains a lot of interactions with products. Two and four different people are shopping respectively, see figure 5. Products taken by people have different shapes, colors, and sizes in the video sequences. Furthermore, all products are identical in the heaps. LAB3 and MALL2 are two datasets where multiple people interact together. Two to four people interact simultaneously in the scene, see figure 5.

**Table 1.** Recall-Precision table for the Interact state for various descriptors on two datasets

Dataset	Video	Frames	MHI R	MHI P	AMI R	AMI P	LMC R	LMC P	IC R	IC P	MI R	MI P
<b>MALL1</b>	1	327	0.4787	0.3261	0.4894	0.4646	0.5102	0.3937	0.5306	0.5977	0.8163	0.6667
	2	444	0.2178	0.2716	0.1386	0.5833	0.5149	0.4815	0.9307	0.9592	0.9505	1
	3	434	0.4919	0.7625	0.5403	0.7128	0.6822	0.869	0.6667	0.7368	0.7525	0.5802
	4	336	0.6386	0.5699	0.5783	0.6857	0.1148	0.1591	0.6905	0.9063	0.7976	0.8701
	5	164	0	0	0.125	0.0833	1	0.3333	0.5	1	0.5	1
	6	232	0.7797	0.7797	0.5085	0.4688	0.8852	0.6429	0.8475	0.9434	0.8983	0.9815
	<i>mean</i>		<b>0.4345</b>	<b>0.4516</b>	<b>0.3967</b>	<b>0.4998</b>	<b>0.6179</b>	<b>0.4799</b>	<b>0.6943</b>	<b>0.8572</b>	<b>0.7859</b>	<b>0.8498</b>
<b>LAB1</b>	1	545	0.1898	0.3514	0.6423	0.5946	0.0092	1	0.7299	0.7692	0.8321	0.8085
	2	672	0.1441	0.2133	0.036	0.1739	0.2697	0.3333	0.6585	0.648	0.6748	0.6288
	3	704	0.2581	0.48	0.4247	0.4031	0.5185	0.332	0.6774	0.9333	0.7473	0.9392
	4	771	0.0854	0.4242	0.2561	0.2979	0.153	0.5185	0.7203	0.7687	0.7552	0.7347
	5	518	0.2364	0.1711	0.3091	0.4146	0.9153	0.7013	0.9818	0.75	0.9818	0.7941
	<i>mean</i>		<b>0.1828</b>	<b>0.328</b>	<b>0.3336</b>	<b>0.3768</b>	<b>0.3731</b>	<b>0.577</b>	<b>0.7536</b>	<b>0.7738</b>	<b>0.7982</b>	<b>0.7811</b>

**Table 2.** Recall-Precision Table for the Interact state using two descriptors on complex datasets

Dataset	Video	Frames	Recall IC	Precision IC	Recall MI	Precision MI
<b>MALL2 MP</b>	1	215	0.8929	0.8333	0.8214	0.902
	2	208	0.6462	0.8235	0.7846	0.8947
	3	735	0.7151	0.7278	0.9101	0.72
	4	153	1	0.5106	0.5417	0.9286
	5	382	0.8333	0.8404	0.6583	0.8404
	6	504	0.7273	0.8571	0.8561	0.8828
	<i>mean</i>		<b>0.8025</b>	<b>0.7655</b>	<b>0.762</b>	<b>0.8614</b>
<b>LAB3 MP</b>	1	212	0.7282	0.8929	0.8738	0.9375
	2	211	0.9063	0.8969	0.7604	0.9012
	3	300	0.784	0.7538	0.856	0.7643
	4	303	0.7561	0.5636	0.7317	0.6383
	5	259	0.8158	0.6596	0.8026	0.6854
	<i>mean</i>		<b>0.7981</b>	<b>0.7534</b>	<b>0.8049</b>	<b>0.7853</b>

**Tests on datasets:** In order to recognize product grabbing events, we use a cross validation process, see table 1 and 2. In other words, to recognize events on a video, we use all the other sequences of the dataset as training and then calculate recall and precision. It is interesting to note that using this process, we only use a few minutes of video as training with a few actors.



**Fig. 5.** Screenshots from LAB3 datasets on the first columns and MALL1 and MALL2 datasets on the right

After testing the four descriptors on various datasets, see table 1, we noticed that the appearance is not necessarily preserved from one sequence to another. Furthermore some videos show customers with shopping cart or basket that are detected as foreground. These detections modify completely the appearance of the persons.

Then we decided to combine the two descriptors offering the best results to test more datasets, see table 1. We compare result using only the Interaction context (IC) descriptor and combined local motion and interaction context (MI) descriptor. The MI performs as good as IC for precision, but offers better results for recall on the first datasets. On multiple people datasets, MI performs slightly better than IC on average. However, local motion description tends to be noisier, due to occlusions and a few object tracking mismatches. IC remains robust in these situations and the recognition rates in multi-person datasets are as good as in the first datasets, see table 2.

MALL performs better than LAB on recall and precision for the Interact state due to the position of the camera, located directly above the products and closer on MALL than on LAB, see table 2 and figure 5. We understand that the camera location is really important to maximize the accuracy of the system. A position close to the products performs better on Interact state recognition. However, having the camera too close to the products make us lose information about the customers, since they are only detected when they are near the products.

**Computation time:** The application has to generate responses quickly as soon as a specific event is detected. The program is tested on a Pentium M, 1.73 Ghz with 500Mb of RAM. The application can analyze 6 to 10 frames per second for an image resolution of 704x576 or 640x480 respectively. Motion detection is the most computational expensive process.

## 9 Conclusion

This paper presents a novel type of application, using computer vision in the field of marketing, to improve interaction between customers and digital media. We evaluate various methods for behavior analysis and Interact context description outperforms

the other methods. By combining Interact context and local motion context description, we improve these results. Interactions with products are well detected with a precision of 0.79 and a recall of 0.85.

As future work, the method can be generalized and tested on various complex scenes: products on vertical racks, complex products like clothes. It would also be interesting to look for other behaviors and scenarios that can be characterized and detected using this technique.

## References

- Blank, M., Gorelick, L., Shechtman, E., Irani, M., Basri, R.: Actions as space-time shapes. *ICCV* 2, 1395–1402 (2005)
- Bobick, A.F., Davis, J.W.: The recognition of human movement using temporal templates. *IEEE Trans. on Pattern Analysis and Machine Intel.* 23, 257–267 (2001)
- Chang, C., Lin, C.: LIBSVM: a library for support vector machines (2001), <http://www.csie.ntu.edu.tw/~cjlin/libsvm>
- Efros, A., Berg, A., Mori, G., Malik, J.: Recognizing action at a distance. In: *ICCV* (2003)
- Hu, W., Tan, T., Wang, L., Maybank, S.: A survey on visual surveillance of object motion and behaviours. *IEEE Transac. on Syst., Man, and Cyb.*, 334–352 (2004)
- Hu, Y., Cao, L., Lv, F., Yan, S., Gong, Y., Huang, T.S.: Action detection in complex scenes with spatial and temporal ambiguities. In: *ICCV* (2009)
- Ikizler, N., Forsyth, D.: Searching video for complex activities with finite state models. In: *CVPR* (2007)
- Kim, W., Lee, J., Kim, M., Oh, D., Kim, C.: Human action recognition using ordinal measure of accumulated motion. In: *EURASIP JASP* (2010)
- Laptev, I., Marszalek, M., Schmid, C., Rozenfeld, B.: Learning realistic human actions from movies. In: *CVPR* (2008)
- Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: *7th IJCAI*, pp. 674–679 (1981)
- PETS: Performance Evaluation of Tracking and Surveillance, <http://winterpets09.net/>
- Poppe, R.: A survey on vision-based human action recognition. *Im. & Vis. Comp.* J. 28, 976–990 (2010)
- Rodriguez, M.D., Ahmed, J., Shah, M.: Action MACH: a spatio-temporal maximum average correlation height filter for action recognition. In: *CVPR* (2008)
- Schüldt, C., Laptev, I., Caputo, B.: Recognizing human actions: a local SVM approach. In: *ICPR*, vol. 3, pp. 32–36 (2004)
- Sicre, R., Nicolas, H.: Shopping scenarios semantic analysis in videos. In: *CBMI* (2010)
- Smeaton, A., Over, P., Kraaij, W.: Evaluation campaigns and TRECVID. In: *ACM MIR* (2006)
- Tran, D., Sorokin, A.: Human activity recognition with metric learning. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) *ECCV 2008, Part I*. LNCS, vol. 5302, pp. 548–561. Springer, Heidelberg (2008)
- Turaga, P., Chellappa, R.: Machine recognition of human activities: a survey. *IEEE Trans. on Circ. and Syst. for Video Tech.* 18(11), 1473–1488 (2008)
- Weinland, D., Ronfard, R., Boyer, E.: Free view-point action recognition using motion history volumes. *CVIU* (104), 249–257 (2006)
- Yilmaz, A., Javed, O., Shah, M.: Object tracking: a survey. *ACM Comput. Surveys* (2006)
- Zivkovic, Z., van der Heijden, F.: Efficient adaptive density estimation per image pixel for the task of background subtraction. *Pattern Recognition Letters* 27(7) (2006)

# Author Index

- Abdelrahman, Mostafa III-9  
Alba, Alfonso I-35, I-417  
Alger, Jeffrey R. II-292  
Al-Hamadi, Ayoub I-153, I-253, II-574  
Al-Huseiny, Muayed S. II-252  
Ali, Asem III-79  
Alvarez, Damián III-171  
Ambrosch, Kristian I-437  
Amresh, Ashish I-54  
Antani, Sameer K. III-261  
Arandjelović, Ognjen III-89  
Arbab-Zavar, Banafshe III-625  
Arceneaux IV, George II-564  
Arce-Santana, Edgar I-35, I-417  
Argyros, A.A. II-584  
Argyros, Antonis A. I-405  
Artinger, Eva II-429  
Asari, Vijayan K. III-49, III-474  
Assoum, A. I-276  
Atsumi, Masayasu II-696  
Azary, Sherif II-606
- Badakhshannoory, Hossein II-342  
Bai, Li I-371, II-637  
Balázs, Péter III-339  
Bales, M. Ryan I-211  
Barman, Sarah II-669  
Bärz, Jakob I-582  
Basile, Teresa M.A. I-571  
Baumann, Florian I-286  
Bebis, George III-161  
Beichel, Reinhard II-312  
Belhadj, Farès III-524  
Berger, Marie-Odile I-231  
Beyer, Ross I-688  
Bhooshan, Sunil III-458  
Bi, Chongke I-328  
Bloch, Isabelle I-393  
Boggess, Erin I-350  
Boggus, Matt II-213, II-501  
Borghi, A. II-678  
Borst, Christoph W. I-719, I-729  
Bouchrika, Imed III-625  
Boyer, Vincent III-504, III-524
- Branch, John William I-602  
Bresson, Xavier II-97  
Brillet, Pierre-Yves II-302  
Brimkov, Valentin E. I-592  
Bringay, Sandra III-534  
Bukhari, Faisal II-11  
Burch, Michael I-338, III-447  
Buthupitiya, Senaka III-1
- Caldairou, Benoît I-13  
Camargo, Aldo I-698  
Campos-Delgado, Daniel U. I-35  
Cao, Yanpeng I-654  
Caponetti, Laura I-571  
Caracciolo, Valentina II-491  
Carter, John N. III-625  
Casanova, Manuel F. III-9  
Caunce, Angela I-132  
Cavalcanti, Pablo G. I-190  
Cernuschi-Frías, Bruno III-271  
Cerón, Alexander III-349  
Chalfoun, Joe I-23, I-549  
Chambon, Sylvie II-182  
Chang, Remco II-564  
Chan, T.F. II-678  
Cheikh, Faouzi Alaya II-491  
Chelberg, David III-417  
Chen, Bertha III-604  
Chen, Dongqing III-9  
Chen, Huiyan III-368, III-407  
Chen, Qian I-449, III-229  
Chen, Runen II-687  
Chen, Wei I-427  
Chen, Xi I-612  
Cheng, Heng-Tze III-1  
Cheng, Irene II-406  
Chesi, G. III-109  
Chiang, Pei-Ying I-108  
Cho, Siu-Yeung II-129  
Choe, Yoonsuck II-322  
Choi, Inho III-199  
Chung, Ronald III-280  
Coffey, Dane II-351  
Constantinou, Christos E. III-604

- Cootes, Tim I-132  
 Cope, James S. II-669  
 Cordes, Kai I-264  
 Corsi, Christopher I-74  
 Cottrell, Garrison W. I-199  
 Couture-Veschambre, Ch. II-416  
 Crawfis, Roger II-213, II-501, II-511  
 Crawford, P. II-533  
 Cretu, Ana-Maria II-232  
 Crivelli, Tomás III-271  
 Crouzil, Alain II-182  
 Cummings, Alastair H. II-332
- Dailey, Matthew N. II-11  
 Daněk, Ondřej III-387  
 Darbon, J. II-678  
 Das, Dipankar II-439  
 Das, Kaushik I-719  
 Davis, James W. I-120, I-381, III-613  
 de Santos Sierra, Alberto I-479  
 Denzler, Joachim II-459  
 Desquesnes, Xavier II-647  
 Dhome, Michel III-219  
 Dickerson, Julie A. I-350  
 Diem, Markus III-29  
 Dillard, Scott E. II-64  
 Dima, Alden A. I-23, I-549, II-736  
 Do, Phuong T. III-484  
 Dodds, Z. III-151  
 D'Orazio, Tiziana III-291  
 Dornaika, F. I-276  
 Du, Shengzhi III-320  
 Dubois, Eric III-189  
 Dubuisson, Séverine I-393  
 Duschl, Markus II-429
- Ehlers, Arne I-286  
 Eichmann, David III-139  
 Eikel, Benjamin I-622  
 Elhabian, Shireen III-9, III-79  
 Elias, Rimon II-161  
 Elliott, John T. I-23, I-549  
 Elmoataz, Abderrahim I-539, II-647  
 English, C. II-53  
 Ernst, Katharina I-286  
 Esposito, Floriana I-571
- Fabian, Tomas II-716  
 Fabián, Tomáš III-310  
 Fairhurst, M.C. I-461
- Falk, Robert III-79  
 Fanello, Sean R.F. II-616  
 Farag, Ahmed III-9  
 Farag, Aly III-9, III-79  
 Farag, Amal III-79  
 Fechter, Todd A. II-394  
 Feijóo, Raúl I-529  
 Fellner, Dieter W. III-514  
 Feltell, David I-371  
 Fetita, Catalin II-302  
 Fierrez, J. I-461, I-489  
 Fijany, Amir II-469  
 Filliben, James J. I-23  
 Finnegan, David I-666  
 Fiorio, Christophe II-85  
 Fischer, Matthias I-622  
 Flores, Arturo I-199  
 Fong, Terry I-688  
 Fontaine, Jean-Guy II-469  
 Forsthöfel, Dana I-211  
 Förstner, Wolfgang I-654  
 Fowers, Spencer G. III-368, III-407  
 Foytik, Jacob III-49  
 Fujishiro, Issei I-328  
 Fujiwara, Takanori I-306  
 Fünfzig, Christoph I-54
- Gales, Guillaume II-182  
 Gao, Zhiyun III-129  
 García-Casarrubios Muñoz, Ángel I-479  
 García, Hernán III-171  
 Garz, Angelika III-29  
 Gaura, Jan III-310  
 Geiger, Cathleen I-666  
 Geismann, Philip I-243  
 Geist, Robert I-74  
 Giménez, Alfredo II-554  
 Gomez, Steven R. II-373  
 Gong, Jianwei III-407  
 Gong, Minglun II-481  
 Gori, Ilaria II-616  
 Graham, James III-9, III-79  
 Grammenos, D. II-584  
 Grand-brochier, Manuel III-219  
 Grazzini, Jacopo II-64  
 Grenier, Philippe II-302  
 Grout, Randall III-129  
 Grover, Shane II-361  
 Gschwandtner, Michael III-19

- Gu, Yi III-437  
 Gueorguieva, S. II-416  
 Guerra Casanova, Javier I-479  
 Guo, Yu I-96  
 Gupta, Raj Kumar II-129  
 Gutierrez, Marco I-529
- Hamann, Bernd II-554  
 Hanbury, Allan II-75  
 Hansen, Tina I-582  
 Hantos, Norbert III-339  
 Hao, Qing II-292, III-359  
 Hardeberg, Jon Y. I-361  
 Hatsuda, Hiroshi III-594  
 He, Qiang I-698  
 He, Zifen III-377  
 Hempe, Nico II-202  
 Hishida, Hiroyuki III-39  
 Hlawitschka, Mario II-554  
 Hödlmoser, Michael II-1  
 Hoeber, Orland II-481  
 Hoffman, Eric III-129  
 Hoffmann, Kenneth R. III-359  
 Hoi, Yiemeng III-359  
 Holtze, Colin III-129  
 Hoque, Enamul II-481  
 Horiuchi, Takahiko I-181  
 Hosseini, Fouzhan II-469  
 House, Donald II-192  
 Hu, Xiao II-292, III-359  
 Huang, Rui II-139  
 Hung, Y.S. II-21, III-109  
 Hussain, Muhammad I-64
- Ibrahim, Mina I.S. I-499  
 Ignakov, D. II-53  
 Ikeda, Osamu I-678  
 Imiya, Atsushi I-561  
 Iwamaru, Masaki I-306
- Jähn, Claudius I-622  
 Jeon, Ju-II II-659  
 Jia, Ming I-350  
 Jiang, Caigui I-96  
 Jianu, Radu II-373, III-494  
 Johnson, Gregory II-222  
 Ju, Myung-Ho II-273  
 Jung, Keechul II-726
- Kambhamettu, Chandra I-519, I-666, II-170  
 Kampel, Martin I-163, II-1  
 Kanan, Christopher I-199  
 Kang, Hang-Bong II-273  
 Kang, Hyun-Soo II-659, III-239  
 Kao, Chiu-Yen II-117  
 Karpenko, Simon I-173  
 Kashu, Koji I-561  
 Kawai, Takamitsu I-634  
 Keefe, Daniel F. II-351, II-564  
 Kerren, Andreas I-316  
 Keshner, Emily II-222  
 Khalili, Ali II-469  
 Khan, Rehanullah II-75  
 Kim, Daijin III-199  
 Kim, Jibum III-119  
 Kim, Myoung-Hee I-45, III-209  
 Kim, Taemin I-688, II-283  
 Klinker, Gudrun II-429  
 Knoll, Alois I-243  
 Kobayashi, Yoshinori II-439  
 Kohlmeyer, Axel II-382  
 Korbel, M. III-151  
 Korcheck, Dennis P. III-484  
 Korsakov, Fedor II-351  
 Kotera, Hiroaki I-221  
 Koutlemanis, P. II-584  
 Kovács, Levente III-59  
 Kozubek, Michal III-387  
 Krumnikl, Michal III-310, III-465  
 Kuhl, Ellen III-604  
 Kuno, Yoshinori II-439  
 Kuo, C.-C. Jay I-108  
 Kuo, May-Chen I-108  
 Kwon, Yunmi I-86
- Laidlaw, David H. II-373, III-494  
 Lavee, Gal II-706  
 Leach, Andrew I-592  
 Leece, M. III-151  
 Lee, Dah-Jye III-407  
 Lei, K. III-151  
 Leo, Marco III-291  
 Lesperance, N. III-151  
 Lézoray, Olivier I-539, II-647  
 Li, Baoxin II-449, III-249  
 Li, Bo II-151  
 Li, Chunming II-117  
 Li, Ling I-350

- Li, Xin III-368  
 Liebeskind, David S. II-292, III-359  
 Lima, Jesus Romero III-574  
 Ling, Haibin II-222  
 Ling, Haibin I-296  
 Lipari, Nicholas G. I-729  
 Liu, Jundong III-417  
 Liu, Wei II-242, II-262  
 Liu, Yonghuai I-644  
 Loménie, Nicolas I-1  
 Lopes, Carlos B.O. I-190
- Ma, Yingdong I-449, III-229  
 Maeder, Anthony II-545  
 Mahalingam, Gayathri I-519  
 Mahmoodi, Sasan I-499, II-252  
 Mailing, Agustin III-271  
 Mandal, Mrinal II-406  
 Mannan, Md. Abdul II-439  
 Mansouri, Alamin I-361  
 Marín, Mirna Molina III-574  
 Mark, L.H. II-53  
 Martin, Ralph R. I-644  
 Martin, Rhys III-89  
 Mas, Andre II-85  
 Mason, J.S.D. I-489  
 Mastroianni, Michael I-592  
 Matsumoto, S. III-151  
 Matsunaga, Takefumi I-751  
 Matsushita, Ryo I-306  
 Matula, Pavel III-387  
 Maška, Martin III-387  
 Mayol-Cuevas, Walterio W. II-596  
 Mazzeo, Pier Luigi III-291  
 McDonald, John I-654  
 McGraw, Tim I-634  
 McInerney, T. II-533  
 McLaughlin, Tim II-394  
 McLean, Linda III-604  
 Meyering, Wietske I-529  
 Meziat, Carole I-1  
 Michaelis, Bernd I-153, I-253, II-574  
 Michikawa, Takashi III-39  
 Min, Kyungha I-86  
 Moan, Steven Le I-361  
 Moratto, Zachary I-688, II-283  
 Moreland, John R. III-484  
 Morimoto, Yuki I-707  
 Morita, Satoru III-554, III-584
- Mourning, Chad III-417  
 Müller, Christoph III-447  
 Müller, Oliver I-264  
 Müller, Stefan I-582
- Naegel, Benoît I-13  
 Nazemi, Kawa III-514  
 Nedrich, Matthew I-120  
 Nefian, Ara III-181  
 Nefian, Ara V. I-688, II-283, III-1  
 Neo, H.F. III-427  
 Nguyen, Hieu V. II-637  
 Nguyen, Quang Vinh II-545  
 Nicolas, H. III-635  
 Nicolescu, Mircea III-161  
 Nikolaev, Dmitry I-173  
 Nixon, Mark S. I-499, II-252, II-332, III-625  
 Noury, Nicolas I-231  
 Nykl, Scott III-417
- Ohtake, Yutaka III-39  
 Okamoto, Koji I-306  
 Okouneva, G. II-53  
 O'Leary, Patrick II-361  
 Ono, Kenji I-707  
 Oota, Satoshi III-39  
 Orozco, Álvaro III-171  
 Ortega-Garcia, J. I-461, I-489  
 Ortner, Margarete II-302  
 Osher, S. II-678  
 Osher, Stanley II-97, II-117  
 Oshita, Masaki I-751  
 Ostermann, Jörn I-264  
 Othmani, Ahlem I-1
- Pang, Xufang I-612  
 Papoutsakis, Konstantinos E. I-405  
 Park, Anjin II-726  
 Park, Dong-Jun III-139  
 Park, Jae-Hyeung III-239  
 Parrigan, Kyle I-143  
 Pasing, Anton Markus II-394  
 Passat, Nicolas I-13  
 Pathan, Saira Saleem I-153  
 Payeur, Pierre II-232  
 Pecheur, Nicolas III-534  
 Pedersen, Marius II-491  
 Pelfrey, Brandon II-192  
 Peña, B. Adán II-394  
 Peng, Kun II-151

- Pérez, Eduardo Islas III-574  
 Peskin, Adele P. I-23, I-549, II-736  
 Petpon, Amnart III-69  
 Petriu, Emil M. II-232  
 Peyronnet, S. II-678  
 Pirri, Fiora II-616  
 Prasad, Lakshman II-64  
 Pree, Wolfgang III-19  
 Prêteux, Françoise II-302  
 Prieto, Flavio III-349  
 Ptucha, Raymond III-301  
 Purgathofer, Werner II-41  
 Puxbaum, Philipp I-437
- Rada, Jessica Bahena III-574  
 Rahman, Md Mahmudur III-261  
 Raschke, Michael I-338  
 Rashid, Omer I-253  
 Rastgar, Houman III-189  
 Rebelo, Marina I-529  
 Reichinger, Andreas II-41  
 Reina, Guido III-447  
 Reisner-Kollmann, Irene II-41  
 Reitz, Judith II-394  
 Remagnino, Paolo II-669  
 Ribeiro, Eraldo II-242, II-262  
 Rivera, Mariano I-417  
 Rivlin, Ehud II-706  
 Roche, Mathieu III-534  
 Rohith, M.V. I-666, II-170  
 Rohrschneider, Markus I-316  
 Rosenbaum, René II-554, III-99  
 Rosenhahn, Bodo I-264, I-286  
 Rosin, Paul L. I-644  
 Rossmann, Jürgen II-202  
 Rossol, Nathaniel II-406  
 Roullier, Vincent I-539  
 Rudzsky, Michael II-706  
 Rundensteiner, Elke A. II-522
- Sánchez Ávila, Carmen I-479  
 Sablatník, Robert III-29  
 Sadeghi, Mohammad T. III-329  
 Sadek, Samy II-574  
 Saeedi, Parvaneh II-342  
 Safari, Saeed II-469  
 Saha, Punam K. III-129  
 Saint-Cyr, P. II-53  
 Sakai, Tomoya I-561  
 Salazar, Augusto III-171, III-349
- Salehizadeh, Mohammad III-329  
 Sallaberry, Arnaud III-534  
 Sanchez T., German I-602  
 Sandberg, Kristian II-107  
 Sankaranarayanan, Karthik I-381  
 Sarmis, T. II-584  
 Sauvaget, Catherine III-504  
 Savakis, Andreas I-509, II-606, III-301  
 Sayed, Usama II-574  
 Scalzo, Fabien II-292, III-359  
 Schaefer, Gerald I-173  
 Scharcanski, Jacob I-190  
 Scheuermann, Gerik I-316  
 Schulten, Klaus II-382  
 Schultz, Richard R. I-698  
 Schumann, Heidrun III-99  
 Sharma, Shipra III-458  
 Sherman, William R. II-361  
 Shi, Y. Justin II-222  
 Shontz, Suzanne M. III-119  
 Sicre, R. III-635  
 Simone, Gabriele II-491  
 Slaboda, Jill II-222  
 Smith, Marvin III-181  
 Smith, William A.P. II-139  
 Sojka, Eduard III-310  
 Sokolov, Valeriy I-173  
 Somanath, Gowri I-666, II-170  
 Someya, Satoshi I-306  
 Son, Jeany I-45  
 Song, Ran I-644  
 Song, SooMin I-45, III-209  
 Song, Zhan I-612, II-31, II-628, II-687  
 Sonka, Milan III-129  
 Sourin, Alexei III-564  
 Souvenir, Richard I-143, II-564  
 Spagnolo, Paolo III-291  
 Spurlock, Scott II-564  
 Srisuk, Sanun III-69  
 Šrubař, Štepán III-310  
 Stab, Christian III-514  
 Stadler, Peter F. I-316  
 Stocker, Herbert III-564  
 Stone, John E. II-382  
 Stöttinger, Julian II-75  
 Streib, Kevin III-613  
 Strong, Grant II-481  
 Suarez, Jordane III-524  
 Sundaram, Sudeep II-596  
 Sun, Feng-Tso III-1

- Sun, Jin I-296  
 Sur, Frédéric I-231  
 Suzuki, Hiromasa III-39  
 Synave, R. II-416  
 Tague, Rhys II-545  
 Takahashi, Shigeo I-328  
 Tamimi, Zakiya II-747  
 Ta, Vinh-Thong I-539, II-647  
 Tang, A.W.K. II-21  
 Tange, Manabu I-306  
 Tavakkoli, Alireza III-161  
 Taylor, Chris I-132  
 Teisseire, Maguelonne III-534  
 Teo, C.C. III-427  
 Teoh, Andrew B.J. III-427  
 Teoh, Soon Tee I-739  
 Tessendorf, Jerry I-74  
 Thoma, George R. III-261  
 Thorpe, Christopher I-296  
 Tilmant, Christophe III-219  
 Tome, P. I-461  
 Tominaga, Shoji I-181  
 Tompkins, R. Cortland III-49  
 Triki, Olfa III-544  
 Tsagkatakis, Grigorios I-509  
 Tu, Chunling III-320  
 Tzevanidis, K. II-584  
 Uhl, Andreas I-469, III-19  
 Ullrich, Alexander I-316  
 Unaldi, Numan III-474  
 Uno, Makoto I-181  
 Vandivort, Kirby L. II-382  
 Wyk, Barend Jacobus van III-320  
 Vera-Rodriguez, R. I-489  
 Vidal, Joseph A. II-394  
 Voisin, Yvon I-361  
 Wacker, Esther-Sabrina II-459  
 Waechter, Christian II-429  
 Walczak, Alan M. III-359  
 Wan, Jiang III-397  
 Wang, Chaoli III-437  
 Wang, Fei I-96  
 Wang, Tinghui III-397  
 Wang, Xiaoyu II-564  
 Wang, Yao II-312  
 Wang, Zibin III-280  
 Ward, Matthew O. II-522  
 Wei, Lei III-564  
 Weiskopf, Daniel I-338, III-447  
 Wernert, Eric A. II-361  
 Westall, James I-74  
 Whiting, Eric T. II-361  
 Widynski, Nicolas I-393  
 Wild, Peter I-469  
 Wilkin, Paul II-669  
 Wills, D. Scott I-211  
 Wills, Linda M. I-211  
 Wittman, Todd II-97  
 Wu, Jimmy I-592  
 Wurtele, Eve Syrkin I-350  
 Xie, Nianhua I-296, II-222  
 Xie, Wuyuan II-31  
 Xie, Zaixian II-522  
 Xi, Junqiang III-368  
 Xiong, Guangming III-368, III-407  
 Xu, Huihui III-417  
 Xue, Daqing II-511  
 Yang, Hanxuan II-628, II-687  
 Yang, Heekyung I-86  
 Yang, Huei-Fang II-322  
 Yang, Michael Ying I-654  
 Yang, Yunyun II-117  
 Ye, Jian II-97  
 Ying, Xianghua II-151  
 Youssef, Menatoallah III-49  
 Yu, Jingyi I-296  
 Zabulis, X. II-584  
 Zambanini, Sebastian I-163  
 Zéraï, Mourad III-544  
 Zha, Hongbin II-151  
 Zhang, Liang III-189  
 Zhang, Qiang III-249  
 Zhang, Xiaolong II-31, II-449  
 Zhang, Xiaolong B. II-21  
 Zhang, Ying III-1  
 Zhang, Yinhui III-377  
 Zhang, Yunsheng III-377  
 Zhao, Feng III-397  
 Zhao, Yanguo II-628  
 Zhao, Ye II-747  
 Zheng, Feng II-628  
 Zheng, Nanning I-96  
 Zhu, Yongxin III-397  
 Zweng, Andreas I-163