

LNCS 6718

José Francisco Martínez-Trinidad
Jesús Ariel Carrasco-Ochoa
Cherif Ben-Youssef Brants
Edwin Robert Hancock (Eds.)

Pattern Recognition

Third Mexican Conference, MCPR 2011
Cancun, Mexico, June/July 2011
Proceedings

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

José Francisco Martínez-Trinidad
Jesús Ariel Carrasco-Ochoa
Cherif Ben-Youssef Brants
Edwin Robert Hancock (Eds.)

Pattern Recognition

Third Mexican Conference, MCPR 2011
Cancun, Mexico, June 29 - July 2, 2011
Proceedings



Springer

Volume Editors

José Francisco Martínez-Trinidad
National Institute of Astrophysics, Optics and Electronics (INAOE)
Computer Science Department
Luis Enrique Erro No. 1, 72840 Sta. Maria Tonantzintla, Puebla, Mexico
E-mail: fmartine@inaoe.mx

Jesús Ariel Carrasco-Ochoa
National Institute for Astrophysics, Optics and Electronics (INAOE)
Computer Science Department
Luis Enrique Erro No. 1, 72840 Sta. Maria Tonantzintla, Puebla, Mexico
E-mail: ariel@inaoe.mx

Cherif Ben-Youssef Brants
Cancun Technological Institute (ITC)
Av. Kabah, Km. 3, 77515 Cancun, Quintana Roo, Mexico
E-mail: cherifby@itcancun.edu.mx

Edwin Robert Hancock
University of York, Department of Computer Science
Deramore Lane, York, YO10 5GH, UK
E-mail: erh@cs.york.ac.uk

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-21586-5 e-ISBN 978-3-642-21587-2
DOI 10.1007/978-3-642-21587-2
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2011929350

CR Subject Classification (1998): I.4, I.5, I.2, H.3, H.4

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

© Springer-Verlag Berlin Heidelberg 2011

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.

The use of general descriptive names, registered names, trademarks, etc. in this publication does not imply, even in the absence of a specific statement, that such names are exempt from the relevant protective laws and regulations and therefore free for general use.

Typesetting: Camera-ready by author, data conversion by Scientific Publishing Services, Chennai, India

Printed on acid-free paper

Springer is part of Springer Science+Business Media (www.springer.com)

Preface

The Mexican Conference on Pattern Recognition 2011 (MCPR 2011) was the third event in the series organized by the Computer Science Department of the National Institute for Astrophysics Optics and Electronics (INAOE) of Mexico. This year the conference was organized in conjunction with the Cancun Technological Institute, and under the auspices of the Mexican Association for Computer Vision, Neurocomputing and Robotics (MACVNR), which is affiliated to the International Association for Pattern Recognition (IAPR). This conference aims to provide a forum for the exchange of scientific results, practice, and new knowledge, as well as promoting co-operation among research groups in pattern recognition and related areas in Mexico, Central America and the world.

MCPR 2011 was held in Cancun, Mexico. As in the second edition, MCPR 2011 attracted worldwide participation. Contributions were received from 17 countries. In total 69 papers were submitted, out of which 37 were accepted for publication in these proceedings and for presentation at the conference.

The conference was enriched by the contributions made by the three invited speakers:

- Kim Boyer (IAPR invited speaker), Head of the Department of Electrical, Computer, and Systems Engineering at Rensselaer, USA
- Joachim M. Buhmann, Department of Computer Science, Institute of Computational Science, ETH Zurich, Switzerland
- Carlos Coello Coello, Department of Computer Science, CINVESTAV-IPN, Mexico

We would like to express our sincere gratitude to the invited speakers. Thanks are also extended to Edwin Hancock for the help and discussions concerning the organization of this event.

The review process was carried out by the Scientific Committee, composed of internationally recognized scientists, all experts in their respective fields, which resulted in these excellent conference proceedings. We are indebted to them for their efforts and the quality of the reviews.

In addition, the authors of the best submissions will be invited to expand and further develop their papers for possible publication in a thematic special issue of the journal *Pattern Recognition Letters*, to be published in 2012.

We believe that the conference provided a fruitful forum to enrich the collaboration between the Mexican pattern recognition researchers and the broader international pattern recognition community. We hope this proceedings volume from the Third Mexican Conference on Pattern Recognition will prove useful to the reader.

July 2011

José Francisco Martínez-Trinidad
Jesús Ariel Carrasco-Ochoa
Cherif Ben-Youssef Brants
Edwin Hancock

Organization

MCPR 2011 was hosted and sponsored by the Computer Science Department of the National Institute of Astrophysics, Optics and Electronics (INAOE) and the Cancun Technological Institute.

General Conference Co-chairs

Edwin Hancock	Department of Computer Science at the University of York, UK
José Francisco Martínez-Trinidad	Computer Science Department, National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
Jesús Ariel Carrasco-Ochoa	Computer Science Department, National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
Cherif Ben-Youssef Brants	Cancun Technological Institute, Mexico

Local Arrangements Committee

Carmen Meza Tlalpan
Gorgonio Cerón Benítez
Gabriela López Lucio

Scientific Committee

Alquézar Mancho, R.	Universitat Politècnica de Catalunya, Spain
Asano, A.	Hiroshima University, Japan
Bagdanov, A.	Universitat Autònoma de Barcelona, Spain
Batyrshin, I.	Mexican Petroleum Institute
Bayro-Corrochano, E.	CINVESTAV-Guadalajara, Mexico
Benedi, J. M.	Universidad Politécnica de Valencia, Spain
Bigun, J.	Halmstad University, Sweden
Borges, D. L.	Universidade de Brasília, Brazil
Castelan, M.	CINVESTAV-Chihuahua, Mexico
Cesar, R.M.	University of São Paulo, Brazil
Corchado, E.	University of Burgos, Spain

Del Bimbo, A.	Università di Firenze, Italy
Dong, G.	Wright State University, USA
Ercil, A.	Boğaziçi University, Turkey
Facon, J.	Pontifícia Universidade Católica do Paraná, Brazil
Ferri, F.J.	Universitat de València, Spain
Gelbukh, A.	CIC-IPN, Mexico
Gibert, K.	Universitat Politècnica de Catalunya, Spain
Goldfarb, L.	University of New Brunswick, Canada
Graña, M.	University of the Basque Country, Spain
Grau, A.	Universitat Politècnica de Catalunya, Spain
Guzmán-Arenas, A.	CIC-IPN, Mexico
Haindl, M.	Institute of Information Theory and Automation, Czech Republic
Hanbury, A.	Vienna University of Technology, Austria
Hernando, J.	Universitat Politècnica de Catalunya, Spain
Heutte, L.	Université de Rouen, France
Hlavac, V.	Czech Technical University, Czech Republic
Igual, L.	University of Barcelona, Spain
Jiang, X.	University of Münster, Germany
Kampel, M.	Vienna University of Technology, Austria
Kim, S.W.	Myongji University, Republic of Korea, Korea
Klette, R.	University of Auckland, New Zealand
Kober, V.	CICESE, Mexico
Koster, W.	Universiteit Leiden, The Netherlands
Kropatsch, W.	Vienna University of Technology, Australia
Laurendeau, D.	Université Laval, Canada
Lopez de Ipiña, K.	University of the Basque Country, Spain
Lorenzo-Ginori, J. V.	Universidad Central de Las Villas, Cuba
Mascarenhas, N.D.	University of São Paulo, Brazil
Mayol-Cuevas, W.	University of Bristol, UK
Mejail, M.	Universidad de Buenos Aires, Argentina
Mora, M.	Catholic University of Maule, Chile
Morales, E.	INAOE, Mexico
Murino, V.	University of Verona, Italy
Nolazco, J.A.	ITESM-Monterrey, Mexico
Pardo, A.	Universidad Católica del Uruguay, Uruguay
Pérez de la Blanca-Capilla, N.	Universidad de Granada, Spain
Petrou, M.	Imperial College, UK

Pina, P.	Instituto Superior Técnico, Portugal
Pinho, A.	University of Aveiro, Portugal
Pinto, J.	Instituto Superior Técnico, Portugal
Pistori, H.	Dom Bosco Catholic University, Brazil
Raposo-Sanches, J.M.	Instituto Superior Técnico, Portugal
Real, P.	University of Seville, Spain
Rodríguez, R.	ICIMAF, Cuba
Ross, A.	West Virginia University, USA
Rueda, L.	University of Windsor, Canada
Ruiz-Shulcloper, J.	CENATAV, Cuba
Sánchez, J.S.	Universitat Jaume I, Spain
Sanniti di Baja, G.	Istituto di Cibernetica, CNR, Italy
Sansone, C.	Università di Napoli, Italy
Santana, R.	Universidad Politécnica de Madrid, Spain
Shirai, Y.	Ritsumeikan University, Japan
Shmaliy, Y.S.	Guanaajuato University, Mexico
Sossa Azuela, J. H.	CIC-IPN, Mexico
Sousa-Santos, B.	Universidade de Aveiro, Portugal
Stathaki, T.	Imperial College London, UK
Sucar, L. E.	INAOE, Mexico
Torres, M. I.	University of the Basque Country, Spain
Valev, V.	Institute of Mathematics and Informatics, Bulgaria
Wang, S.	University of Sherbrooke, Canada

Additional Referees

Ayala-Raggi, S.	Mottalli, M.
Ballan, L.	Olvera-López, J.A.
Calvo De Lara, J.R.	Piro, P.
Cerri, A.	Raghavendra, R.
Duval Poo, M.A.	Reyes-García, C.A.
Escalante-Balderas, H.J.	Rezaei, M.
Gago-Alonso, A.	San Biagio, M.
Hermann, S.	Silva, A.
Li, N.	Vega-Pons, S.
Morales, S	Villaseñor-Pineda, L.

Sponsoring Institutions

National Institute of Astrophysics, Optics and Electronics (INAOE)
 Cancun Technological Institute (ITCancun)
 Mexican Association for Computer Vision, Neurocomputing and Robotics
 (MACVNR)
 International Association for Pattern Recognition (IAPR)

Table of Contents

Kenote Addresses

Resilient Subclass Discriminant Analysis with Application to Prelens Tear Film Interferometry	1
<i>Kim L. Boyer and Dijia Wu</i>	
Context Sensitive Information: Model Validation by Information Theory	12
<i>Joachim M. Buhmann</i>	
Evolutionary Multi-Objective Optimization: Basic Concepts and Some Applications in Pattern Recognition	22
<i>Carlos A. Coello Coello</i>	

Pattern Recognition and Data Mining

Comparative Diagnostic Accuracy of Linear and Nonlinear Feature Extraction Methods in a Neuro-oncology Problem	34
<i>Raúl Cruz-Barbosa, David Bautista-Villavicencio, and Alfredo Vellido</i>	
Efficient Group of Permutants for Proximity Searching.....	42
<i>Karina Figueroa Mora, Rodrigo Paredes, and Roberto Rangel</i>	
Solving 3-Colouring via 2SAT	50
<i>Guillermo De Ita, César Bautista, and Luis C. Altamirano</i>	
Classifier Selection by Clustering	60
<i>Hamid Parvin, Behrouz Minaei-Bidgoli, and Hamideh Shahpar</i>	
Ensemble of Classifiers Based on Hard Instances	67
<i>Isis Bonet, Abdel Rodríguez, Ricardo Grau, and María M. García</i>	
Scalable Pattern Search Analysis	75
<i>Eric Sadit Tellez, Edgar Chavez, and Mario Graff</i>	
Application of Pattern Recognition Techniques to Hydrogeological Modeling of Mature Oilfields	85
<i>Leonid Sheremetov, Ana Cosultchi, Ildar Batyrshin, and Jorge Velasco-Hernandez</i>	
On Trend Association Analysis of Time Series of Atmospheric Pollutants and Meteorological Variables in Mexico City Metropolitan Area	95
<i>Victor Almanza and Ildar Batyrshin</i>	

Associative Memory Approach for the Diagnosis of Parkinson's Disease	103
<i>Elena Acevedo, Antonio Acevedo, and Federico Felipe</i>	

Computer Vision and Robotics

Thermal Video Analysis for Fire Detection Using Shape Regularity and Intensity Saturation Features	118
<i>Mario I. Chacon-Murguia and Francisco J. Perez-Vargas</i>	
People Detection Using Color and Depth Images	127
<i>Joaquín Salas and Carlo Tomasi</i>	
Measuring Rectangularity Using GR-Signature.....	136
<i>Jihen Hentati, Mohamed Naouai, Atef Hamouda, and Christiane Weber</i>	
Multi-modal 3D Image Registration Based on Estimation of Non-rigid Deformation.....	146
<i>Roberto Rosas-Romero, Oleg Starostenko, Jorge Rodríguez-Asomoza, and Vicente Alarcon-Aquino</i>	
Performance of Correlation Filters in Facial Recognition	155
<i>Everardo Santiago-Ramirez, J.A. Gonzalez-Fraga, and J.I. Ascencio-Lopez</i>	
Evaluation of Binarization Algorithms for Camera-Based Devices	164
<i>M. Nava-Ortiz, W. Gómez-Flores, A. Díaz-Pérez, and G. Toscano-Pulido</i>	
A Hybrid Approach for Pap-Smear Cell Nucleus Extraction	174
<i>M. Orozco-Monteagudo, Hichem Sahli, Cosmin Mihai, and A. Taboada-Crispi</i>	

Image Processing

Segmentation of Noisy Images Using the Rank M-Type L-Filter and the Fuzzy C-Means Clustering Algorithm	184
<i>Dante Mújica-Vargas, Francisco J. Gallegos-Funes, and Rene Cruz-Santiago</i>	
Design of Correlation Filters for Pattern Recognition Using a Noisy Training Image	194
<i>Pablo M. Aguilar-González and Vitaly Kober</i>	
Image Fusion Algorithm Using the Multiresolution Directional-Oriented Hermite Transform	202
<i>Sonia Cruz-Techica and Boris Escalante-Ramirez</i>	

Normalized Cut Based Edge Detection	211
<i>Mario Barrientos and Humberto Madrid</i>	
Homogeneity Cues for Texel Size Estimation of Periodic and Near-Periodic Textures	220
<i>Rocio A. Lizarraga-Morales, Raul E. Sanchez-Yanez, and Victor Ayala-Ramirez</i>	
Adaptive Thresholding Methods for Documents Image Binarization	230
<i>Bilal Bataineh, Siti N.H.S. Abdullah, K. Omar, and M. Faidzul</i>	
Foveated ROI Compression with Hierarchical Trees for Real-Time Video Transmission	240
<i>J.C. Galan-Hernandez, V. Alarcon-Aquino, O. Starostenko, and J.M. Ramirez-Cortes</i>	
Neural Networks and Signal Processing	
Neural Networks to Guide the Selection of Heuristics within Constraint Satisfaction Problems	250
<i>José Carlos Ortiz-Bayliss, Hugo Terashima-Marín, and Santiago Enrique Conant-Pablos</i>	
Microcalcifications Detection Using PFCM and ANN	260
<i>A. Vega-Corona, J. Quintanilla-Domínguez, B. Ojeda-Magaña, M.G. Cortina-Januchs, A. Marcano-Cedeño, R. Ruelas, and D. Andina</i>	
Software Development Effort Estimation in Academic Environments Applying a General Regression Neural Network Involving Size and People Factors	269
<i>Cuauhtémoc López-Martín, Arturo Chavoya, and M.E. Meda-Campaña</i>	
An Ensemble of Degraded Neural Networks	278
<i>Eduardo Vázquez-Santacruz and Debrup Chakraborty</i>	
Genetic Fuzzy Relational Neural Network for Infant Cry Classification	288
<i>Alejandro Rosales-Pérez, Carlos A. Reyes-García, and Pilar Gómez-Gil</i>	
Speech Compression Based on Frequency Warped Cepstrum and Wavelet Analysis	297
<i>Francisco J. Ayala and Abel Herrera</i>	

Dust Storm Detection Using a Neural Network with Uncertainty and Ambiguity Output Analysis	305
<i>Mario I. Chacon-Murguía, Yearim Quezada-Holguín, Pablo Rivas-Perea, and Sergio Cabrera</i>	

Extraction of Buildings Footprint from LiDAR Altimetry Data with the Hermite Transform	314
<i>José Luis Silván-Cárdenas and Le Wang</i>	

Natural Language and Document Processing

Automatic Acquisition of Synonyms of Verbs from an Explanatory Dictionary Using Hyponym and Hyperonym Relations	322
<i>Noé Alejandro Castro-Sánchez and Grigori Sidorov</i>	

Using Finite State Models for the Integration of Hierarchical LMs into ASR Systems	332
<i>Raquel Justo and M. Inés Torres</i>	

Use of Elliptic Curves in Term Discrimination	341
<i>Darnes Vilariño, David Pinto, Carlos Balderas, Mireya Tovar, Beatriz Beltrán, and Sofia Paniagua</i>	

Author Index	351

Resilient Subclass Discriminant Analysis with Application to Prelens Tear Film Interferometry*

Kim L. Boyer¹ and Dijia Wu^{1,2}

¹ Signal Analysis and Machine Perception Laboratory
Department of Electrical, Computer, and Systems Engineering
Rensselaer Polytechnic Institute
Troy, NY, USA

² Siemens Corporate Research
Princeton, NJ, USA
kim@ecse.rpi.edu, dija.wu@siemens.com

Abstract. The study of tear film thickness and breakup has important implications for understanding tear physiology and dynamics. We have developed a complete end-to-end automated system for robust and accurate measurements of the tear film thickness from interferometric video as a function of position and time (following a blink). This paper will primarily address the problem of identifying dry regions on the surface of the contact lens, which is one of the four major components of the system. (The others are motion stabilization, image normalization, and phase demodulation to infer absolute thickness and map the surface. To address the challenging wet/dry segmentation problem, we propose a new Gaussian clustering method for feature extraction in high dimensional spaces. Each class is modeled as a mixture of Gaussians, clustered using Expectation-Maximization in the lower-dimensional Fisher's discriminant space. We show that this approach adapts to a wide range of distributions and is insensitive to training sample size. We present experimental results on the real-world problem of identifying regions of breakup (drying) of the prelens tear film from narrowband interferometry for contact lens wearers *in vivo*.

Keywords: Mixture of Gaussians, Expectation-Maximization, Feature Extraction, Clustering, Prelens Tear Film, Interferometry, Dry Eye Syndrome.

1 Introduction

The thickness of the tear film on the surface of the human eye has important implications in the study of tear film physiology and its fluid dynamics. Knowledge of the tear film thickness as a function of time and position over the eye following a

* This paper corresponds to an invited keynote address and contains some material that previously appeared in the proceedings of the 2009 *IEEE International Conference on Computer Vision*.

blink is necessary to develop and/or verify models of tear film deposition [1] and to analyze the flow in the tear film. For example, the surface tension gradients of the tears will pull the tears upward toward regions of greater surface tension, which is thought to account for the upward drift of the film surface after a blink. For given surface tension gradients and tear viscosity, the tear velocity will be proportional to tear film thickness and the total flow will be proportional to the square of the thickness [2]. In addition, the curvature of the outer surface of the tear film generates pressure in the film and variations in the surface curvature will therefore cause tangential tear flow. For given pressure gradients and viscosity, the tear velocity will be proportional to the square of tear film thickness, and the total rate of flow will be proportional to the cube of thickness [1, 3]. The same dependence on tear film thickness also applies to the downward flow arising from gravity, but this is a small effect [4] for a normal tear film thickness of about 3 to 4 μm [5-7].

In this study we focus on the thickness measurements of one particular type of tear film, the prelens tear film (PLTF) on the surface of a contact lens. That is, the thickness we will measure is the distance between the air surface of the tears and the anterior contact lens surface (more on this below). The PLTF is particularly important for several specific reasons. First, the outer layer of the tears provides a uniform coating over the contact lens, making it a smooth (*i.e.* low distortion) optical surface. If the outer layer becomes rough or irregular as a result of tear drying and breakup, light will be scattered and image (vision) quality impaired. Another function of the PLTF is to provide comfort and lubrication to the palpebral conjunctiva, especially during the blink. In addition, the superficial lipid layer of the tear film reduces evaporation of the film, maintaining contact lens hydration. Dryness and discomfort with contact lens wearers have been reported by as many as 50% of contact lens wearers and are two major reasons associated with premature contact lens discontinuation [8]. The increased evaporation of the PLTF, followed by contact lens dehydration and depletion of the post-lens tear film by absorption into the contact lens, may be the mechanism of contact lens related dry eye in these subjects.

Despite the importance of tear film thickness to understanding its behavior with an eye to developing therapies for dry eye problems, the true thickness of the tear film under various conditions (prelens, post lens, precorneal) remains controversial; widely different results have appeared in the literature [9]. The underlying reason for these highly variable results arises from the significant challenges presented in making such a measurement. Current methods of measuring human tear film thickness can be categorized as invasive or non-invasive. Invasive methods require the insertion of substances or objects into the tear film, such as absorbent paper discs [4] and fluorescein [10, 11] and are generally inconsistent. Non-invasive approaches are limited to various forms of interferometry. Among these, angle-dependent fringes [12, 13] are well-suited only for films thicker than the lipid, tear film layers; wavelength-dependent fringes [14, 15] can measure only at a single location each time; but thickness-dependent fringes [16] can provide a two-dimensional distribution of thickness over the full surface and for a properly chosen wavelength can properly handle films as thin as the tear layer. However, thickness-dependent fringes produce only *relative* depth information, and even the depth gradient orientation is ambiguous, unless a reference level can be provided. This approach is unsuitable for studying a flat, unchanging surface – but that is not a concern in this domain.

This paper shows how to provide a reference level (zero, corresponding to a dry lens surface) by segmenting a video frame into wet and dry regions. The larger body of work of which this paper describes only a part solves the phase ambiguity problems and the complete system produces a time-varying map of tear film depth over the eye's surface.

We present a novel contribution to pattern recognition and computer vision in this paper, motivated by the challenging problem of interpreting interferometric video of the surface of the human eye and the associated pretears tear film. The contribution addresses a difficult wet/dry texture segmentation problem – but represents a more fundamental contribution to Gaussian mixture models generally.

The popularity of Gaussian mixture models (GMM) for density estimation derives from their flexibility in representing the wide variety of distributions encountered in real applications [17-19]. Approximating the distribution of each class as a weighted sum of Gaussians, as in Subclass Discriminant Analysis (SDA) [20] or Mixture Discriminant Analysis (MDA) [21, 22], renders the resulting classifier more adaptable. When each class comprises multiple disjoint clusters (subclasses), the gain over traditional discriminant analysis techniques (*e.g.* Fishers' DA or Linear DA) can be especially significant.

2 Resilient Subclass Discriminant Analysis: Clustering

A key problem that arises in applying GMM is that of clustering the data to identify the individual component Gaussians that represent the subclasses. SDA [20] uses a fast nearest neighbor (NN) clustering method that works well even for small sample sizes – but it assumes equal subclass priors, which is unrealistic in most applications. Moreover, in using the Euclidean metric, NN is sensitive to the number of irrelevant (uninformative) features. Expectation-Maximization (EM) is a popular and powerful algorithm with good convergence properties that has found success with MDA [22] in a number of applications [19]. However, EM requires a larger training set to produce accurate estimates of the mixture parameters, and becomes unstable when the within-subclass scatter matrices approach singularity, which is typical in problems having high dimensionality and small training sets.

This paper presents a simple, reliable clustering method we call Resilient Subclass Discriminant Analysis (RSDA) that can estimate the Gaussian mixture parameters irrespective of training set size. The underlying concept is that, in each EM iteration LDA is first used to project the data onto a much lower dimensional space with maximum class separability, and the data are then clustered in this new space. When compared to the conventional EM approach, the proposed technique offers improved numerical stability because the new subclass covariance matrices are much smaller and therefore more likely to be invertible for a given sample size. Moreover, this approach often reduces the computational complexity, despite the added projection step at each iteration, because the assignment of samples to subclasses, the most computationally demanding step in conventional EM, is now accomplished on a space of far lower dimensionality.

We point out that our algorithm differs from the Gaussian parsimonious clustering models [25, 26], which are also supposed to improve robustness with respect to small

training sample size. As opposed to the adaptability of RSDA, these GMM parameterization models impose specific restrictions on the class covariance matrices and, therefore, can fit only certain classes of distributions.

2.1 RSDA Computational Procedure

We describe the RSDA clustering procedure below. In the following, N_c and N_t are the sample sizes for classes c and t , respectively, and N is the total sample size. The total number of classes is C , each with M_c subclasses.

- 1 Given an assumption for the number of subclasses (clusters) M_c in class c , use the K-means algorithm and accept the clustering results $\{d_{ji}; d_{ji} = 1 \text{ if } x_j \in \text{cluster } i; 0 \text{ else}\}$ to initialize the subclass priors, mean vectors, and covariance matrices:

$$\rho_{ci}^{[0]} = \frac{\sum_{j=1}^{N_c} d_{ji}}{N_c} \quad (1a) \quad \mu_{ci}^{[0]} = \frac{\sum_{j=1}^{N_c} d_{ji} x_j}{\sum_{j=1}^{N_c} d_{ji}} \quad (1b)$$

$$\Sigma_{ci}^{[0]} = \frac{\sum_{j=1}^{N_c} d_{ji} (x_j - \mu_{ci}^{[0]})(x_j - \mu_{ci}^{[0]})^T}{\sum_{j=1}^{N_c} d_{ji}} \quad (1c)$$

- 2 At iteration k , calculate the averaged within subclass and between subclass scatter matrices:

$$\hat{S}_W = \frac{1}{N} \sum_{c=1}^C \sum_{i=1}^{M_c} N_c \rho_{ci}^{[k]} \Sigma_{ci}^{[k]} \quad (2)$$

$$\hat{S}_B = \frac{1}{N^2} \sum_{c=1}^{C-1} \sum_{i=1}^{M_c} \sum_{t=c+1}^C \sum_{l=1}^{M_t} N_c N_t \rho_{ci}^{[k]} \rho_{tl}^{[k]} (\mu_{ci}^{[k]} - \mu_{tl}^{[k]})(\mu_{ci}^{[k]} - \mu_{tl}^{[k]})^T \quad (3)$$

Eq. (3) differs from the traditional definition of between class scatter in that it emphasizes class separability over intra-subclass scatter. For comparison, the more traditional definition is:

$$\tilde{S}_B = \frac{1}{N} \sum_{c=1}^C \sum_{i=1}^{M_c} N_c \rho_{ci}^{[k]} (\mu_{ci}^{[k]} - \mu^{[k]})(\mu_{ci}^{[k]} - \mu^{[k]})^T \quad (4)$$

where $\mu^{[k]}$ is the total mean vector. Note that, because we use Fisher's discriminant variates for dimensionality reduction rather than direct LDA classification, the subclass covariance matrices are not constrained to be the same so long as the dimension reduction model (DRM) and linear design condition (LDC) assumptions are valid [23].

- 3 Compute $p = (\Sigma_c M_c) - 1$ discriminant vectors V by solving for the generalized eigenvectors of the between subclass scatter matrix with respect to the within subclass scatter matrix:

$$\hat{S}_B V = \hat{S}_W V \Lambda \quad (5)$$

Λ is a $p \times p$ diagonal matrix of generalized eigenvalues sorted by size. We next use V to project the original (high-dimensional) samples $\{x_j\}$ onto the p -dimensional Fisher's space: $x_j' = V^T x_j$. The subclass means and sample covariance matrices are also projected onto the lower dimensional space:

$$\mu_{ci}^{[k]'} = V^T \mu_{ci}^{[k]} \quad \Sigma_{ci}^{[k]'} = V^T \Sigma_{ci}^{[k]} V$$

- 4 With the estimated parameters (means, covariances) and the projected discriminant coordinates in hand, we can calculate the probability of each sample x_j' being drawn from each subclass i in class c using Bayes' rule and the law of total probability (the E-step):

$$p(z_j = i | x_j', \theta_c^{[k]}) = \frac{\rho_{ci}^{[k]} g(x_j'; \mu_{ci}^{[k]'}, \Sigma_{ci}^{[k]'})}{\sum_{i=1}^{M_c} \rho_{ci}^{[k]} g(x_j'; \mu_{ci}^{[k]'}, \Sigma_{ci}^{[k]'})} \quad (6)$$

- 5 Now we can update the new subclass prior probabilities, mean vectors, and covariance matrices by maximizing the conditional expectation of the log-likelihood function (the M-step):

$$\begin{aligned} \rho_{ci}^{[k+1]} &= \frac{1}{N_c} \sum_{j=1}^{N_c} p(z_j = i | x_j', \theta_c^{[k]}) \\ \mu_{ci}^{[k+1]} &= \frac{1}{N_c} \sum_{j=1}^{N_c} \frac{x_j}{\rho_{ci}^{[k+1]}} p(z_j = i | x_j', \theta_c^{[k]}) \\ \Sigma_{ci}^{[k+1]} &= \frac{1}{N_c} \sum_{j=1}^{N_c} \frac{1}{\rho_{ci}^{[k+1]}} p(z_j = i | x_j', \theta_c^{[k]}) (x_j - \mu_{ci}^{[k+1]})(x_j - \mu_{ci}^{[k+1]})^T \end{aligned} \quad (7)$$

- 6 Return to Step 2 (recalculate the scatter matrices) and repeat to convergence.

As just shown, RSDA uses FDA to reduce the data dimension before each E-step based on the between and within subclass scatter matrices from the previous M-step. It is easy to show that the discriminant vectors obtained from Eq. (5) are the same as those obtained using the total covariance matrix, which remains constant throughout, in place of the average within subclass covariance matrix:

$$\hat{S}_B V = \hat{S}_T V \Lambda, \text{ where } \hat{S}_T = \hat{S}_B + \hat{S}_W$$

The null space of the total covariance matrix provides no (sub)class discrimination information [27]. Therefore, we can safely discard it prior to RSDA. As long as the subclass conditional covariance matrices in the reduced dimension space have full rank, numerical problems can be avoided entirely. It is usually far easier to meet the full rank condition in the lower dimensional space because $p \ll d$ in general. But because the estimates of the covariance matrices are biased prior to convergence, the resulting discriminant vectors are also biased. Accordingly, the class means and the covariance matrices must be updated back on the original high (d) dimensional feature space in each M step.

2.2 Computational Complexity

Relative to the conventional EM algorithm, the additional computational cost presented by RSDA arises from two stages of the procedure. First is the generalized eigenvector decomposition of the between subclass scatter matrix with respect to the within subclass scatter matrix in Step 3. This is usually solved by Cholesky factorization with a computational complexity of $O(d^3)$. The second source of additional cost is the projection of the original d -dimensional feature vectors onto the p -dimensional Fisher's discriminant space, which calls for $O(Npd)$ multiplications and additions. However, the calculation of the posterior probability in Step 4 (the E-step) is now much less costly because the Mahalanobis distance in Eq. (6) is computed in the lower dimensional space, saving $O(Nd^2 - Np^2)$ computations. Therefore, when the total sample size N is sufficiently large and the original feature space dimension d is significantly larger than the reduced one p – typical in many problems of interest, including our application (below) – substantial computational savings can be realized by RSDA vs. conventional EM.

2.3 Comparison with Other GMM Parameterization Models

RSDA improves the robustness of EM in Gaussian mixture estimation when the sample size is low relative to the dimensionality of the feature space. Other methods have been proposed to address this situation, including the so-called Parsimonious Gaussian Mixture Models [25, 26]. These models place restrictions on the class conditional covariance matrices, reducing the number of free parameters to estimate and, thereby, stabilizing the numerical process.

Our results [28, 29] in comparing these models show that RSDA – by not enforcing such constraints – is capable of effectively modeling a far greater range of data sets, with corresponding improvements in classification. Moreover, RSDA can accomplish this while also being more computationally efficient in most cases, especially with respect to those models requiring iterative calculation of adjustable diagonal and adjustable positive semidefinite symmetric matrices, respectively.

3 Segmentation of the PLTF from Narrowband Interferometry

We demonstrate the capabilities of RSDA with a set of experiments on our own problem in computer vision: texture-based segmentation of narrowband interferometry of the prelens tear film *in vivo*. We established the optimal number of

clusters per class using 10-fold cross validation. The quadratic classifier based on the Mahalanobis distance from each (sub)class centroid is then used.

Image segmentation based on visual texture is a classic problem in computer vision. In this experiment (which motivated our developing RSDA in the first place), we begin with a set of prelens tear film (PLTF) images acquired using a modified Doane's interferometer [30, 31]. This custom instrument superposes two (or more) coherent light waves, one reflected from the surface of the tear film and the other from the anterior surface of the contact lens, as shown in Fig. 1. The resulting oscillations in net reflected intensity as the waves constructively and destructively interfere, analogous to Newton's rings, indicate the thickness of the tear film as a function of position and time (we collect video), modulo half the optical wavelength.

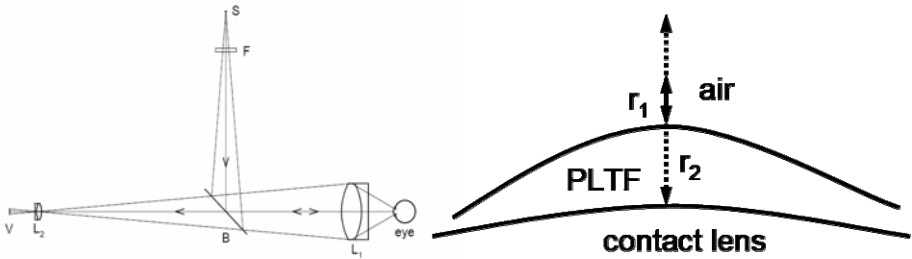


Fig. 1. [Left] Simple schematic of a modified Doane's interferometer, used to image the prelens tear film (PLTF). [Right] Principle of operation. An incident wave is reflected by two surfaces; r_1 is the reflection from the tear film surface and r_2 is the reflection from the contact lens.

Fig. 2 presents a pair of typical PLTF frames, each including dry (zero thickness) areas in which the tear film has broken up and areas that remain wet. The detection of dry areas and the assessment of the tear film thickness as a function of position and time (post-blink) is important to the study of tear physiology and fluid dynamics associated with dry eye syndromes in contact lens wearers. As we see in the images, the wet and dry regions present distinct visual textures. It follows that detecting and measuring the location and extent of tear film breakup can be formulated as a texture segmentation problem. We begin by computing a standard feature set of Gabor filter bank responses on a pixel-by-pixel basis, followed by RSDA to classify the extracted feature vectors; more detail is available in [32].

An inspection of the images reveals that wet regions are relatively uniform in texture, displaying sinusoidal fringes whose contours and spatial frequency correspond to the local gradient of the tear film surface. Our experiments show that wet regions can be classified as arising from a single Gaussian distribution. However, two distinct texture patterns exist in dry regions, arising primarily from differences in the contact lens material from one manufacturer to another, or from one lens type to another. The "dry class" is therefore multimodal, requiring multiple Gaussian subclasses to capture the complete distribution. Some form of GMM is therefore desirable. In particular, owing to the ratio of dimensionality to training data, RSDA is an excellent tool to use against this classification problem.

We determined the number of subclasses by running RSDA with different trial values for the numbers of subclasses: $M_{DRY} \times M_{WET} = \{1, 2, \dots, 6\} \times \{1, 2, \dots, 6\}$. Increasing the number of dry subclasses from 1 to 2 improved classification performance by about 6% (e.g., from 84% to 90%); this improvement was relatively insensitive to the number of wet subclasses assumed. However, so long as $M_{DRY} \geq 2$, performance was relatively insensitive to the value of M_{WET} , ranging from a low of 89.6% to a high of 91.0% for $M_{DRY} = 4$. Although we believe this range is so small as to be statistically insignificant (and thus confirming our hypothesis of two dry subclasses), we assumed four dry subclasses for the results to follow – thus forcing RSDA to build a richer model of the dry class conditional distribution.

The classification rate for $M_{DRY} = M_{WET} = 1$, in which case RSDA reduces to LDA, was relatively poor at just 84.0%. The poorest performance among all 36 combinations was achieved with $M_{DRY} = 6$ and $M_{WET} = 1$, corresponding to over modeling the wet class conditional distribution (more subclasses than needed) while under modeling the dry (insufficient number of subclasses). But, given the observations in the preceding paragraph, the real culprit is clearly the under modeling of the dry class conditional distribution. Tests with the other parsimonious GMM parameterizations failed to achieve the same level of performance; those assuming diagonal subclass conditional covariance matrices achieved the best rate among these models at just over 88%.

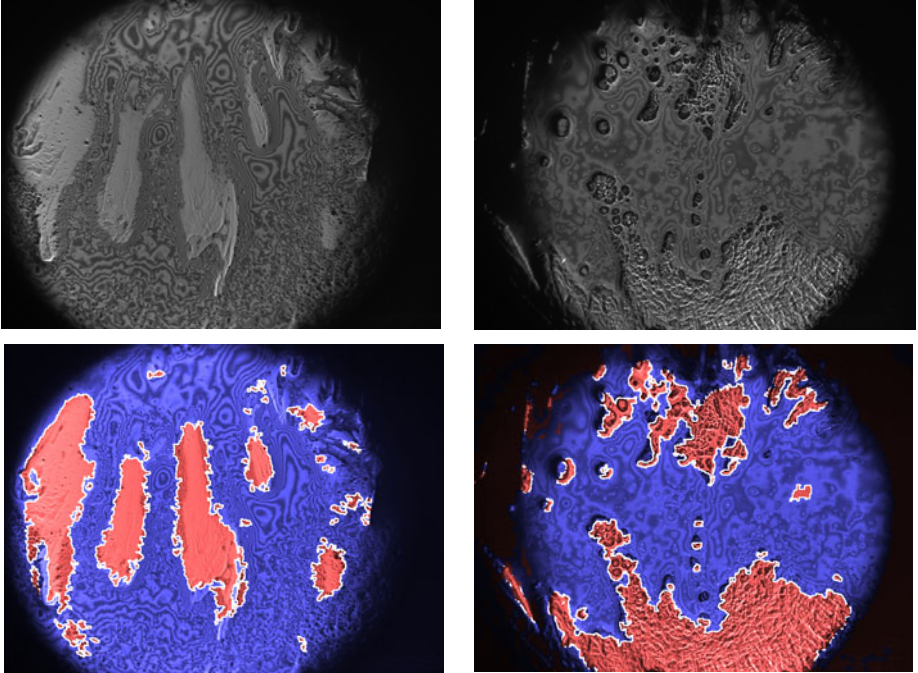


Fig. 2. PLTF interferometry. Originals (top) and segmented (bottom) using RSDA feature selection from Gabor filter outputs. Dry regions are indicated in red; wet regions in blue. Ground truth for training and classification rate calculation verified by faculty in the Ohio State University College of Optometry. Note the differences in the texture of the dry regions: very smooth for the left example and very rough for the example on the right.

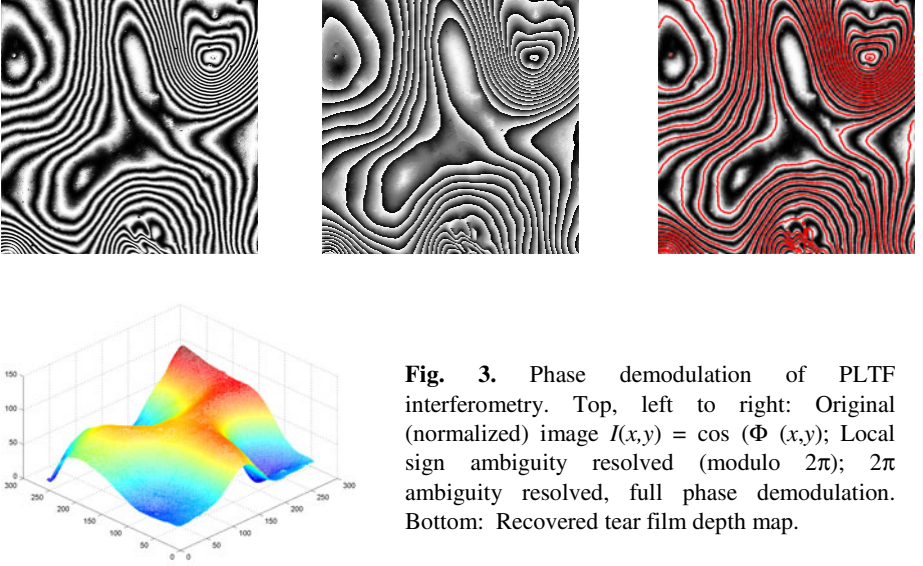


Fig. 3. Phase demodulation of PLTF interferometry. Top, left to right: Original (normalized) image $I(x,y) = \cos(\Phi(x,y))$; Local sign ambiguity resolved (modulo 2π); 2π ambiguity resolved, full phase demodulation. Bottom: Recovered tear film depth map.

4 Final Comments

Our new Gaussian clustering method is based on the simple idea of conducting the E-step in the lower dimensional Fisher space. It offers improved numerical stability compared to conventional EM for small training sets, while effectively accommodating a wider variety of distributions without restrictions on the subclass conditional covariance matrices. Using this approach, we successfully tackled the challenging problem of detecting dry spots on the surfaces of contact lens wearers *in vivo* from interferometric video. With pixel-based texture segmentation, we detect the tear film breakup from feature vectors extracted using a bank of 30 Gabor filters tuned to five different spatial frequencies and six orientations. The E-step, however, is accomplished in a four dimensional space, based on the five total subclasses (one wet, four dry), assumed for these tests. The results are quite accurate: 91.0% of pixels are correctly classified (wet or dry) using RSDA, compared to 84.0% using the well-known LDA, or 84.9% using traditional EM followed by a Bayesian classifier. A three subclass (two dry) solution, using a two dimensional Fisher space would likely work just as well.

Owing to space limitations, the discussion here has been confined to outlining RSDA and demonstrating its effectiveness in wet/dry segmentation of the prelens tear film. However, to show something about “the rest of the story” we present one reconstructed frame of a prelens tear film depth map in Fig. 3. The original normalized image is first decoded with the inverse cosine (not shown) and a sign image is computed (not shown) in an optimization step based on the reasonable assumption that the *phase gradient* (as opposed to the *image gradient*) is smooth everywhere, except at critical points. A wrap count image (not shown) is then produced in an optimization step based on the Itoh condition [33] to resolve the 2π

ambiguity and complete the phase demodulation. Scaling according to the wavelength of the light source then produces the depth map. For details and more results see [34, 35].

References

1. Wong, H., Fatt, I., Radke, C.J.: Deposition and thinning of the human tear film. *J. Colloid and Interface Science* 184(1), 44–51 (1996)
2. Berger, R.E., Corrsin, S.: A surface tension gradient mechanism for driving the pre-corneal tear film after a blink. *J. Biomechanics* 7, 225–238 (1974)
3. Sharma, A., Tiwari, S., Khanna, R., Tiffany, J.M.: Hydrodynamics of meniscus-induced thinning of the tear film. *Adv. Exp. Med. Biol.* 438, 425–431 (1998)
4. Ehlers, N.: The thickness of the precorneal tear film. *Acta Ophthalmol (Copenh)* 81, 92–100 (1965)
5. King-Smith, P.E., Fink, B.A., Fogt, N., Nichols, K.K., Hill, R.M., Wilson, G.S.: The thickness of the human precorneal tear film: Evidence from reflection spectra. *Invest. Ophthalmol. Vis. Sci.* 41(11), 3348–3359 (2000)
6. Fogt, N., King-Smith, P.E., Tuell, G.: Interferometric measurement of tear film thickness by use of spectral oscillations. *J. Opt. Soc. Am. A* 15(1), 268–275 (1998)
7. Wang, J., Fonn, D., Simpson, T.L., Jones, L.: Precorneal and pre- and postlens tear film thickness measured directly with optical coherence tomography. *Invest. Ophthalmol. Vis. Sci.* 44, 2524–2528 (2003)
8. Doughty, M., Fonn, D., Richter, D., Simpson, T., Coffrey, B., Gordon, K.: A patient questionnaire approach to estimating the prevalence of dry eye symptoms in patients presenting to optometric practices across Canada. *Optom. Vis. Sci.* 74, 624–631 (1997)
9. King-Smith, P.E., Fink, B.A., Hill, R.M., Koelling, K.W., Tiffany, J.M.: The thickness of the tear film. *Current Eye Research* 29(4-5), 357–368 (2004)
10. Benedetto, D.A., Clinch, T.E., Laibson, P.R.: In vivo observation of tear dynamics using fluorophotometry. *Arch. Ophthalmol.* 102(3), 410–412 (1984)
11. Benedetto, D.A., Shah, D.O., Kaufman, H.E.: The instilled fluid dynamics and surface chemistry of polymers in the precocular tear film. *Invest. Ophthalmol. Vis. Sci.* 14, 887–902 (1975)
12. Green, D.G., Frueh, B.R., Shapiro, J.M.: Corneal thickness measured by interferometry. *J. Opt. Soc. Am.* 65, 119–123 (1975)
13. Prydal, J.I., Artal, P., Woon, H., Campbell, F.W.: Study of human precorneal tear film thickness and structure using laser interferometry. *Invest. Ophthalmol. Vis. Sci.* 33, 2006–2011 (1992)
14. King-Smith, P.E., Fink, B.A., Fogt, N.: Three interferometric methods for measuring the thickness of the layers of the tear film. *Optom. Vis. Sci.* 76, 19–32 (1999)
15. Nichols, J.J., King-Smith, P.E.: Thickness of the pre- and post-contact lens tear film measured in vivo by interferometry. *Invest. Ophthalmol. Vis. Sci.* 44, 68–77 (2003)
16. Doane, M.G., Gelason, W.J.: Tear layer mechanics. *Clinical Contact Lens Practice*, 1–17 (1994)
17. McLachlan, G.J., Basford, K.E.: *Mixture Models: Inference and Applications to Clustering*. M. Dekker, New York (1988)
18. Jain, A.K., Duin, R.P.W., Mao, J.: Statistical Pattern Recognition: A review. *IEEE Trans. Pattern Anal. Mach. Intell.* 22(1), 4–37 (2000)

19. Ju, J., Kolaczyk, E.D., Gopal, S.: Gaussian mixture discriminant analysis and sub-pixel land cover classification in remote sensing. *Remote Sensing of Environment* 84(4), 550–560 (2003)
20. Zhu, M., Martinez, A.M.: Subclass discriminant analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* 28(8), 1274–1286 (2006)
21. Hastie, T., Tibshirani, R.: Discriminant analysis by Gaussian mixtures. *Journal of the Royal Statistical Society, Series B (Methodological)* 58(1), 155–176 (1996)
22. Hastie, T., Tibshirani, R., Buja, A.: Flexible discriminant and mixture models. In: Kay, J., Titterton, D. (eds.) *Statistics and Neural Networks: Advances at the Interface*. Oxford University Press, Oxford (1999)
23. Fisher, R.A.: The statistical utilization of multiple measurements. *Annals of Eugenics* 8, 376–386 (1938)
24. Rao, C.R.: *Linear Statistical Inference and its Applications*, 2nd edn. Wiley Interscience, Hoboken (2002)
25. Celeux, G., Govaert, G.: Gaussian parsimonious clustering models. *Pattern Recognition* 28(5), 781–793 (1995)
26. Halbe, Z., Aladjem, M.: Model-based mixture discriminant analysis – an experimental study. *Pattern Recognition* 38(3), 437–440 (2005)
27. Zhang, S., Sim, T.: Discriminant subspace analysis: A Fukunaga-Koontz approach. *IEEE Trans. Pattern Anal. Mach. Intell.* 29(10), 1732–1746 (2007)
28. Wu, D., Boyer, K.L.: Resilient subclass discriminant analysis. In: *12th International Conference on Computer Vision*, Kyoto, Japan, October 2009, pp. 389–396 (2009)
29. Wu, D., Boyer, K.L.: A new Gaussian clustering method for high dimensional classification problems. In: *International Conference on Pattern Recognition and Information Processing*, Minsk, Belarus (May 2009) (Invited keynote)
30. Doane, M.G.: An instrument for in vivo tear film interferometry. *Optom. Vis. Sci.* 66(6), 383–388 (1989)
31. King-Smith, P.E., Fink, B.A., Nichols, J.J., Nichols, K.K., Hill, R.M.: Interferometric Imaging of the full thickness of the precorneal tear film. *J. Opt. Soc. Am. A, Opt. Image Sci. Vis.* 23(9), 2097–2104 (2006)
32. Wu, D., Boyer, K.L.: Sign ambiguity resolution for phase demodulation in interferometry with application to prelens tear film analysis. In: *IEEE Conference on Computer Vision and Pattern Recognition*, San Francisco, CA (June 2010)
33. Itoh, K.: Analysis of the phase unwrapping problem. *Appl. Opt.* 21(14), 2470 (1982)
34. Wu, D., Boyer, K.L.: Markov random field based phase demodulation of interferometric images. *Computer Vision and Image Understanding* 115(6), 759–770 (2011)
35. Wu, D., Boyer, K.L., Nichols, J.J., King-Smith, P.E.: Texture based prelens tear film segmentation in interferometry images. *Machine Vision and Applications* 21, 253–259 (2010)

Context Sensitive Information: Model Validation by Information Theory

Joachim M. Buhmann

ETH Zürich, Departement Informatik, CAB G 69.2

Universitätsstraße 6, CH-8092 Zürich, Switzerland

jbuhmann@inf.ethz.ch

<http://www.ml.inf.ethz.ch>

Abstract. A theory of patterns analysis has to provide a criterion to filter out the relevant information to identify patterns. The set of potential patterns, also called hypothesis class of the problem, defines admissible explanations of the available data and it specifies the context for a patterns analysis task. Fluctuations in the measurements limit the precision which we can achieve to identify such patterns. Effectively, the distinguishable patterns define a code in a fictitious communication scenario where the selected cost function together with a stochastic data source plays the role of a noisy “channel”. Maximizing the capacity of this channel determines the penalized costs of the pattern analysis problem with a data dependent regularization strength. The tradeoff between *informativeness* and *robustness* in statistical inference is mirrored in the balance between high information rate and zero communication error, thereby giving rise to a new notion of context sensitive information.

1 Introduction

We are drowning in data, but starving for knowledge!

This often deplored dilemma of the information age characterizes our rudimentary understanding of the concept “information”: We are much more efficient in collecting and storing data than in analyzing them to extract the relevant bits for intelligent decision making. The information society is flooded with digital data but only a tiny fraction seems to enter the value chain of the information society. Raw data are generated at a much higher rate than we can convert these exascale data volumes into information and further refine them to knowledge. Only this transformation of digital data guarantees that digital information is a precious resource and that value is generated by information processing. To filter out the relevant information for solving a data analysis problem from the vast amount of superfluous signals and noise, we need a new concept of information – *context sensitive information*. This paper builds upon a theory of cluster analysis (see [3]) which enables the modeler to measure the informativeness of statistical models or, equivalently, to choose suitable cost functions for solving a pattern recognition problem.

Pattern recognition in statistics and machine learning requires to infer patterns or structures in data which are stable in the presence of noise perturbations. Typical examples of such structures are data partitionings / clusterings, low dimensional embeddings of high dimensional or non-metric data or inference of discrete structures like trees from relational data. In almost all of these cases, the hypothesis class of the potential patterns is much smaller than the data space. Consequently, only a fraction of the noise fluctuations in the measurements will interfere with our search for optimal structures or interpretations of the data. Cost functions and algorithms, which preserve the relevant signal in the data to identify patterns but still filter out measurement noise as much as possible, should be preferred over brittle, noise sensitive methods.

In this paper, we formalize pattern analysis problems as optimization problems of appropriately chosen cost functions. The cost function assigns low costs to preferred patterns and high costs to unfavorable ones. How reliably we can distinguish patterns in the hypothesis class, is measured by a fictitious communication process between a sender and a receiver. The condition of error free communication determines the smallest sets of statistically indistinguishable patterns. The theoretical framework, which generalizes the model validation method for clustering by approximation set coding [3], is founded on maximum entropy inference.

2 Statistical Learning for Pattern Recognition

Given are a **set of objects** $\mathbf{O} = \{o_1, \dots, o_n\} \in \mathcal{O}$ and **measurements** $\mathbf{X} \in \mathcal{X}$ to characterize these objects. $\mathcal{O}, \mathcal{X}$ denotes the object or measurement space, respectively. Such measurements might be d -dimensional vectors $\mathbf{X} = \{X_i \in \mathbb{R}^d, 1 \leq i \leq n\}$ or relations $\mathbf{D} = (D_{ij}) \in \mathbb{R}^{n \cdot n}$ which describe the (dis)-similarity between object o_i and o_j . More complicated data structures than vectors or relations, e.g., three-way data or graphs, are used in various applications. In the following, we use the generic notation $\mathbf{X}$ for measurements. Data denote object-measurement relations $\mathcal{O} \times \mathcal{X}$, e.g., vectorial data $\{X_i : 1 \leq i \leq n\}$ describe surjective relations between objects o_i and measurements $X_i := X(o_i)$.

The **hypotheses** of a pattern recognition problem are functions assigning data to patterns out of a pattern space $\mathcal{P}$, i.e.,

$$c : \mathcal{O} \times \mathcal{X} \rightarrow \mathcal{P}, \quad (\mathbf{O}, \mathbf{X}) \mapsto c(\mathbf{O}, \mathbf{X}) \quad (1)$$

The pattern space for clustering problems is the set of possible assignments of data to k groups, i.e., $\mathcal{P} = \{1, \dots, k\}^n$, $n = |\mathbf{O}|$ denoting the number of objects. For parameter estimation problems like PCA or SVD, the patterns are possible values of the orthogonal matrices and the pattern space is a subset of the d -dimensional Euclidean rotations. To simplify the notation, we omit the first argument of c in cases where $\mathbf{X}$ uniquely identifies the object set $\mathbf{O}$. A clustering is then defined as $c : \mathcal{X} \rightarrow \{1, \dots, k\}^n$.

The **hypothesis class** for a pattern recognition problem is defined as the set of functions assigning data to elements of the pattern space, i.e.,

$$\mathcal{C}(\mathbf{X}) = \{c(\mathbf{O}, \mathbf{X}) : \mathbf{O} \in \mathcal{O}\} \quad (2)$$

For the clustering problem with n objects and given measurements we can distinguish $\mathcal{O}(k^n)$ such functions. In parameter estimation of parametric probability models we have to coarsen the continuous underlying space with regular grids or in a random fashion which yields $\mathcal{O}((\Omega/\epsilon)^p)$, $\Omega \subset \mathbb{R}$ different functions, p being the number of parameters.

3 Empirical Risk Approximation

Exploratory pattern analysis in combination with model selection requires to assess the quality of hypotheses $c \in \mathcal{C}$, that are assignments of data to patterns. We adopt a cost function (risk) viewpoint in this paper which attributes a non-negative cost value

$$R : \mathcal{C} \times \mathcal{X} \rightarrow \mathbb{R}_+, \quad (c, \mathbf{X}) \mapsto R(c, \mathbf{X}) \quad (3)$$

to each hypothesis ($\mathbb{R}_+ := [0, \infty)$).

The classical theory of statistical learning [98] advocates to use the empirical minimizer as the solution of the inference problem. The best empirical pattern $c^\perp(\mathbf{X})$ minimizes the empirical risk (ERM) of the pattern analysis problem given the measurements $\mathbf{X}$, i.e.,

$$c^\perp(\mathbf{X}) \in \arg \min_c R(c, \mathbf{X}). \quad (4)$$

The ERM theory requires for learnability of classifications that the hypothesis class is not too complex (i.e., finite VC-dimension) and, as a consequence, the ERM solution $c^\perp(\mathbf{X})$ converges to the optimal solution which minimizes the expected risk. A corresponding criterion has been derived for regression [1].

This classical learning theory is not applicable when the size of the hypothesis class grows with the number of objects like in clustering or other optimization problems with a combinatorial flavor. Without strong regularization we cannot hope to identify a single solution which globally minimizes the expected risk in the asymptotic limit. Therefore, we replace the concept of a unique function as the solution for the learning problem with a weighted set of functions. The weights are defined as

$$w : \mathcal{C} \times \mathcal{X} \times \mathbb{R}_+ \rightarrow [0, 1], \quad (c, \mathbf{X}, \beta) \mapsto w_\beta(c, \mathbf{X}). \quad (5)$$

The set of weights is denoted as $\mathcal{W}_\beta(\mathbf{X}) = \{w_\beta(c, \mathbf{X}) : c \in \mathcal{C}\}$.

How should we choose the weights $w_\beta(c, \mathbf{X})$ that large weights are only assigned to functions with low costs? The partial ordering constraint

$$R(c, \mathbf{X}) \leq R(\tilde{c}, \mathbf{X}) \Leftrightarrow w_\beta(c, \mathbf{X}) \geq w_\beta(\tilde{c}, \mathbf{X}), \quad (6)$$

ensures that functions with minimal costs $R(c^\perp, \mathbf{X})$ assume the maximal weight value which is normalized to one w.l.o.g., i.e., $0 \leq w_\beta(c, \mathbf{X}) \leq 1$. The non-negativity constraint of weights allows us to write the weights as $w_\beta(c, \mathbf{X}) = \exp(-\beta f(R(c, \mathbf{X})))$ with the monotonic function $f(x)$. Since $f(x)$ amounts to a monotone rescaling of the costs $R(c, \mathbf{X})$ we resort w.l.o.g. to the common choice of Boltzmann weights with the inverse computational temperature β , i.e.,

$$w_\beta(c, \mathbf{X}) = \exp(-\beta R(c, \mathbf{X})) . \quad (7)$$

4 Generalization and the Two Instance Scenario

To determine the optimal regularization of a pattern recognition method we have to define and estimate the generalization performance of hypotheses. We adopt the two instance scenario with training and test data described by respective object sets $\mathbf{O}^{(1)}, \mathbf{O}^{(2)}$ and measurements $\mathbf{X}^{(1)}, \mathbf{X}^{(2)} \sim \mathbb{P}(\mathbf{X})$. Both sets of measurements are drawn i.i.d. from the same probability distribution $\mathbb{P}(\mathbf{X})$. Furthermore, $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$ uniquely identify the training and test object sets $\mathbf{O}^{(1)}, \mathbf{O}^{(2)}$ so that it is sufficient to list $\mathbf{X}^{(j)}$ as references to object sets $\mathbf{O}^{(j)}, j = 1, 2$. The training and test data $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$ define two optimization problems $R(., \mathbf{X}^{(1)}), R(., \mathbf{X}^{(2)})$. The two instance scenario or two sample set scenario is widely used in statistics and statistical learning theory [8], i.e., to bound the deviation of empirical risk from expected risk, but also for two-terminal systems in information theory [5].

Statistical pattern analysis requires that inferred patterns have to generalize from training data to test data since noise in the data might render the ERM solution $c^\perp(\mathbf{X}^{(1)}) \neq c^\perp(\mathbf{X}^{(2)})$ unstable. How can we evaluate the generalization properties of solutions to a pattern recognition problem? Before we can compute the costs $R(., \mathbf{X}^{(2)})$ on test data of approximate solutions $c(\mathbf{X}^{(1)}) \in \mathcal{C}_\gamma(\mathbf{X}^{(1)})$ on training data we have to identify a pattern $\tilde{c}(\mathbf{X}^{(2)}) \in \mathcal{C}(\mathbf{X}^{(2)})$ which corresponds to $c(\mathbf{X}^{(1)})$. A priori, it is not clear how to compare patterns $c(\mathbf{X}^{(1)})$ for measurements $\mathbf{X}^{(1)}$ with patterns $c(\mathbf{X}^{(2)})$ for measurements $\mathbf{X}^{(2)}$. Therefore, we define a bijective mapping

$$\psi : \mathcal{X}^{(1)} \rightarrow \mathcal{X}^{(2)}, \quad \mathbf{X}^{(1)} \mapsto \psi(\mathbf{X}^{(1)}) . \quad (8)$$

The mapping ψ allows us to identify a pattern hypothesis for training data $c \in \mathcal{C}(\mathbf{X}^{(1)})$ with a pattern hypothesis for test data $c \in \mathcal{C}(\psi(\mathbf{X}^{(2)}))$. The reader should note that such a mapping ψ might change the object indices. In cases when the measurements are elements of an underlying metric space, then a natural choice for ψ is the nearest neighbor mapping.

The mapping ψ enables us to evaluate pattern costs on test data $\mathbf{X}^{(2)}$ for patterns $c(\mathbf{X}^{(1)})$ selected on the basis of training data $\mathbf{X}^{(1)}$. Consequently, we can determine how many training patterns with large weights share also large weights on test data, i.e.,

$$\Delta Z_\beta(\mathbf{X}^{(1)}, \mathbf{X}^{(2)}) := \sum_{c \in \mathcal{C}(\mathbf{X}^{(2)})} w_\beta(c, \psi(\mathbf{X}^{(1)})) w_\beta(c, \mathbf{X}^{(2)}) . \quad (9)$$

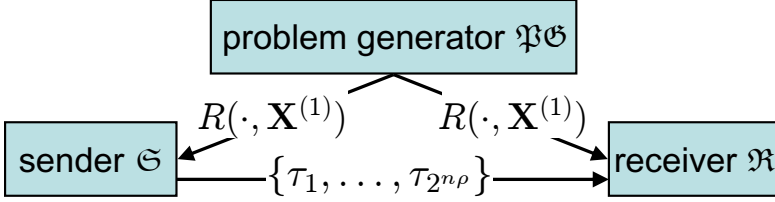


Fig. 1. Generation of a set of $2^{n\rho}$ code problems by e.g. permuting the object indices

A large subset of hypotheses with jointly large weights indicates that low cost hypotheses on training data $\mathbf{X}^{(1)}$ also perform with low costs on test data. The tradeoff between stability and informativeness for Boltzmann weights (7) is controlled by maximizing β under the constraint of large weight overlap $\Delta Z_\beta(\mathbf{X}^{(1)}, \mathbf{X}^{(2)}) / \sum_c w_\beta(c, \mathbf{X}^{(2)}) \approx 1$ for given risk function $R(\cdot, \mathbf{X})$.

5 Coding by Approximation

In the following, we describe a communication scenario with a sender $\mathfrak{S}$, a receiver $\mathfrak{R}$ and a problem generator $\mathfrak{PG}$. The problem generator serves as a noisy channel between sender and receiver. Communication takes place by approximately optimizing cost functions, i.e., by calculating weight sets $Z_\beta(\mathbf{X}^{(1)})$, $Z_\beta(\mathbf{X}^{(2)})$. This coding concept will be referred to as weighted approximation set coding (ASC) since the weights are concentrated on approximate minimizers of the optimization problem. The noisy channel is characterized by a pattern cost function $R(c, \mathbf{X})$ which determines the channel capacity of the ASC scenario. Validation and selection of pattern recognition models is then achieved by maximizing the channel capacity over a set of cost functions $R_\theta(\cdot, \mathbf{X})$, $\theta \in \Theta$ where θ indexes the various pattern recognition objectives.

Before we describe the communication protocol we have to define the code for communication. The objective $R(c, \mathbf{X}^{(1)})$ with the training data $\mathbf{X}^{(1)}$ define the noisy channel. We interpret the set of weights $\mathcal{W}_\beta(\mathbf{X}^{(1)})$ as the message to be communicated between sender and receiver. Since the peak of the weight distribution identifies patterns with low costs, we have to generate a set of weight sets in an analogous way to Shannon's codebook. The two instance scenario, however, provides only one set of measurements $\mathbf{X}^{(1)}$, and consequently only one weight set. Therefore, we have to introduce a set of (random) equivariant transformations τ applied to the training data $\mathbf{X}^{(1)}$ such that the minimizer $c^\perp$ can be located at a respective (random) position in the hypothesis space. Formally, the equivariance condition states that

$$c(\tau \circ \mathbf{X}) = \tau \circ c(\mathbf{X}), \quad (10)$$

i.e., a hypothesis on transformed data is equivalent to a transformation of the hypothesis on the original data. Special cases of such transformations τ are random permutations when optimizing combinatorial optimization cost functions

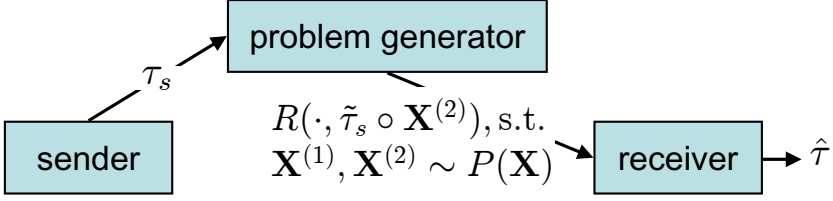


Fig. 2. Communication process: (1) the sender selects transformation τ_s , (2) the problem generator draws $\mathbf{X}^{(2)} \sim \mathbb{P}(\mathbf{X})$ and applies $\tilde{\tau}_s = \psi \circ \tau_s \circ \psi^{-1}$ to it, and the receiver estimates $\hat{\tau}$ based on $\tilde{\mathbf{X}} = \tilde{\tau}_s(\mathbf{X}^{(2)})$

like clustering models or graph cut problems. In parametric statistics, the transformations are parameter grids of e.g. rotations when estimating the orthogonal transformations of PCA or SVD. A possibly exponentially large set of transformations $\mathcal{T} = \{\tau_j : 1 \leq j \leq 2^{n\rho}\}$ then serves as the code in this communication process with a rate ρ . The set of transformations gives also rise to a set of equivalent optimization cost functions $R(c, \tau_1 \circ \mathbf{X}^{(1)}), \dots, R(c, \tau_{2^{n\rho}} \circ \mathbf{X}^{(1)})$. It is important to note that we do not change the measurement values in this construction but their relation to the hypothesis class.

Sender $\mathfrak{S}$ and receiver $\mathfrak{R}$ agree on a cost function for pattern recognition $R(c, \mathbf{X}^{(1)})$ and on a mapping function ψ . The following procedure is then employed to generate the code for the communication process:

1. Sender $\mathfrak{S}$ and receiver $\mathfrak{R}$ obtain data $\mathbf{X}^{(1)}$ from the problem generator $\mathfrak{PG}$.
2. $\mathfrak{S}$ and $\mathfrak{R}$ calculate the weight set $\mathcal{W}_\beta(\mathbf{X}^{(1)})$.
3. $\mathfrak{S}$ generates a set of (random) transformations $\mathcal{T} := \{\tau_1, \dots, \tau_{2^{n\rho}}\}$. The transformations define a set of optimization problems $R(c, \tau_j(\mathbf{X}^{(1)}))$, $1 \leq j \leq 2^{n\rho}$ to determine weight sets $\mathcal{W}_\beta(\tau_j(\mathbf{X}^{(1)}))$, $1 \leq j \leq 2^{n\rho}$.
4. $\mathfrak{S}$ sends the set of transformations $\mathcal{T}$ to $\mathfrak{R}$ who determines the set of weight sets $\{\mathcal{W}_\beta(\tau_j(\mathbf{X}^{(1)}))\}_{j=1}^{2^{n\rho}}$.

The rationale behind this procedure is the following: Given the measurements $\mathbf{X}^{(1)}$ the sender has randomly covered the hypothesis class $\mathcal{C}(\mathbf{X}^{(1)})$ by respective weight sets $\{\mathcal{W}_\beta(\tau_j(\mathbf{X}^{(1)})) : 1 \leq j \leq 2^{n\rho}\}$. Communication can take place if the weight sets are stable under the stochastic fluctuations of the measurements. The criterion for reliable communication is defined by the ability of the receiver to identify the transformation which has been selected by the sender. After this setup procedure, both sender and receiver have a list of weight sets available.

How is the communication between sender and receiver organized? During communication, the following steps take place as depicted in fig. 2:

1. The sender $\mathfrak{S}$ selects a transformation τ_s as message and send it to the problem generator $\mathfrak{PG}$.
2. $\mathfrak{PG}$ generates a new data set $\mathbf{X}^{(2)}$, establishes correspondence ψ between $\mathbf{X}^{(1)}$ and $\mathbf{X}^{(2)}$. $\mathfrak{PG}$ then applies the selected transformation τ_s , yielding $\tilde{\mathbf{X}} = \psi \circ \tau_s \circ \psi^{-1}(\mathbf{X}^{(2)})$.

3. $\mathfrak{P}\mathfrak{G}$ send $\tilde{\mathbf{X}}$ to the receiver $\mathfrak{R}$ without revealing τ_s .
4. $\mathfrak{R}$ calculates the weight set $\mathcal{W}_\beta(\tilde{\mathbf{X}})$.
5. $\mathfrak{R}$ estimates the selected transformation τ_s by using the decoding rule

$$\hat{\tau} = \arg \max_{\tau \in \mathcal{T}} \sum_{c \in \mathcal{C}(\mathbf{X}^{(2)})} w_\beta(c, \psi \circ \tau(\mathbf{X}^{(1)})) w_\beta(c, \tilde{\mathbf{X}}). \quad (11)$$

In the case of discrete hypothesis classes, then the communication channel is bounded from above by the cardinality of $\mathcal{C}(\mathbf{X})$ if two conditions hold: (i) the channel is noise free $\mathbf{X}^{(1)} \equiv \mathbf{X}^{(2)}$; (ii) the transformation set is sufficiently rich that every hypothesis can be selected as a global minimizer of the cost function.

6 Error Analysis of Approximation Set Coding

To determine the optimal approximation precision for an optimization problem $R(\cdot, \mathbf{X})$ we have to derive necessary and sufficient conditions which have to hold in order to reliably identify the transformations $\tau_s \in \mathcal{T}$. The parameter β , which controls the concentration of weights and thereby the resolution of the hypothesis class, has to be adapted to the size of the transformation set $|\mathcal{T}|$. Therefore, we analyse the error probability of the decoding rule (11) which is associated with a particular cost function $R(\cdot, \mathbf{X})$ and a rate ρ . The maximal value of β under the condition of zero error communication is defined as *approximation capacity* since it determines the approximation precision of the coding scheme.

A communication error occurs if the sender selects τ_s and the receiver decodes $\hat{\tau} = \tau_j, j \neq s$. To estimate the probability of this event, we introduce the weight overlaps

$$\Delta Z_\beta^j := \sum_{c \in \mathcal{C}(\mathbf{X}^{(2)})} w_\beta(c, \psi \circ \tau_j(\mathbf{X}^{(1)})) w_\beta(c, \tilde{\mathbf{X}}), \quad \tau_j \in \mathcal{T}. \quad (12)$$

The number ΔZ_β^j measures the number of hypotheses which have low costs $R(c, \psi \circ \tau_j(\mathbf{X}^{(1)}))$ and $R(c, \tilde{\mathbf{X}})$.

The probability of a communication error is given by a substantial overlap ΔZ_β^j induced by $\tau_j \in \mathcal{T} \setminus \{\tau_s\}, 1 \leq j \leq 2^{n\rho}$, i.e.,

$$\begin{aligned} \mathbb{P}(\hat{\tau} \neq \tau_s | \tau_s) &= \mathbb{P} \left(\max_{1 \leq j \leq 2^{n\rho}} \Delta Z_\beta^j \geq \Delta Z_\beta^s \middle| \tau_s \right) \\ &= \mathbb{E}_{\mathbf{X}^{(1,2)}} \mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}} \mathbb{I} \{ \max_{j \neq s} \Delta Z_\beta^j \geq \Delta Z_\beta^s \} \end{aligned} \quad (13)$$

with the indicator function $\mathbb{I}\{expr\} = \begin{cases} 1 & expr \text{ is true,} \\ 0 & \text{otherwise.} \end{cases}$. The notation $\mathbf{X}^{(1,2)} =$

$(\mathbf{X}^{(1)}, \mathbf{X}^{(2)})$ denotes the expectation w.r.t. both training data $\mathbf{X}^{(1)}$ and test data $\mathbf{X}^{(2)}$. The expectation $\mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}}$ is calculated w.r.t. the set of random transformations $\tau_j, 1 \leq j \leq 2^{n\rho}, j \neq s$ where we have conditioned on the sender

selected transformation τ_s . The joint probability distribution of all transformations $\mathbb{P}(\mathcal{T}) = \prod_{j=1}^{2^{n\rho}} \mathbb{P}(\tau_j)$ decomposes into product form since all transformations are randomly drawn from the set of all possible transformations $\{\tau_j\}$. It corresponds to the Shannon's random codebook design in information theory.

The error probability (13) can now be approximated by deriving an upper bound for the indicator function and, in a second step, by using the union bound for the maximum. The indicator function is bounded by $\mathbb{I}\{x \geq a\} \leq \frac{x}{a}$ for all $x \geq 0$.

The confusion probability with any other message $\tau_j, j \neq s$ for given training data $\mathbf{X}^{(1)}$ and test data $\mathbf{X}^{(2)}$ conditioned on τ_s is defined by

$$\begin{aligned} \mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}} \mathbb{I}\{\max_{j \neq s} \Delta Z_{\beta}^j \geq \Delta Z_{\beta}^s\} &\stackrel{(a)}{\leq} \mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}} \frac{\max_{j \neq s} \Delta Z_{\beta}^j}{\Delta Z_{\beta}^s} \stackrel{(b)}{\leq} \mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}} \sum_{\tau_j \neq \tau_s} \frac{\Delta Z_{\beta}^j}{\Delta Z_{\beta}^s} \\ &= \sum_{\tau_j \neq \tau_s} \frac{1}{|\{\tau_j\}|} \sum_{\{\tau_j\}} \frac{\Delta Z_{\beta}^j}{\Delta Z_{\beta}^s} \stackrel{(c)}{=} (2^{n\rho} - 1) \frac{Z_{\beta}^{(1)} Z_{\beta}^{(2)}}{|\{\tau_s\}| \Delta Z_{\beta}^s} \\ &\stackrel{(d)}{\leq} 2^{n\rho} \exp(-n\mathcal{I}_{\beta}(\tau_s, \hat{\tau})) \end{aligned} \quad (14)$$

with $Z_{\beta}^{(1,2)} := Z_{\beta}(\mathbf{X}^{(1,2)}) = \sum_{c \in \mathcal{C}(\mathbf{X}^{(1,2)})} w_{\beta}(c, \mathbf{X}^{(1,2)})$. In derivation (14) the expectation $\mathbb{E}_{\mathcal{T} \setminus \{\tau_s\}} [\mathbb{I}\{\Delta Z_{\beta}^j \geq \Delta Z_{\beta}^s\}]$ is conditioned on τ_s which has been omitted to increase the readability of the formulas. The summation $\sum_{\{\tau_j\}}$ is indexed by all possible realizations of the transformation τ_j that are uniformly selected. The first inequality (a) bounds the indicator function from above, the second inequality is due to the union bound of the maximum operator. Averaging over a random transformation τ_j (c) breaks any statistical dependency between sender and receiver weight sets which corresponds to the error case in jointly typical coding [4]; the number of possible transformations $|\{\tau_j\}|$ is identified with $|\{\tau_s\}|$ since all transformations have been chosen i.i.d. (d) We have introduced the mutual information between the sender message τ_s and the receiver message $\hat{\tau}$

$$\mathcal{I}_{\beta}(\tau_s, \hat{\tau}) = \frac{1}{n} \log \left(\frac{|\{\tau_s\}| \Delta Z_{\beta}^s}{Z_{\beta}^{(1)} Z_{\beta}^{(2)}} \right) = \frac{1}{n} \left(\log \frac{|\{\tau_s\}|}{Z_{\beta}^{(1)}} + \log \frac{|\mathcal{C}^{(2)}|}{Z_{\beta}^{(2)}} - \log \frac{|\mathcal{C}^{(2)}|}{\Delta Z_{\beta}^s} \right). \quad (15)$$

Inequality in step (d) results from dropping the correction -1 .

The interpretation of eq. (15) is straightforward: The first logarithm measures the entropy of the number of transformations which can be resolved up to a minimal uncertainty encoded by the sum of the approximation weights $Z_{\beta}^{(1)}$ in the space of clusterings on the sender side. The logarithm $\log(|\mathcal{C}^{(2)}|/Z_{\beta}^{(2)})$ calculates the entropy of the receiver patterns which are quantized by $Z_{\beta}^{(2)}$. The third logarithm measures the joint entropy of $(\tau_s, \hat{\tau})$ which depends on the integrated weight product $\Delta Z_{\beta}^s = \sum_{c \in \mathcal{C}(\mathbf{X}^{(2)})} w_{\beta}(c, \psi \circ \tau_s \circ \mathbf{X}^{(1)}) w_{\beta}(c, \hat{\mathbf{X}})$. Inserting (14) into (13) yields the upper bound for the error probability

$$\mathbb{P}(\hat{\tau} \neq \tau_s | \tau_s) \leq \mathbb{E}_{\mathbf{X}^{(1,2)}} \exp(-n(\mathcal{I}_{\beta}(\tau_s, \hat{\tau}) - \rho \log 2)) \quad (16)$$

The communication rate $n\rho \log 2$ is limited by the mutual information $\mathcal{I}_\beta(\tau_s, \hat{\tau})$ for asymptotically error-free communication.

7 Information Theoretical Model Selection

The analysis of the error probability suggests the following inference principle for controlling the appropriate regularization strengths which implements a form of model selection: the approximation precision is controlled by β which has to be maximized to derive more precise solutions or patterns. For small β the rate ρ will be low since we resolve the space of solutions only in a coarse grained fashion. For too large β the error probability does not vanish which indicates confusions between $\tau_j, j \neq s$ and τ_s . The optimal β -value is given by the largest β or, equivalently the highest approximation precision

$$\beta^* = \arg \max_{\beta \in [0, \infty)} \mathcal{I}_\beta(\tau_s, \hat{\tau}). \quad (17)$$

Another choice to be made in modeling is to select a suitable cost function $R(\cdot, \mathbf{X})$ for the pattern recognition problems at hand. Let us assume that a number of cost functions $\{R_\theta(\cdot, \mathbf{X}), \theta \in \Theta\}$ are considered as candidates. The approximation capacity $\mathcal{I}_\beta(\tau_s, \hat{\tau})$ depends on the cost function through the Gibbs weights. Therefore, we can rank the different models according to their $\mathcal{I}_\beta(\tau_s, \hat{\tau})$ values. Robust and informative cost functions yield a higher approximation capacity than simpler or more brittle models. A rational choice is to select the cost function

$$R^*(c, \mathbf{X}) = \arg \max_{\theta \in \Theta} \mathcal{I}_\beta(\tau_s, \hat{\tau} | R_\theta) \quad (18)$$

where both the random variables τ_s and $\hat{\tau}$ depend on $R_\theta(c, \mathbf{X}), \theta \in \Theta$. The selection rule (18) prefers the model which is “expressive” enough to exhibit high information content (e.g., many clusters in clustering) and, at the same time robustly resists to noise in the data set. The bits or nats which are measured in the ASC communication setting are context sensitive since they refer to a hypothesis class $\mathcal{C}(\mathbf{X})$, i.e., how finely or coarsely functions can be resolved in $\mathcal{C}$.

8 Conclusion

Model selection and validation requires to estimate the generalization ability of models from training to test data. “Good” models show a high expressiveness and they are robust w.r.t. noise in the data. This tradeoff between *informativeness* and *robustness* ranks different models when they are tested on new data and it quantitatively describes the underfitting/overfitting dilemma. In this paper we have explored the idea to use weighted approximation sets of clustering solutions as a communication code. The *approximation capacity* of a cost function provides a selection criterion which renders various models comparable in terms of their respective bit rates. The number of reliably extractable bits of a pattern analysis cost function $R(\cdot, \mathbf{X})$ defines a “task sensitive information measure” since it only

accounts for the fluctuations in the data $\mathbf{X}$ which actually have an influence on identifying an individual pattern or a set of patterns.

The maximum entropy inference principle suggests that we should average over the statistically indistinguishable solutions in the optimal weighted approximation set. Such a model averaging strategy replaces the original cost function with the free energy and, thereby, it defines a continuation methods with maximal robustness. Algorithmically, maximum entropy inference can be implemented by annealing methods [7, 2, 6]. The urgent question in many data analysis applications, which regularization term should be used without introducing an unwanted bias, is naturally answered by the entropy. The second question, how the regularization parameter should be selected, is answered by ASC: Choose the parameter value which maximizes the approximation capacity!

ASC for model selection can be applied to all combinatorial or continuous optimization problems which depend on noisy data. The noise level is characterized by two sample sets $\mathbf{X}^{(1)}, \mathbf{X}^{(2)}$. Two samples provide far too little information to estimate the probability density of the measurements but two large samples contain sufficient information to determine the uncertainty in the solution space. The equivalence of ensemble averages and time averages of ergodic systems is heavily exploited in statistical mechanics and it also enables us in this paper to derive a model selection strategy based on two samples.

Acknowledgment. This work has been partially supported by the DFG-SNF research cluster FOR916 and by the FP7 EU project SIMBAD.

References

1. Alon, N., Ben-David, S., Cesa-Bianchi, N., Haussler, D.: Scale-sensitive dimensions, uniform convergence, and learnability. *J. ACM* 44(4), 615–631 (1997)
2. Buhmann, J.M., Kühnel, H.: Vector quantization with complexity costs. *IEEE Tr. Information Theory* 39(4), 1133–1145 (1993)
3. Buhmann, J.M.: Information theoretic model validation for clustering. In: *IEEE International Symposium on Information Theory*, Austin Texas. IEEE, New York (2010), <http://arxiv.org/abs/1006.0375>
4. Cover, T.M., Thomas, J.A.: *Elements of Information Theory*. John Wiley & Sons, New York (1991)
5. Csizár, I., Körner, J.: *Information Theory: Coding theorems for discrete memoryless systems*. Academic Press, New York (1981)
6. Hofmann, T., Buhmann, J.M.: Pairwise data clustering by deterministic annealing. *IEEE Tr. Pattern Analysis and Machine Intelligence* 19(1), 1–14 (1997)
7. Rose, K., Gurewitz, E., Fox, G.: Vector quantization by deterministic annealing. *IEEE Transactions on Information Theory* 38(4), 1249–1257 (1992)
8. Vapnik, V.N.: *Estimation of Dependences Based on Empirical Data*. Springer, Heidelberg (1982)
9. Vapnik, V.N., Chervonenkis, A.Y.: On the uniform convergence of relative frequencies of events to their probabilities. *Theory of Probability and its Applications* 16, 264–280 (1971)

Evolutionary Multi-Objective Optimization: Basic Concepts and Some Applications in Pattern Recognition

Carlos A. Coello Coello

CINVESTAV (Evolutionary Computation Group)

Departamento de Computación

Av. IPN No. 2508, Col. San Pedro Zacatenco

México, D.F. 07360, Mexico

cocoello@cs.cinvestav.mx

Abstract. This paper provides a brief introduction to the so-called multi-objective evolutionary algorithms, which are bio-inspired metaheuristics designed to deal with problems having two or more (normally conflicting) objectives. First, we provide some basic concepts related to multi-objective optimization and a brief review of approaches available in the specialized literature. Then, we provide a short review of applications of multi-objective evolutionary algorithms in pattern recognition. In the final part of the paper, we provide some possible paths for future research in this area, which are promising, from the author's perspective.

1 Introduction

In the real world, there are many problems in which it is desirable to optimize two or more objective functions at the same time. These are known as *multi-objective optimization problems* (MOPs), and their solution involves finding not one, but a set of solutions that represent the best possible trade-offs among the objective functions being optimized. Such trade-offs constitute the so-called *Pareto optimal set*, and their corresponding objective function values form the so-called *Pareto front*.

A number of mathematical programming techniques have been developed to solve MOPs [1]. However, they have several limitations, from which the most important are that they tend to be very susceptible to the specific features of the MOP being solved (e.g., the shape or continuity of the Pareto front), and that they normally generate a single solution per run. Such limitations have motivated the development of alternative approaches from which metaheuristics [2] have been, with no doubt, the most popular and effective choice available so far (see for example [3]). From the many metaheuristics in current use, Evolutionary

¹ A **metaheuristic** is a high level strategy for exploring search spaces by using different methods [2]. Metaheuristics have both a diversification (i.e., exploration of the search space) and an intensification (i.e., exploitation of the accumulated search experience) procedure.

Algorithms (EAs) are, clearly, the most popular in today's specialized literature. EAs are inspired on the "survival of the fittest" principle from Darwin's evolutionary theory [4], and simulate the evolutionary process in a computer, as a way to solve problems. EAs have become very popular as multi-objective optimizers because of their ease of use (and implementation) and generality (e.g., EAs are less sensitive than mathematical programming techniques to the initial points used for the search and to the specific features of a MOP). EAs have also an additional advantage: since they are population-based techniques, it is possible for them to manage a set of solutions at a time, instead of only one, as normally done by traditional mathematical programming techniques. This allows EAs to generate several elements from the Pareto optimal set in a single run.

The first Multi-Objective Evolutionary Algorithm (MOEA) was proposed in the mid-1980s by David Schaffer [5]. However, it was until the mid-1990s that MOEAs started to attract serious attention from researchers. Nowadays, it is possible to find applications of MOEAs in practically all domains [2].

The rest of this paper is organized as follows. In Section 2, we provide some basic multi-objective optimization concepts required to make this paper self-contained. An introduction to evolutionary algorithms is provided in Section 3. Section 4 contains a brief description of the main MOEAs in current use. In Section 5, a short review of some applications of MOEAs in three pattern recognition tasks (image segmentation, feature selection and classification) is provided. Section 6 indicates some potential paths for future research in this area. Finally, the main conclusions of this paper are provided in Section 7.

2 Basic Concepts

We are interested in solving problems of the type [3]:

$$\text{minimize } \mathbf{f}(\mathbf{x}) := [f_1(\mathbf{x}), f_2(\mathbf{x}), \dots, f_k(\mathbf{x})] \quad (1)$$

subject to:

$$g_i(\mathbf{x}) \leq 0 \quad i = 1, 2, \dots, m \quad (2)$$

$$h_i(\mathbf{x}) = 0 \quad i = 1, 2, \dots, p \quad (3)$$

where $\mathbf{x} = [x_1, x_2, \dots, x_n]^T$ is the vector of decision variables, $f_i : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, k$ are the objective functions and $g_i, h_j : \mathbb{R}^n \rightarrow \mathbb{R}$, $i = 1, \dots, m$, $j = 1, \dots, p$ are the constraint functions of the problem.

To describe the concept of optimality in which we are interested, we will introduce next a few definitions.

² The author maintains the EMOO repository, which currently contains over 5800 bibliographic references related to evolutionary multi-objective optimization. The EMOO repository is located at: <http://delta.cs.cinvestav.mx/~ccoello/EMOO/>

³ Without loss of generality, we will assume only minimization problems.

Definition 1. Given two vectors $\mathbf{x}, \mathbf{y} \in \mathbb{R}^k$, we say that $\mathbf{x} \leq \mathbf{y}$ if $x_i \leq y_i$ for $i = 1, \dots, k$, and that $\mathbf{x}$ **dominates** $\mathbf{y}$ (denoted by $\mathbf{x} \prec \mathbf{y}$) if $\mathbf{x} \leq \mathbf{y}$ and $\mathbf{x} \neq \mathbf{y}$.

Definition 2. We say that a vector of decision variables $\mathbf{x} \in \mathcal{X} \subset \mathbb{R}^n$ is **non-dominated** with respect to $\mathcal{X}$, if there does not exist another $\mathbf{x}' \in \mathcal{X}$ such that $\mathbf{f}(\mathbf{x}') \prec \mathbf{f}(\mathbf{x})$.

Definition 3. We say that a vector of decision variables $\mathbf{x}^* \in \mathcal{F} \subset \mathbb{R}^n$ ($\mathcal{F}$ is the feasible region) is **Pareto-optimal** if it is nondominated with respect to $\mathcal{F}$.

Definition 4. The **Pareto Optimal Set** $\mathcal{P}^*$ is defined by:

$$\mathcal{P}^* = \{\mathbf{x} \in \mathcal{F} | \mathbf{x} \text{ is Pareto-optimal}\}$$

Definition 5. The **Pareto Front** $\mathcal{PF}^*$ is defined by:

$$\mathcal{PF}^* = \{\mathbf{f}(\mathbf{x}) \in \mathbb{R}^k | \mathbf{x} \in \mathcal{P}^*\}$$

We thus wish to determine the Pareto optimal set from the set $\mathcal{F}$ of all the decision variable vectors that satisfy (2) and (3). Note however that in practice, not all the Pareto optimal set is normally desirable (e.g., it may not be desirable to have different solutions that map to the same values in objective function space) or achievable.

3 A Short Introduction to Evolutionary Algorithms

Although the origins of evolutionary algorithms (EAs) can be traced back to the early 1930s [6], it was until the 1960s that the three main types of EAs were developed: genetic algorithms [7], evolution strategies [8] and evolutionary programming [9].

EAs are very suitable for solving multi-objective optimization problems because they operate on a set of solutions (called *population*), which allows them to generate several elements of the Pareto optimal set in a single run (contrasting with mathematical programming techniques, which normally generate a single nondominated solution per execution). Additionally, EAs are less susceptible to the discontinuity and the shape of the Pareto front, which is another important advantage over traditional mathematical programming methods [3].

Multi-objective Evolutionary Algorithms (MOEAs) extend traditional EAs in two main aspects:

- **Selection Mechanism:** In MOEAs, the aim is to select nondominated solutions and not the solutions with the highest fitness. Additionally, and according to the definition of Pareto optimality, all the nondominated solutions in a population are normally considered as equally good.
- **Diversity Maintenance:** MOEAs require a mechanism that preserves diversity and avoids convergence to a single solution (this will eventually happen because of stochastic noise, if an EA is run for a sufficiently long time).

Regarding selection, several approaches have been adopted over the years, going from simple linear aggregating functions [10] and population-based schemes [5] to ranking approaches based on Pareto optimality [4,11,12] and schemes based on performance measures [13].

Diversity has also been a popular research topic, and a wide variety of methods are currently available to maintain diversity in the population of an EA, including fitness sharing and niching [14,15], clustering [16,17], geographically-based schemes [18], and the use of entropy [19,20], among others.

A third component of modern MOEAs is elitism, which normally consists of using an external archive (called a “secondary population”) that can interact (or not) in different ways with the main (or “primary”) population of a MOEA. Although the main goal of this archive is to store the nondominated solutions generated throughout the search, it has also been used to maintain diversity [21]. The approximation of the Pareto optimal set produced by a MOEA can be found in the final contents of this external archive and this is normally the result reported as the outcome of a MOEA’s execution.

4 Multi-Objective Evolutionary Algorithms

Although there is a wide variety of MOEAs available in the specialized literature, only a handful of them are in wide use. The following are, in the view of the author, the most representative MOEAs in current use:

1. **Strength Pareto Evolutionary Algorithm (SPEA):** It was conceived as an elegant merge of several MOEAs that were developed during the mid-1990s [17]. Its main features are an external archive (called the external nondominated set), which stores the nondominated solutions generated during the search. The union of both the external nondominated set and the main population participate in the selection process, during which a *strength* value is computed for each individual. This strength is proportional to the number of solutions that a certain individual dominates. Then, the fitness of each member of the current population is computed according to the strengths of all the external nondominated solutions that dominate it. The external nondominated set can significantly grow in size, consequently reducing the selection pressure, and slowing down the search. Since this is undesirable, a clustering technique is adopted to prune the contents of the external nondominated set so that its size remains bounded within a certain (user-defined) threshold.
2. **Pareto Archived Evolution Strategy (PAES):** This approach was introduced in 2000 [21] and is probably the most simple MOEA than can be conceived. It consists of a (1+1) evolution strategy (i.e., a single parent that generates a single offspring) in combination with an external archive that stores the nondominated solutions found so far. This archive is used as a reference set against which each mutated individual is being compared. Such an

(external) archive adopts a procedure that divides objective function space in a recursive manner. Then, each solution is placed in a certain grid location based on the values of its objectives (which are used as its “geographical location”). A map of such a grid is maintained, indicating the number of solutions that reside in each grid location. When a new nondominated solution is ready to be stored in the archive, but there is no room for it (this is because the size of the external archive is bounded), a check is made on the grid location to which the solution would belong. If this grid location is less densely populated than the most densely populated grid location, then a solution (randomly chosen) from the most populated grid location is deleted in order to allow the storage of the new solution. This aims to redistribute solutions, favoring the less densely populated regions of the Pareto front.

3. **Strength Pareto Evolutionary Algorithm 2 (SPEA2)**: This is a revised version of SPEA, which has three main differences with respect to it [22]: (1) it incorporates a fine-grained fitness assignment strategy which takes into account, for each individual, the number of individuals that dominate it and the number of individuals by which it is dominated; (2) it uses a nearest neighbor density estimation technique which guides the search more efficiently, and (3) it has an enhanced archive truncation method that guarantees the preservation of boundary solutions.
4. **Nondominated Sorting Genetic Algorithm II (NSGA-II)**: This approach is a revised version of one of the earliest MOEAs, called the Nondominated Sorting Genetic Algorithm (NSGA), which was originally introduced in the mid 1990s [11]. The NSGA-II adopts a more efficient ranking procedure than its predecessor. Additionally, it estimates the density of solutions surrounding a particular individual in the population by computing the average distance of two points on either side of this point along each of the objectives of the problem. This value is the so-called *crowding distance*. During selection, the NSGA-II uses a crowded-comparison operator which takes into consideration both the nondomination rank of an individual in the population and its crowding distance (i.e., nondominated solutions are preferred over dominated solutions, but between two solutions with the same nondomination rank, the one that resides in the less crowded region is preferred). The NSGA-II does not use an external archive as most of the modern MOEAs in current use. Instead, the elitist mechanism of the NSGA-II consists of combining the best parents with the best offspring obtained (i.e., a $(\mu + \lambda)$ -selection). Due to its clever mechanisms, the NSGA-II is much more efficient (computationally speaking) than its predecessor, and its performance is so good that it has become very popular in the last few years, triggering a significant number of applications, and becoming some sort of landmark against which new MOEAs have to be compared in order to merit publication.

5. **Multiobjective Evolutionary Algorithm Based on Decomposition** (MOEA/D): This approach was introduced in 2007 [23]. It decomposes a problem into a number of scalar optimization sub-problems which are simultaneously optimized. When optimizing each subproblem, only information from its neighboring sub-problems is adopted. This allows this approach to be both very efficient and very effective. MOEA/D is a good example of a MOEA that was able to successfully incorporate concepts from mathematical programming (scalarization functions in this case) into a metaheuristic. Although not as popular as the NSGA-II, MOEA/D has attracted a lot of interest over the years, mainly because of its reputation as a hard-to-defeat MOEA.

Many other MOEAs are currently available (see for example [24,25]), but none of them is widely used in the literature. This, however, has not discouraged algorithm developers who are have now focused their efforts on aspects such as computational efficiency [26] and scalability [13].

5 Some Applications in Pattern Recognition

Once can find today a wide variety of applications of MOEAs in pattern recognition. For illustrative purposes only, three types of common applications will be briefly described next:

1. **Image Segmentation:** We call “segmentation” to the clustering of the pixels of an image based on certain criteria. The output of a segmentation process is usually another image with raw pixel data, which constitutes either the boundary of a region or all the points in the region itself. Segmentation is known to be, in general, a very difficult task. Treated as a multi-objective optimization problem, image segmentation can involve a number of objectives [27]. For example, Bhanu and Lee [28] considered five objectives when applying a genetic algorithm with a linear aggregating function to an image segmentation problem: (1) edge-border coincidence, (2) boundary consistency, (3) pixel classification, (4) object overlap, and (5) object contrast. Shirakawa and Nagao [29], however, considered the minimization of only two objectives (they adopted SPEA2 [22] in this case): (1) overall deviation between the data items and their corresponding cluster center (minimizing this objective increases the number of clusters) and (2) edge value, which evaluates the overall summed distances on boundaries between the regions. However, regardless of the objectives adopted, a good motivation for using MOEAs in image segmentation is that they allow the generation of several output images, representing the different trade-offs among the objectives. This gives the user more options to choose from, instead of the single image that is traditionally obtained when using single-objective optimization techniques.
2. **Feature Selection:** Feature selection refers to the extraction of features for differentiating one class of objects from another. The output of this process

is a vector of values of the measured features. In this case, when feature selection is treated as a multi-objective problem, the two most common objectives are: (1) the minimization of the number of features and (2) the minimization of the error associated with the solution obtained. For example, Hamdani et al. [30] considered as their second objective the classification error of a nearest neighbor (1-NN) classifier [31] in an application in which the NSGA-II [32] was used as their search engine. Similarly, Morita et al. [33] adopted as their second objective a validity index that measures the quality of the clusters formed. In this case, the authors were dealing with a handwritten word recognition task in which the original NSGA [11] was used as their search engine. It is also possible to introduce additional objectives related, for example, to cost [34] or some problem-specific characteristics [35]. This illustrates the flexibility that MOEAs provide when applied to pattern recognition tasks.

3. **Classification:** This is a process in which each input value is placed into a class (from several available) based on the information provided by its descriptors. When treated as a multi-objective problem, classification normally involves objectives such as minimizing the complexity of the classifier (e.g., its number of rules) while maximizing its accuracy (i.e., the error of the classifier). However, other objectives such as the generality of the rules, their understandability or their complexity can also be adopted. For example, Iglesia et al. [36,37] maximized confidence and coverage of the rules in a partial classification problem (the so-called *nugget discovery task*), in which the NSGA-II [32] was adopted as the search engine. In contrast, Bandyopadhyay et al. [38] considered three objectives: minimize (1) the number of misclassified training points and (2) the number of hyperplanes, and maximize (3) the product classwise correct recognition rates. The search engine adopted in this case, was as an approach introduced by the authors, and called the constrained elitist multiobjective genetic algorithm based classifier (CEMOGA-Classifer). One of the advantages of the use of MOEAs in classification is that they can overcome problems commonly associated to traditional (i.e., single-objective) classifiers, such as overfitting/overlearning and ignoring smaller classes.

6 Potential Areas for Further Research

As we have seen, MOEAs have been applied to several problems in pattern recognition. However, there are other possible paths for future research that may be worth exploring. For example:

- **Integration:** The development of fully automated pattern recognition systems should be a long-term goal related to the use of MOEAs in this area. Such systems should be applicable to different databases with minimum (or no) human intervention. The development of such systems could require the hybridization of MOEAs with other techniques (e.g., fuzzy logic and/or machine learning techniques) as well as the design of new architectures that

allow an efficient and effective integration of different types of approaches throughout the different stages involved in a pattern recognition process (see for example [39], in which an automatic image pattern recognition system based on genetic programming [40] is proposed for the classification of medical images). However, MOEAs are an excellent choice for this sort of task because of their capability to deal with conflicting objectives.

- **Efficiency:** MOEAs are certainly powerful optimization tools, but they normally have a relatively high computational cost because of the number of objective function evaluations that they require in order to produce reasonably good results. This is particularly important in tasks such as image segmentation in which each objective function evaluation will be normally costly. In order to deal with this problem, it is possible to adopt approaches such as fitness approximation [41], parallelization [42], or surrogate methods [43]. It is worth mentioning, however, that the incorporation of such techniques in pattern recognition tasks, although promising, is still scarce.
- **Use of other Metaheuristics:** A number of other bio-inspired metaheuristics have become increasingly popular in the last few years [44], and most of them have been applied to pattern recognition tasks, although their use has been fairly limited until now. The following are representative examples of these new metaheuristics:
 - **Particle Swarm Optimization (PSO):** It was proposed in the mid-1990s [45] and simulates the movements of a flock of birds which seek food. In PSO, the behavior of each individual (or *particle*) is affected by either the best local (i.e., within a certain neighborhood) or the best global (i.e., with respect to the entire population or *swarm*) individual. Although this approach also adopts a population and a fitness measure, unlike EAs, it allows individuals to benefit from their past experiences. PSO has been used in some pattern recognition tasks (see for example [46]), but not much in a multi-objective form.
 - **Artificial Immune Systems (AIS):** From a computational point of view, our immune system can be seen as a highly parallel intelligent system that is able to learn and retrieve previous knowledge (in other words, it has “memory”), while solving highly complex recognition and classification tasks. These interesting features motivated the development of the so-called artificial immune systems in the early 1990s [47,48]. This sort of approach has been used in a wide variety of tasks, including classification and pattern recognition in general (see for example [49]). However, as in the previous case, the use of AISs as multi-objective solvers of pattern recognition problems is still rare.
 - **Ant Colony Optimization (ACO):** It is inspired on the behavior shown by colonies of real ants which deposit a chemical substance on the

ground called *pheromone* [50]. The pheromone influences the behavior of the ants: they tend to take those paths in which there is a larger amount of pheromone. Therefore, pheromone trails can be seen as an indirect communication mechanism used by the ants. This system also presents several interesting features from a computational point of view, and has triggered a significant amount of research. The first metaheuristic inspired on this notion was called *ant system* and was originally proposed for the traveling salesman problem. Over the years, this approach (and its several variations, which are now collectively denominated *ant colony optimization* or ACO algorithms) has been applied to a wide variety of combinatorial optimization problems, including some pattern recognition tasks (see for example [51]). Nevertheless, its use in multi-objective pattern recognition tasks is very scarce.

7 Conclusions

This paper has provided a general overview of multi-objective evolutionary algorithms and some of their possible applications in pattern recognition. In order to make the paper self-contained, a short introduction to evolutionary algorithms has also been provided. After that, the main components that distinguish MOEAs from EAs were discussed, and the main MOEAs in current use were briefly described. In the final part of the paper, some possible paths for future research in this area were discussed. The main aim of this paper is to motivate the development of more research on the use of MOEAs (or any other type of multi-objective metaheuristic) for the solution pattern recognition problems.

Acknowledgements

The author acknowledges support from CONACyT through project 103570.

References

1. Miettinen, K.M.: Nonlinear Multiobjective Optimization. Kluwer Academic Publishers, Boston (1999)
2. Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Computing Surveys* 35(3), 268–308 (2003)
3. Coello Coello, C.A., Lamont, G.B., Van Veldhuizen, D.A.: *Evolutionary Algorithms for Solving Multi-Objective Problems*, 2nd edn. Springer, New York (2007), ISBN 978-0-387-33254-3
4. Goldberg, D.E.: *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison-Wesley Publishing Company, Reading (1989)
5. Schaffer, J.D.: Multiple Objective Optimization with Vector Evaluated Genetic Algorithms. In: *Genetic Algorithms and their Applications: Proceedings of the First International Conference on Genetic Algorithms*, pp. 93–100. Lawrence Erlbaum, Mahwah (1985)

6. Fogel, D.B.: Evolutionary Computation. Toward a New Philosophy of Machine Intelligence. The Institute of Electrical and Electronic Engineers, New York (1995)
7. Holland, J.H.: Concerning efficient adaptive systems. In: Yovits, M.C., Jacobi, G.T., Goldstein, G.D. (eds.) *Self-Organizing Systems*, pp. 215–230. Spartan Books, Washington, DC (1962)
8. Schwefel, H.P.: Kybernetische evolution als strategie der experimentellen forschung in der strömungstechnik. Dipl.-Ing. thesis (1965) (in German)
9. Fogel, L.J.: Artificial Intelligence through Simulated Evolution. John Wiley, New York (1966)
10. Hajela, P., Lin, C.Y.: Genetic search strategies in multicriterion optimal design. *Structural Optimization* 4, 99–107 (1992)
11. Srinivas, N., Deb, K.: Multiobjective Optimization Using Nondominated Sorting in Genetic Algorithms. *Evolutionary Computation* 2(3), 221–248 (1994)
12. Fonseca, C.M., Fleming, P.J.: Genetic Algorithms for Multiobjective Optimization: Formulation, Discussion and Generalization. In: Forrest, S. (ed.) *Proceedings of the Fifth International Conference on Genetic Algorithms*, University of Illinois at Urbana-Champaign, San Mateo, California, pp. 416–423. Morgan Kaufman Publishers, San Francisco (1993)
13. Beume, N., Naujoks, B., Emmerich, M.: SMS-EMOA: Multiobjective selection based on dominated hypervolume. *European Journal of Operational Research* 181(3), 1653–1669 (2007)
14. Goldberg, D.E., Richardson, J.: Genetic algorithm with sharing for multimodal function optimization. In: Grefenstette, J.J. (ed.) *Genetic Algorithms and Their Applications: Proceedings of the Second International Conference on Genetic Algorithms*, pp. 41–49. Lawrence Erlbaum, Hillsdale (1987)
15. Deb, K., Goldberg, D.E.: An Investigation of Niche and Species Formation in Genetic Function Optimization. In: Schaffer, J.D. (ed.) *Proceedings of the Third International Conference on Genetic Algorithms*, George Mason University, San Mateo, California, June 1989, pp. 42–50. Morgan Kaufmann, San Francisco (1989)
16. Toscano Pulido, G., Coello Coello, C.A.: Using clustering techniques to improve the performance of a multi-objective particle swarm optimizer. In: Deb, K., et al. (eds.) *GECCO 2004. LNCS*, vol. 3102, pp. 225–237. Springer, Heidelberg (2004)
17. Zitzler, E., Deb, K., Thiele, L.: Comparison of Multiobjective Evolutionary Algorithms on Test Functions of Different Difficulty. In: Wu, A.S. (ed.) *Proceedings of the 1999 Genetic and Evolutionary Computation Conference, Workshop Program*, Orlando, Florida, July 1999, pp. 121–122 (1999)
18. Knowles, J., Corne, D.: Properties of an Adaptive Archiving Algorithm for Storing Nondominated Vectors. *IEEE Transactions on Evolutionary Computation* 7(2), 100–116 (2003)
19. Kita, H., Yabumoto, Y., Mori, N., Nishikawa, Y.: Multi-Objective Optimization by Means of the Thermodynamical Genetic Algorithm. In: Ebeling, W., Rechenberg, I., Voigt, H.-M., Schwefel, H.-P. (eds.) *PPSN 1996. LNCS*, vol. 1141, pp. 504–512. Springer, Heidelberg (1996)
20. Cui, X., Li, M., Fang, T.: Study of Population Diversity of Multiobjective Evolutionary Algorithm Based on Immune and Entropy Principles. In: *Proceedings of the Congress on Evolutionary Computation 2001 (CEC 2001)*, May 2001, vol. 2, pp. 1316–1321. IEEE Service Center, Piscataway (2001)
21. Knowles, J.D., Corne, D.W.: Approximating the Nondominated Front Using the Pareto Archived Evolution Strategy. *Evolutionary Computation* 8(2), 149–172 (2000)

22. Zitzler, E., Laumanns, M., Thiele, L.: SPEA2: Improving the Strength Pareto Evolutionary Algorithm. In: Giannakoglou, K., Tsahalidis, D., Periaux, J., Papailou, P., Fogarty, T. (eds.) EUROGEN 2001 Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems, Athens, Greece, pp. 95–100 (2001)
23. Zhang, Q., Li, H.: MOEA/D: A Multiobjective Evolutionary Algorithm Based on Decomposition. *IEEE Transactions on Evolutionary Computation* 11(6), 712–731 (2007)
24. Deb, K., Mohan, M., Mishra, S.: Evaluating the ϵ -Domination Based Multi-Objective Evolutionary Algorithm for a Quick Computation of Pareto-Optimal Solutions. *Evolutionary Computation* 13(4), 501–525 (2005)
25. Toscano Pulido, G., Coello Coello, C.A.: The micro genetic algorithm 2: Towards online adaptation in evolutionary multiobjective optimization. In: Fonseca, C.M., Fleming, P.J., Zitzler, E., Deb, K., Thiele, L. (eds.) EMO 2003. LNCS, vol. 2632, pp. 252–266. Springer, Heidelberg (2003)
26. Knowles, J.: ParEGO: A Hybrid Algorithm With On-Line Landscape Approximation for Expensive Multiobjective Optimization Problems. *IEEE Transactions on Evolutionary Computation* 10(1), 50–66 (2006)
27. Chin-Wei, B., Rajeswari, M.: Multiobjective Optimization Approaches in Image Segmentation—The Directions and Challenges. In: International on Advances in Soft Computing and its Applications, March 2010, vol. 2(1), pp. 40–65 (2010)
28. Bhanu, B., Lee, S.: Genetic Learning for Adaptive Image Segmentation. Kluwer Academic Publishers, Boston (1994)
29. Shirakawa, S., Nagao, T.: Evolutionary Image Segmentation Based on Multiobjective Clustering. In: 2009 IEEE Congress on Evolutionary Computation (CEC 2009), Trondheim, Norway, May 2009, pp. 2466–2473. IEEE Press, Los Alamitos (2009)
30. Hamdani, T.M., Won, J.-M., Alimi, M.A.M., Karray, F.: Multi-objective feature selection with NSGA II. In: Beliczynski, B., Dzielinski, A., Iwanowski, M., Ribeiro, B. (eds.) ICANNGA 2007. LNCS, vol. 4431, pp. 240–247. Springer, Heidelberg (2007)
31. Dasarathy, B.V.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1990)
32. Deb, K., Pratap, A., Agarwal, S., Meyarivan, T.: A Fast and Elitist Multiobjective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6(2), 182–197 (2002)
33. Morita, M., Sabourin, R., Bortolozzi, F., Suen, C.: Unsupervised Feature Selection Using Multi-Objective Genetic Algorithm for Handwritten Word Recognition. In: Proceedings of the 7th International Conference on Document Analysis and Recognition (ICDAR 2003), Edinburgh, Scotland, August 2003, pp. 666–670 (2003)
34. Emmanouilidis, C., Hunter, A., MacIntyre, J.: A Multiobjective Evolutionary Setting for Feature Selection and a Commonality-Based Crossover Operator. In: 2000 Congress on Evolutionary Computation, July 2000, vol. 1, pp. 309–316. IEEE Computer Society Press, Piscataway (2000)
35. Zaliz, R.R., Zwir, I., Ruspini, E.: Generalized Analysis of Promoters: A Method for DNA Sequence Description. In: Coello Coello, C.A., Lamont, G.B. (eds.) Applications of Multi-Objective Evolutionary Algorithms, pp. 427–449. World Scientific, Singapore (2004)

36. de la Iglesia, B., Reynolds, A., Rayward-Smith, V.J.: Developments on a multi-objective metaheuristic (MOMH) algorithm for finding interesting sets of classification rules. In: Coello Coello, C.A., Hernández Aguirre, A., Zitzler, E. (eds.) EMO 2005. LNCS, vol. 3410, pp. 826–840. Springer, Heidelberg (2005)
37. de la Iglesia, B., Richards, G., Philpott, M., Rayward-Smith, V.: The application and effectiveness of a multi-objective metaheuristic algorithm for partial classification. *European Journal of Operational Research* 169, 898–917 (2006)
38. Bandyopadhyay, S., Pal, S.K., Aruna, B.: Multiobjective GAs, Quantitative Indices, and Pattern Classification. *IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics* 34(5), 2088–2099 (2004)
39. Guo, P.F., Bhattacharya, P., Kharna, N.: An Efficient Image Pattern Recognition System Using an Evolutionary Search Strategy. In: *Proceedings of the 2009 IEEE International Conference on Systems, Man, and Cybernetics*, October 2009. IEEE Press, San Antonio (2009)
40. Koza, J.R.: *Genetic Programming. On the Programming of Computers by Means of Natural Selection*. The MIT Press, Cambridge (1992)
41. Jin, Y.: A comprehensive survey of fitness approximation in evolutionary computation. *Soft Computing* 9(1), 3–12 (2005)
42. López Jaimes, A., Coello Coello, C.A.: Applications of Parallel Platforms and Models in Evolutionary Multi-Objective Optimization. In: Lewis, A., Mostaghim, S., Randall, M. (eds.) *Biologically-Inspired Optimisation Methods*, pp. 23–49. Springer, Heidelberg (2009), ISBN 978-3-642-01261-7
43. Lim, D., Jin, Y., Ong, Y.S., Sendhoff, B.: Generalizing Surrogate-Assisted Evolutionary Computation. *IEEE Transactions on Evolutionary Computation* 14(3), 329–355 (2010)
44. Corne, D., Dorigo, M., Glover, F. (eds.): *New Ideas in Optimization*. McGraw-Hill, London (1999)
45. Kennedy, J., Eberhart, R.C.: *Swarm Intelligence*. Morgan Kaufmann Publishers, California (2001)
46. Chander, A., Chatterjee, A., Siarry, P.: A new social and momentum component adaptive PSO algorithm for image segmentation. *Expert Systems with Applications* 38(5), 4998–5004 (2011)
47. Dasgupta, D. (ed.): *Artificial Immune Systems and Their Applications*. Springer, Berlin (1999)
48. Nunes de Castro, L., Timmis, J.: *An Introduction to Artificial Immune Systems: A New Computational Intelligence Paradigm*. Springer, London (2002)
49. Wang, W., Gao, S., Tang, Z.: Improved pattern recognition with complex artificial immune system. *Soft Computing* 13(12), 1209–1217 (2009)
50. Dorigo, M., Stützle, T.: *Ant Colony Optimization*. The MIT Press, Cambridge (2004), ISBN 0-262-04219-3
51. Tambouratzis, G.: Using an Ant Colony Metaheuristic to Optimize Automatic Word Segmentation for Ancient Greek. *IEEE Transactions on Evolutionary Computation* 13(4), 742–753 (2009)

Comparative Diagnostic Accuracy of Linear and Nonlinear Feature Extraction Methods in a Neuro-oncology Problem

Raúl Cruz-Barbosa¹, David Bautista-Villavicencio¹, and Alfredo Vellido²

¹ Universidad Tecnológica de la Mixteca, 69000, Huajuapán, Oaxaca, México
{rcruz,dbautista}@mixteco.utm.mx

² Universitat Politècnica de Catalunya, 08034, Barcelona, Spain
avellido@lsi.upc.edu

Abstract. The diagnostic classification of human brain tumours on the basis of magnetic resonance spectra is a non-trivial problem in which dimensionality reduction is almost mandatory. This may take the form of feature selection or feature extraction. In feature extraction using manifold learning models, multivariate data are described through a low-dimensional manifold embedded in data space. Similarities between points along this manifold are best expressed as geodesic distances or their approximations. These approximations can be computationally intensive, and several alternative software implementations have been recently compared in terms of computation times. The current brief paper extends this research to investigate the comparative ability of dimensionality-reduced data descriptions to accurately classify several types of human brain tumours. The results suggest that the way in which the underlying data manifold is constructed in nonlinear dimensionality reduction methods strongly influences the classification results.

1 Introduction

The diagnostic classification of human brain tumours on the basis of single-voxel proton magnetic resonance spectroscopy (SV-¹H-MRS) information is a non-trivial problem in which dimensionality reduction (DR) is almost mandatory [1]. DR strategies usually take the form of feature selection or feature extraction [2]. In feature extraction using manifold learning models [3], multivariate data are described through a low-dimensional manifold embedded in data space.

Although the Euclidean metric is often used in this setting, similarities between points along the underlying manifold have been shown to be best expressed as geodesic distances or their approximations [4–7]. This is specially important if working with high-dimensional data of unknown intrinsic geometry. Such approximations of the geodesic distances along the manifold can be computationally intensive, and several alternative software implementations of manifold learning models have been recently put forward and compared in terms of their computation times, using several standard and non-standard data sets as benchmarks [8].

Some of the proposed computational time-saving strategies showed great promise in the sense that they were fast while not compromising the amount of data variance explained. They were bundled in a software module that was inserted in a nonlinear dimensionality reduction (NLDR) method, namely ISOMAP [4]. The current brief paper moves this research one step forward to investigate the comparative ability of these dimensionality-reduced data descriptions to accurately classify several types of human brain tumours on the basis of SV- ^1H -MRS information. The performance of the most computationally-effective method is compared to that of two alternative ISOMAP implementations, and to the well-known Principal Component Analysis (PCA) linear technique. Classification is carried out using a simple linear method, namely Linear Discriminant Analysis (LDA), which has previously been successfully applied to this type of data. The results suggest that the way in which the data manifold is constructed in ISOMAP compromises the achieved classification accuracy, although one of the alternative ISOMAP implementations provides, in some of the experiments, comparable accuracy results to those of PCA with fewer features.

2 Methods

2.1 Optimizing the Computation of Geodesic Distances

The explicit calculation of geodesic distances can be computationally impractical. This metric, though, can be approximated by graph distances [9], so that instead of finding the minimum arc-length between two data points lying on a manifold, we would set to find the shortest path between them, where such path is built by connecting the closest successive data points. In this paper, this is done using the K -rule, which allows connecting the K -nearest neighbours. A weighted graph is then constructed by using the data and the set of allowed connections. The data are the vertices, the allowed connections are the edges, and the edge labels are the Euclidean distances between the corresponding vertices. If the resulting graph is disconnected, some edges are added using a minimum spanning tree procedure in order to connect it. Finally, the distance matrix of the weighted undirected graph is obtained by repeatedly applying Dijkstra's algorithm [10], which computes the shortest path between all data samples. This process is illustrated in Fig. 1.

There are different implementation choices for some of the stages involved in the geodesic distance computation (see Fig. 1). Two alternatives for graph representation are the *adjacency matrix* and the *adjacency list*. The former consists in a n by n matrix structure, where n is the number of vertices in the graph. If there is an edge from a vertex i to a vertex j , then the element a_{ij} is 1, otherwise it is 0. This kind of structure provides faster access for some applications but can consume huge amounts of memory. The latter considers that each vertex has a list of which vertices it is adjacent to. This structure is often preferred for sparse graphs as it has smaller memory requirements. Three options for the shortest path algorithm are the standard Dijkstra, Dijkstra using a Fibonacci heap (F-heap) and Floyd-Warshall. For some applications where the obtained graph is

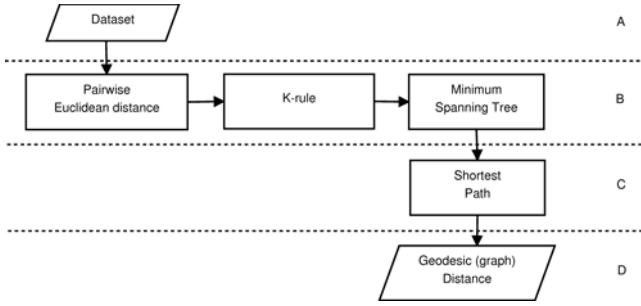


Fig. 1. Graph distance procedure scheme. Stage (A) represents the input data. Stage (B) is for building the weighted, undirected, connected graph. Stage (C) is for computing the geodesic (graph) distance, which is returned in Stage (D).

a sparse graph, Dijkstra’s algorithm can save memory resources by storing the graph in the form of adjacency list and using a F-heap [11] as a priority queue, reducing the time complexity of the algorithm.

Previous research [8] assessed which combination of graph representation and shortest path algorithm produced the best time performance for computing the geodesic distance for datasets with increasing numbers of items. Alternative C++ and Matlab® implementations were also tested.

A combination of adjacency matrix for graph representation and basic Dijkstra as the choice for shortest path algorithm outperformed the other combinations in most settings, due to the faster access to elements in an adjacency matrix, especially for small data sets (as the ones analyzed in this study). The time performance of the C++ implementation of the geodesic distance computation using a matrix representation and the basic Dijkstra algorithm clearly outperformed its Matlab® counterpart. All experiments were performed using a dual-processor 2.3 Ghz BE-2400 desk PC with 2.7Gb RAM.

2.2 Dimensionality Reduction and Classification: ISOMAP and LDA

One way in which DR through feature extraction methods can be categorized is as linear or nonlinear ones. One of the best-known linear methods (and also one that is widely applied to biomedical problems) is PCA. The main aim of PCA is reducing the data dimensionality through an orthogonal transformation, while retaining as much as possible of the data variance along the main extracted dimensions (components).

A recent NLDR method that is increasingly gaining in popularity is ISOMAP [4]. This method is a variant of multi-dimensional scaling, which aims to embed high dimensional data points onto a low dimensional space by preserving inter-point distances as closely as possible. In this method, the geodesic (graph) metric is proposed to compute distances along the manifold instead of the Euclidean one.

The extracted features are then amenable to classification analysis. A basic but efficient classifier (and, again, one commonly used in the analysis of medical data) is LDA. It aims to find a linear combination of features that optimally characterizes or separates different data classes.

3 Materials: SV-¹H-MRS Brain Tumour Database

The experiments in this study concern MRS data acquired at different echo times (short time of echo -STE- and long time of echo -LTE-), as well as with a combination of both by straightforward concatenation. Data belong to a multi-center, international database [12], and consist of: (1) 217 STE spectra, including 58 meningiomas (mm), 86 glioblastomas (gl), 38 metastases (me), 22 astrocytomas grade II (a2), 6 oligoastrocytomas grade II (oa), and 7 oligodendrogliomas grade II (od); (2) 195 LTE spectra, including 55 mm, 78 gl, 31 me, 20 a2, 6 oa, and 5 od. (3) 195 items built by combination (through direct concatenation) of the STE and LTE spectra for the same patients. Only the clinically relevant regions of the spectra were analyzed. They consist of 195 frequency intensity values (measured in parts per million (ppm), an adimensional unit of relative frequency position in the data vector), starting at 4.25 ppm. These frequencies become the observed data features.

The classification experiments involved grouping these tumour types into three classes, namely: G1: low-grade gliomas (a2, oa and od); G2: high-grade malignant tumours (me and gl); and G3: meningiomas (mm). This type of grouping is justified [1, 13] by the well-known difficulty in distinguishing between metastases and glioblastomas, due to their similar spectral pattern.

4 Results and Discussion

The goal of the experiments reported in this section is twofold. Firstly, we aim to show how the way the data manifold is constructed by the ISOMAP model variants affects the classification accuracy. Secondly, we aim to assess the classification results in terms of the number of extracted features and the corresponding accuracy. For all experiments, the K parameter for the K -rule is set to 10 in order to get a connected graph after this rule is applied. The three ISOMAP variants investigated are: a computationally-optimized one implemented in C++ (namely ISOMAP gMod), obtained by embedding the geodesic distance calculation software module described in [8]; and two variants of Tenenbaum's [4] ISOMAP implementation (standard and landmark).

The average classification accuracy results are validated by 10-fold cross-validation. The LDA classification results for G1 vs G2 vs G3 using STE, LTE and STE+LTE spectra are summarily reported in Figs. 2, 3 and 4. The accuracies achieved by the different ISOMAP implementations neatly differ. These differences are the result of the different manners in which the underlying manifold graph representations are obtained by the corresponding variants of the algorithm. ISOMAP standard takes the largest component when the resulting

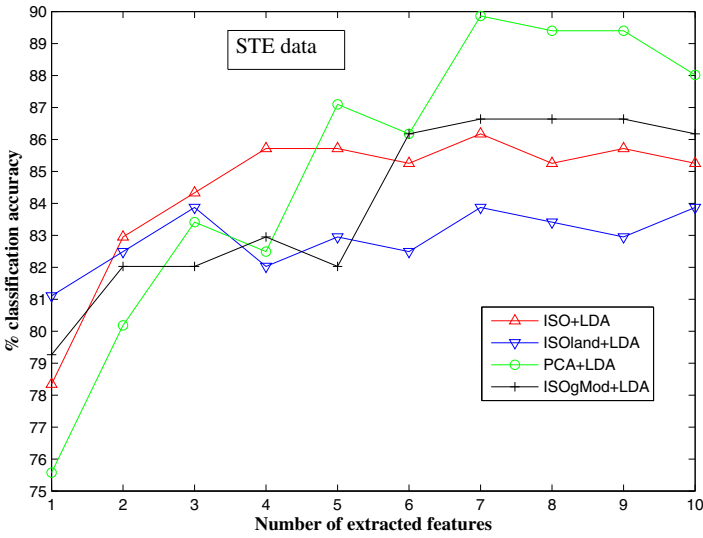


Fig. 2. LDA classification results for G1 vs G2 vs G3 using features extracted from STE spectra. ISOMAP variants: standard (ISO), landmark (ISOLand) and with the computationally-optimized module (ISOGMod).

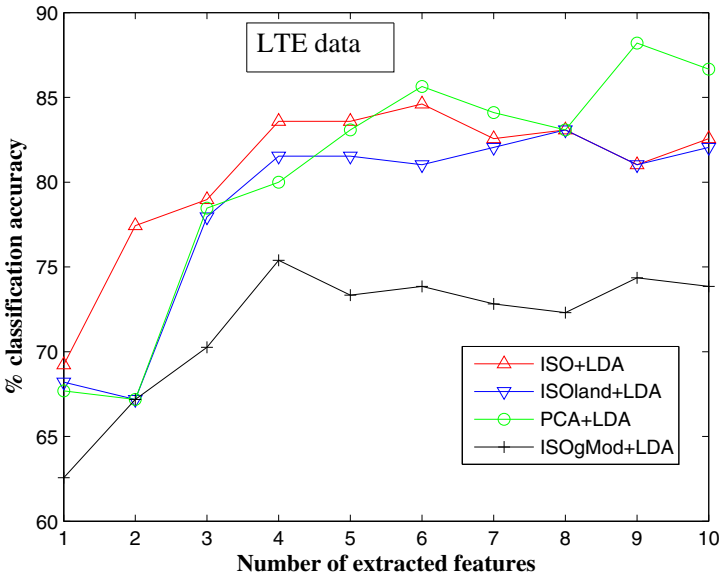


Fig. 3. LDA classification results for G1 vs G2 vs G3 using features extracted from LTE spectra. Legend as in Fig. 2.

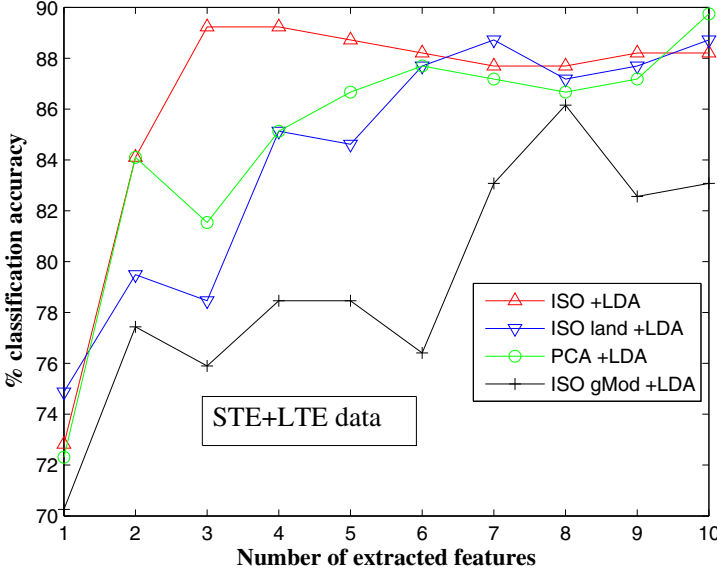


Fig. 4. LDA classification results for G1 vs G2 vs G3 using features extracted from STE+LTE spectra. Legend as in Fig. 2

graph, created by the K -rule, is disconnected. ISOMAP gMod, instead, always builds a connected graph by using a modified minimum spanning-tree procedure, realized by connecting the closest data points between the disconnected components. Finally, ISOMAP landmark randomly selects l landmark points from the original data and the graph is constructed using only those points. Thus, the resulting graph-based distance matrix is strongly dependent on the way the graph is constructed. Since ISOMAP uses the graph distance matrix as input for a multidimensional scaling method which computes the coordinates of the data points in the low dimensional projection space, the extracted features differ notably, hence different classification accuracy results are obtained.

The results reported in Figs. 2 and 3 indicate that, for small numbers of extracted features, the ISOMAP implementations do not provide significantly better LDA accuracies than PCA. For 5 or more features, PCA clearly outperforms ISOMAP to reach just under 90% average accuracy. ISOMAP variants only outperform PCA when data combining the two times of echo are used (as seen in Fig. 4). An extremely parsimonious data representation consisting of just 3 features is enough to obtain an average accuracy just below 90%. Overall, these classification accuracy results are consistent with those reported in literature. For example, in [14], the use of STE+LTE spectra with a PCA+LDA setting achieved a classification accuracy of around 90%, but using a minimum of 8 principal components. The classification results for ISOMAP with gMod implementation deteriorate significantly whenever LTE spectra are used.

5 Conclusion

Brain tumours show a relatively low prevalence in the general population, but their diagnosis and prognosis are challenging and sensitive medical issues. Machine learning and computational intelligence methods can assist medical doctors and expert radiologists in these tasks [15, 16]. The classification of human brain tumours on the basis of high-dimensional SV-¹H-MRS makes the use of DR strategies advisable. Manifold learning techniques using geodesic distances have previously shown promise in this DR task, and there have been efforts to optimize their intensive use of computational resources.

In this paper, we have focused in the ISOMAP NLDR method. Previous research [8] provided evidence that a specific implementation of a geodesic distance computation module in C++ language, as part of the ISOMAP implementation, had an extremely fast performance. In this study, we have carried out preliminary experiments to gauge the ability of the data reduction obtained with the most computationally-effective implementation to provide accurate diagnostic classification for several common brain tumour pathologies on the basis of MRS data. The results reported in this paper show that this most computationally-effective implementation does not perform well in many of the experimental settings. This is evidence that the way in which the data manifold is constructed in NLDR manifold learning methods may compromise the subsequent classification accuracy. The standard ISOMAP implementation, though, is still capable of achieving maximum accuracy in the brain tumour classification problem with far less features than PCA, if a combination of data at different times of echo is used. Further research should include more types of brain tumours, as well as a wider palette of NLDR manifold learning techniques. The interpretability of the features extracted by NLDR, from a medical point of view, should also be investigated.

Acknowledgments. Partial funding for this research was provided by the Mexican SEP PROMEP/103.5/10/5058 and the Spanish MICINN TIN2009-13895-C02-01 research projects. Authors gratefully acknowledge the former INTERPRET European project partners. Data providers: Dr. C. Majós (IDI), Dr. À. Moreno-Torres (CDP), Dr. F.A. Howe and Prof. J. Griffiths (SGUL), Prof. A. Heerschap (RU), Prof. L. Stefanczyk and Dr J. Fortuniak (MUL) and Dr. J. Calvar (FLENI); data curators: Dr. M. Julià-Sapé, Dr. A.P. Candiota, Dr. I. Olier, Ms. T. Delgado, Ms. J. Martín and Mr. A. Pérez (all from GABRMN-UAB). GABRMN coordinator: Prof. C. Arús.

References

1. Vellido, A., Romero, E., González-Navarro, F., Belanche-Muñoz, L., Julià-Sapé, M., Arús, C.: Outlier exploration and diagnostic classification of a multi-centre ¹H-MRS brain tumour database. *Neurocomputing* 72(13-15), 3085–3097 (2009)
2. González-Navarro, F., Belanche-Muñoz, L., Romero, E., Vellido, A., Julià-Sapé, M., Arús, C.: Feature and model selection with discriminatory visualization for diagnostic classification of brain tumours. *Neurocomputing* 73(4-6), 622–632 (2010)

3. Lee, J.A., Verleysen, M.: *Nonlinear Dimensionality Reduction*. Springer, Heidelberg (2007)
4. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. *Science* 290, 2319–2323 (2000)
5. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation* 15(6), 1373–1396 (2003)
6. Roweis, S.T., Lawrence, K.S.: Nonlinear dimensionality reduction by locally linear embedding. *Science* 290, 2323–2326 (2000)
7. Cruz-Barbosa, R., Vellido, A.: Semi-supervised geodesic generative topographic mapping. *Pattern Recognition Letters* 31(3), 202–209 (2010)
8. Bautista-Villavicencio, D., Cruz-Barbosa, R.: On geodesic distance computation: An experimental study. *Advances in Computer Science and Applications, Research in Computing Science* 53, 115–124 (2011)
9. Bernstein, M., de Silva, V., Langford, J.C., Tenenbaum, J.B.: Graph approximations to geodesics on embedded manifolds. Technical report, Stanford University, CA, U.S.A (2000)
10. Dijkstra, E.W.: A note on two problems in connexion with graphs. *Numerische Mathematik* 1, 269–271 (1959)
11. Fredman, M.L., Tarjan, R.E.: Fibonacci heaps and their uses in improved network optimization algorithms. *J. ACM* 34(3), 596–615 (1987)
12. Julià-Sapé, M., et al.: A multi-centre, web-accessible and quality control-checked database of in vivo MR spectra of brain tumour patients. *Magn. Reson. Mater. Phys. MAGMA* 19, 22–33 (2006)
13. Tate, A.R., Majós, C., Moreno, A., Howe, F.A., Griffiths, J.R., Arús, C.: Automated classification of short echo time in In Vivo ^1H brain tumor spectra: a multicenter study. *Magnetic Resonance in Medicine* 49, 29–36 (2003)
14. García-Gómez, J.M., Tortajada, S., Vidal, C., Julià-Sapé, M., Luts, J., Moreno-Torres, A., Van-Huffel, S., Arús, C., Robles, M.: The effect of combining two echo times in automatic brain tumor classification by MRS. *NMR in Biomedicine* 21(10), 1112–1125 (2008)
15. Lisboa, P.J.G., Vellido, A., Tagliaferri, R., Napolitano, F., Ceccarelli, M., Martin-Guerrero, J.D., Biganzoli, E.: Data mining in cancer research. *IEEE Computational Intelligence Magazine* 5(1), 14–18 (2010)
16. Vellido, A., Lisboa, P.J.G.: Neural networks and other machine learning methods in cancer research. In: Sandoval, F., Prieto, A.G., Cabestany, J., Graña, M. (eds.) *IWANN 2007. LNCS*, vol. 4507, pp. 964–971. Springer, Heidelberg (2007)

Efficient Group of Permutants for Proximity Searching

Karina Figueroa Mora¹, Rodrigo Paredes², and Roberto Rangel¹

¹ Universidad Michoacana de San Nicolás de Hidalgo, México

² Universidad de Talca, Chile

karina@fismat.umich.mx, raparede@utalca.cl, a0529275g@correo.fie.umich.mx

Abstract. Modeling proximity searching problems in a metric space allows one to approach many problems in different areas, e.g. pattern recognition, multimedia search, or clustering. Recently there was proposed the permutation based approach, a novel technique that is unbeatable in practice but difficult to compress. In this article we introduce an improvement on that metric space search data structure. Our technique shows that we can compress the permutation based algorithm without losing precision. We show experimentally that our technique is competitive with the original idea and improves it up to 46% in real databases.

1 Introduction

Proximity or similarity searching has become a fundamental task in different areas, for instance artificial intelligence and pattern recognition. The common elements in these areas are a database (e.g. a set of objects) and a similarity measure among its objects (e.g. Euclidean distance). The similarity is modeled by a distance function defined by experts in each application domain, which tells how similar two objects are. The objects are manipulated as black boxes and the only operation permitted is to measure their distance towards another object. Usually, the distance function is quite expensive to compute, therefore our goal is avoiding to make comparisons between objects.

Another problem in these days is that there are huge databases and usually these data use considerably less structure and much less precise queries than traditional database system. Example are multimedia data like images or videos where is common query-by-example search. In view of these challenges a way to face up is building an index that allows us to search quickly. In particular, we are proposing to use a metric space index. A metric space consists in a dataset and a function distance (formally it is described at Section 2).

All metric space search algorithms rely on an index, that is, a data structure that maintains some information on the database in order to save some distance evaluations at search time. Chávez et al. [2] give a complete survey in this area (the two main types of indices), but recently in metric space indices was a third proposal, the permutation-based algorithm [1] which is unbeatable in practice but is difficult to compress. All the proposals to compress this kind of index are prepared to loose precision at retrieval but reducing the size of the index [3, 7].

In this paper we face the compression of the index, by a novel index using cluster techniques. At Section 2 we introduce the basic concepts and discuss the previous work in permutation-based algorithms. At Section 3 we explain our proposal and finally at Section 4 we present the experimental part that support our technique. We finish with our conclusions and future work at Section 5.

2 Basic Concepts and Related Work

Formally, the proximity search problem in a metric space may be stated as follows: there is a universe $\mathbb{X}$ of objects and a nonnegative *distance function* $d : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}^+$ defined among them. The distance satisfies the axioms that make the set a metric space: reflexivity ($d(x, x) = 0$), strict positiveness ($x \neq y \Rightarrow d(x, y) > 0$), symmetry ($d(x, y) = d(y, x)$), and triangle inequality ($d(x, z) \leq d(x, y) + d(y, z)$). Usually, this distance is expensive to compute. We have a finite database $U \subseteq \mathbb{X}$, of size n , which is a subset of the universe of objects.

Basically, there are two kind of queries: *range query* and *K-Nearest Neighbor query* (K-NN). The first one consists in retrieving those objects within a radius to a given query, that is $R(q, r) = \{d(u, q) \leq r \mid \forall u \in U\}$, the second one is to retrieve the K elements of U that are closest to q .

Most of the existing approaches to solve the search problem are *exact algorithms* which retrieve exactly the elements of U as specified above. In [2, 6, 8, 9], most of those approaches are surveyed and explained in detail. These kind of indices usually have a good performance with a few dimensions (two to eight). However, in high dimensions they compare the whole database. An alternative in high dimension is the permutation-based algorithms (PBA).

In [1], the authors show the technique PBA as follows. Let $\mathbb{P} \subseteq U$ be a set of distinguished objects from the database, called *permutants*. Each element of the space, $u \in U$, defines a *permutation* Π_u , where the elements of $\mathbb{P}$ are written in increasing order of distance to u . Ties are broken using any consistent order, for example the order of the elements in $\mathbb{P}$.

Formally, let $\mathbb{P} = \{p_1, p_2, \dots, p_k\}$ and $u \in U$. Then we define Π_u as a permutation of $(1 \dots k)$ so that, for all $1 \leq i < k$ it holds either $d(p_{\Pi_u(i)}, u) < d(p_{\Pi_u(i+1)}, u)$, or $d(p_{\Pi_u(i)}, u) = d(p_{\Pi_u(i+1)}, u)$ and $\Pi_u(i) < \Pi_u(i+1)$.

Given permutations Π_u and Π_q of $(1 \dots k)$, we can compare them using Spearman Rho. It is defined as [4]

$$S_\rho(\Pi_u, \Pi_q) = \sum_{1 \leq i \leq k} (\Pi_u^{-1}(i) - \Pi_q^{-1}(i))^2. \quad (1)$$

Let us give an example of $S_\rho(\Pi_q, \Pi_u)$. Let $\Pi_q = 6, 2, 3, 1, 4, 5$ be the permutation of the query, and $\Pi_u = 3, 6, 2, 1, 5, 4$ that of an element u . A particular element p_3 in permutation Π_u is found two positions off with respect to its

¹ The actual definition in [4] corresponds to $\sqrt{S_\rho(\Pi_q, \Pi_u)}$ in our terminology. We omit the square root because it is monotonous and hence does not affect the ordering.

position in Π_q . The differences of position of each permutant within its permutation are: $1 - 2, 2 - 3, 3 - 1, 4 - 4, 5 - 6, 6 - 5$, and the sum of their squares is $S_\rho(\Pi_q, \Pi_u) = 8$.

There are other similarity measures between permutations [4], however in [1] the authors show that all of them have a similar performance but specially Spearman Rho has a balance between accuracy and time to compute it.

An attempt to compress the index is shown in [3]. They proposed to chose just a few of permutants (the closest ones) in each permutation, and use an inverted index to keep them. In this way they compress the index. They improve in searching time and the space used by the index but retrieval is sacrificed. They can lost up to 20% of the retrieval using less permutants.

There is another attempt to compress the index sacrificing the retrieval which is described in [7]. Basically, they represent the permutation using just two bits, and using another similarity between permutations. They show a good performance but after all retrieval is sacrificed.

3 Our Proposal

The basic idea of a PBA consists in selecting a set of permutants, and produce all the permutations (by comparing every object in the database against the permutants, and sorting in increasing order).

Our proposal consists in using a *set of permutants* instead of a single permutant per each component in the permutation. In this way, we use the same amount of space that the original one but we have more precision.

Formally, we select sets of permutants $\mathbb{G} = \{G_1, G_2, \dots, G_k\}$ where $\mathbb{G} \subseteq \mathbb{U}$ and $G_i \cap G_j = \emptyset, \forall i, j, 1 \leq i, j \leq k$. Each group has m permutants. Then, $\forall u \in \mathbb{U}$, we compared u against the groups, that is, $D_i(u, G_i), 1 \leq i \leq k$ and we sorted D_i by proximity to u . In the next section we discuss the criteria to compute D_i .

3.1 Proximity to a Group

An important factor of the performance in our technique is how to decide the proximity to the groups (i.e., how to compute D). We propose treating each group as a cluster and we can use the methods to classify a cluster.

- Single Linkage: That is, we consider the lowest distance towards all the objects in the group, $D_{i_{min}}(u, G_i) = \min_{p \in G_i} d(p, u)$.
- Complete Linkage: In this case we consider the greatest distance towards all the objects in the group $D_{i_{max}}(u, G_i) = \max_{p \in G_i} d(p, u)$.
- Average: The third propose is the average of distances, that is, $D_{i_{av}}(u, G_i) = \frac{\sum_{p \in G_i} d(p, u)}{|G_i|}$

For simplicity, in the follow we will use only D_{min} , D_{max} o D_{av} .

Figure 1 describes two (of the three) criteria of proximity towards the groups. Permutants are the black points. $G_1 = \{p_1, p_2, p_3\}$, $G_2 = \{p_4, p_5, p_6\}$, and

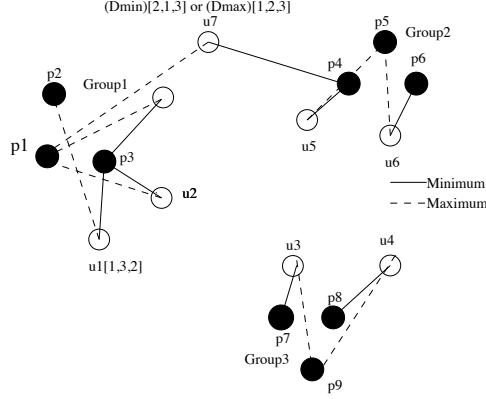


Fig. 1. Criteria for proximity to a group. Continuous line is for the minimum distance (single linkage) to the group, and discontinued line is for the maximum distance (complete linkage).

$G_3 = \{p_7, p_8, p_9\}$. Every point has a line to the closest group according to the criterion used. Notice that each permutation depends on these criteria. For example, u_1 does not change its permutation when considering single or complete criterion, but u_7 does. In fact, using the *single linkage* $\Pi_{u_7} = 2, 1, 3$ ($Dmin(u_7, G_2) \leq Dmin(u_7, G_1) \leq Dmin(u_7, G_3)$); and using the *complete linkage* $\Pi_{u_7} = 1, 2, 3$ ($Dmax(u_7, G_1) \leq Dmax(u_7, G_2) \leq Dmax(u_7, G_3)$). In this figure, we have not drawn the average criterion because it depends on each point and it can confuse the idea. At the experimental section we can see the performance of these criteria.

3.2 Selecting Good Permutants per Group

Figure 1 shows the permutant groups closer each other, but that is not mandatory. In this section we consider other options:

1. Random (RTG). That is choosing elements at random to form a group.
2. Closer to its group (CTG). For this heuristic we propose choosing one permutant p and to pick up closest ones to p for the rest of the group.
3. Farther to its group (FTG). Unlike to the previous one we use the opposite, that is the farther ones to p .

Algorithm 1 shows our proposal. Notice that before to use this algorithm we need to compute every Π_u , $\forall u \in \mathbb{U}$. The fraction f of the database to traverse depends on dimension and number of permutants, to name a few. For more reference see [1].

Analysis. Our technique uses the same amount of space as the original one, that is $\Theta(kn)$, where k is the number of groups. We notice that, we can pack several group identifiers in a single machine word. Also, we keep a small vector

Algorithm 1. $\text{gPermutation}(q, r, f)$

```

1: INPUT:  $q$  is a query and  $r$  its radius,  $f$  is the fraction of the database to traverse.
   We have  $\Pi_u$ ,  $\forall u \in \mathbb{U}$ .
2: OUTPUT: Reports a subset of those  $u \in \mathbb{U}$  that are at distance at most  $r$  to  $q$ .
3: Let  $A[1, n]$  be an array of tuples and  $\mathbb{U} = \{u_1, \dots, u_n\}$ 
4: Compute  $\Pi_q$  with the same criterion used ( $Dmin$ ,  $Dmax$ ,  $Dav$ ) to build every  $\Pi_u$ 
5: Every group was formed using RTG, CTG or FTG
6: for  $i \leftarrow 1$  to  $n$  do
7:    $A[i] \leftarrow \langle u_i, S_\rho(\Pi_{u_i}, \Pi_q) \rangle$ 
8: end for
9: SortIncreasing( $A$ ) // by second component of tuples
10: for  $i \leftarrow 1$  to  $f \cdot n$  do
11:   Let  $A[i] = \langle u, s \rangle$ 
12:   if  $d(q, u) \leq r$  then
13:     Report  $u$ 
14:   end if
15: end for

```

with the identifiers of the permutants within the groups. That is $\Theta(km)$, where m is the group size (we can also try to save some space by packing the identifiers somehow). Another important issue to emphasize is that our technique spends the same time as the original to solve a query because we use the same procedure that the original idea at algorithm [1](#) (lines 6-15).

4 Experimental Section

In this section we evaluate and compare the performance of our technique in different metric spaces, such as synthetic vectors on the unitary cube and real life databases. The experiments were run on an Intel Xeon workstation with 2.4 GHz CPU and 32 GB of RAM with Ubuntu server, running kernel 2.6.32-22.

4.1 Synthetic Databases

In these experiments we used a synthetic database with vectors uniformly distributed on the unitary cube. We use 10,000 points in different dimensions under Euclidean distance. As we can precisely control the dimensionality of the space, we use these experiments to show how much the predictive power of our technique varies with the dimensionality.

Figure [2](#) shows the performance of our technique in different dimensions. We use all the parameters proposed (FTG, CTG, RTG), and Dav , $Dmin$ and $Dmax$. The line with label $m = 1$ is the original permutant idea (one permutant per group). Our technique can improve the original one in dimensions up to 64. As can be seen Dav has better performance than $Dmax$ and $Dmin$, and CTG and RTG has better performance over FTG most of the time.

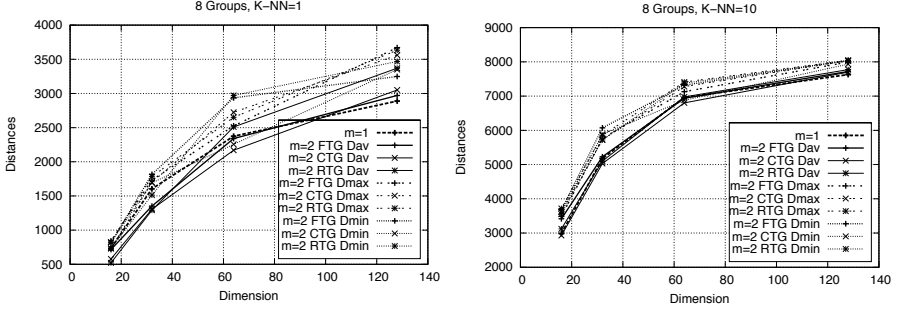


Fig. 2. Vectors uniformly distributed on the unitary cube $n=10,000$, $K\text{-NN}=1,10$

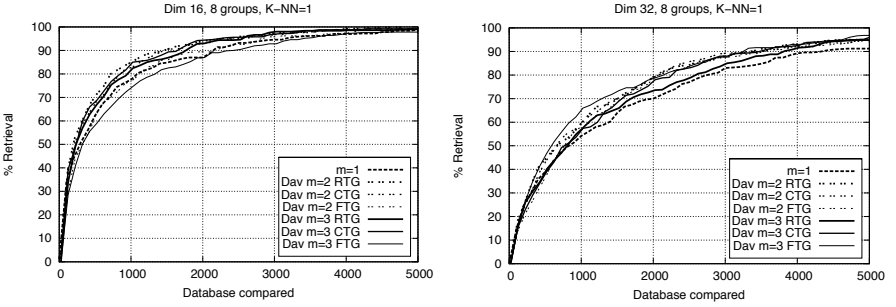


Fig. 3. Dimension 16 and 32, $n=10,000$, $K\text{-NN}=1$. We use the three criteria to sort objects by proximity to the groups, and the three criteria to choose permutants per group.

Figure 3 shows that experiments with $m = 2$ and $m = 3$ retrieve faster than $m = 1$. In this case, for simplicity we only kept $8n$ bytes for the index. We also use 8×2 or 8×3 machine words to identify each of the permutants per group.

However, we can pack the permutant index in just $3n$ bytes plus the space of the permutant identifiers (as we are considering just eight groups, so we need 24 bits for each permutation).

4.2 Real Databases

In this section we show the performance of our heuristic in a real-world space of images.

Flickr. The set of image objects were taken from Flickr, using the URL provided by the SAPIR collection [5]. The content-based descriptors extracted from the images were: Color Histogram $3 \times 3 \times 3$ using RGB color space (a 27 dim vector), Gabor Wavelet (a 48 dim vector), Efficient Color Descriptor (ECD) 8×1 using RGB color space (a 32 dim vector), ECD 8×1 using HSV color space (a 32 dim

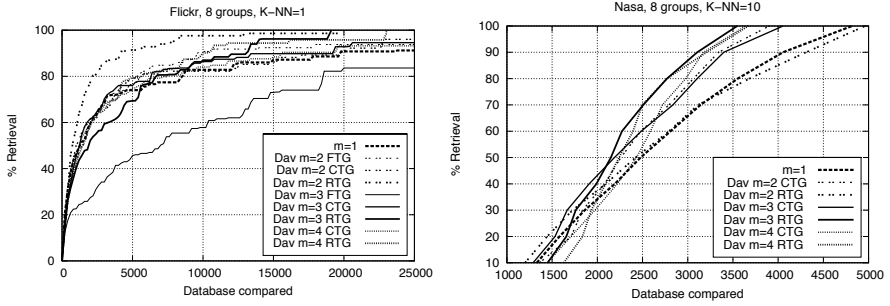


Fig. 4. Real-life database (left) Flickr database, (right) Nasa database. K-NN=1,10.

vector), and Edge Local 4×4 (a 80 dim vector). The distance function used was Euclidean distance. The dataset size was 1 million of images.

Nasa. A set of 40,150 20-dimensional feature vectors, generated from images downloaded from NASA² and with duplicate vectors eliminated. We also use Euclidean distance.

At figure 4 the original idea is labeled (black line) with $m = 1$, that is 1 permutant per group (randomly selected). With our technique we can retrieve the 100% of the query using up to 46% less comparison using two permutants per group ($m = 2$ and RTG). With three permutants per group we can also get faster the 100% of retrieval but it grows slower than $m = 2$. Notice that comparing just 5000 distances our technique has retrieved 90% of the data while the original idea has retrieved only 75% of the data. The plots also show that the FTG strategy has the worst performance when choosing the permutant groups. Our technique has a better performance on a real database.

5 Conclusions and Future Work

The permutation based algorithm for proximity searching is unbeatable in high dimension, but this kind of index has not been improved using less space. All the attempts to compress the index lose precision at retrieval.

In this article we present an improvement on the permutation-based algorithm. The main idea is to have a set of permutants instead of a single permutant (as the original idea). With our heuristic we can improve the permutation based algorithm using the same amount of space as the original idea. The experimental part shows that our technique can improve the performance of the original idea up to 46% in real databases.

In future work we will select different amount of permutants per group. That is, some groups will have more permutants than others.

² At <http://www.dimacs.rutgers.edu/Challenges/Sixth/software.html>

Acknowledgements

This paper was partially supported by National Council of Science and Technology (CONACyT) of México and by Universidad Michoacana de San Nicolás de Hidalgo, México.

References

1. Chávez, E., Figueroa, K., Navarro, G.: Effective proximity retrieval by ordering permutations. *IEEE Trans. on Pattern Analysis and Machine Intelligence (TPAMI)* 30(9), 1647–1658 (2009)
2. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in metric spaces. *ACM Computing Surveys* 33(3), 273–321 (2001)
3. Esuli, A.: Mipai: using the pp-index to build an efficient and scalable similarity search system. In: *Similarity Searching and Applications*, pp. 146–148 (2009)
4. Fagin, R., Kumar, R., Sivakumar, D.: Comparing top k lists. *SIAM J. Discrete Math.* 17(1), 134–160 (2003)
5. Falchi, F., Kacimi, M., Mass, Y., Rabitti, F., Zezula, P.: Sapir: Scalable and distributed image searching. In: *CEUR Workshop Proceedings SAMT (Posters and Demos)*, vol. 300, pp. 11–12 (2007)
6. Hjaltonson, G., Samet, H.: Index-driven similarity search in metric spaces. *ACM Transactions Database Systems* 28(4), 517–580 (2003)
7. Sadit, E., Chávez, E.: On locality sensitive hashing in metric spaces. In: *Similarity Search and Applications*, pp. 67–74. ACM Press, New York (2010), ISBN: 978-1-4503-0420-7
8. Samet, H.: *Foundations of Multidimensional and Metric Data Structures* (The Morgan Kaufmann Series in Computer Graphics and Geometric Modeling). Morgan Kaufmann Publishers Inc., San Francisco (2005)
9. Zezula, P., Amato, G., Dohnal, V., Batko, M.: *Similarity Search: The Metric Space Approach*. In: *Advances in Database Systems*, vol. 32, Springer, Heidelberg (2006)

Solving 3-Colouring via 2SAT

Guillermo De Ita, César Bautista, and Luis C. Altamirano

Computer Sciences, Universidad Autónoma de Puebla, México

{deita,bautista,altamirano}@cs.buap.mx

Abstract. The 3-Colouring of a graph is a classic NP-complete problem. We show that some solutions for the 3-Colouring can be built in polynomial time based on the number of basic cycles existing in the graph. For this, we design a reduction from proper 3-Colouring of a graph G to a 2-CF Boolean formula F_G , where the number of clauses in F_G depends on the number of basic cycles in G . Any model of F_G provides a proper 3-Colouring of G . Thus, F_G is a logical pattern whose models codify proper 3-Colouring of the graph G .

Keywords: 3-Colouring, SAT Problem, Efficient Computing.

1 Introduction

Graph vertex colouring is an active field of research, with many interesting problems [10,7,14]. In the colouring of a graph, we ask to colour the nodes properly with the smallest possible number of colours so that no two adjacent nodes receive the same colour. If such a colouring with k colours exists, the graph is k -colourable. The chromatic number of a graph G , denoted as $\chi(G)$, represents the minimum number of colours for proper colouring G .

To determine the chromatic number $\chi(G)$ of a graph G is a NP-complete problem, even for graphs with degree 3 or higher. As a consequence, there are many unanswered questions related to the colouring of a graph [10].

Many important combinatorial problems are modelled as Constraint Satisfaction Problems (CSP). CSP form a large class of combinatorial logic problems that contains many important ‘real-world’ problems. An instance of a CSP consists of a set V of variables, a domain D , and a set F of constraints. For example, the domain may be $\{0, 1\}$, and the constraints may be clauses of a Boolean formula in Conjunctive Form (CF). The objective is to assign values in D to the variables in such a way that all constraints are satisfied.

One application of the CSP has been to recognize combinatorial patterns in graphs and to apply techniques developed mainly for solving CSP. For example, let $G = (V, E)$ be an undirected graph where $V = \{v_1, \dots, v_n\}$. We can associate a monotone 2-CF formula F_G with variables V , and where $F_G = \bigwedge_{(u,v) \in E} (u \vee v)$.

A set $I \subseteq V$ is called an independent set if no two of its elements are joined by an edge. Let $S_I = \{v_j : j \in I\}$ be an independent set in G , then the assignment defined by $x_i = 0$ if $i \in I$ and $x_i = 1$ otherwise, satisfies F_G . The reason is that in every clause $(x_i \vee x_j)$ (representing an edge $\{v_i, v_j\}$) at least one of the variables

is assigned to 1, since otherwise the nodes v_i and v_j are in the independent set S_I and then there will not be an edge in G between them.

The above reduction shows us how to apply CSP for modeling different combinatorial patterns of a problem, and in general, for representing key structures which allow us to design smart algorithms. In general, the CSP has been a helpful language for modeling and represent combinatorial patterns on graphs.

1.1 Related Work

Usually, the problem of computing the chromatic number of a graph has been attacked applying dynamic programming and improving upper bounds on the number of maximal independent sets.

In the area of exact algorithms for computing the chromatic number of a graph, the first algorithm was designed by Lawler [11] and it has a running-time of $O * ((1 + \sqrt[3]{3})^n) = O(2.4423^n)$, where $O * (\cdot)$ means that polynomial factors are ignored. The Lawler's algorithm had not been improved for 25 years [4].

Following the line of exact algorithms and using maximal independent sets to compute the chromatic number, Eppstein established an $O*((4/3 + (3/4)^{4/3})^n) = O(2.4151^n)$ time algorithm [2]. And Byskov provided an $O(2.4023^n)$ algorithm [5]. All of those algorithms utilize exponential size memory. While Bodlaender proposed an $O(5.283^n)$ running-time algorithm but using a polynomial size memory [4].

When the number of colourings is fixed, faster algorithms have been designed. The currently best known bound for k -colouring, are: $O(1.3289^n)$ for $k = 3$ [2]. For $k = 4$, the $O(1.7504^n)$ algorithm by Byskov [5], and for $k = 5$, the $O(2.1020^n)$ algorithm by Byskov and Eppstein (see [5]). Each one of those algorithms uses polynomial size memory.

There has also been some related work on approximate or heuristic 3-Colouring algorithms. Blum and Karger have shown that any 3-chromatic graph can be coloured with $O(n^{3/14})$ colours in polynomial time [3]. Alon described a technique for colouring random 3-chromatic graphs in expected polynomial time [1]. Vlasie has described a class of instances which are (unlike random 3-chromatic graphs) difficult to colour [13].

On the other hand, for $k \leq 2$ the problem is polynomially solvable, as is the general problem of colouring for many classes of graphs such as: interval graphs, chordal graphs, comparability graphs [7], the 3-Colouring for AT-free graphs [12] and more generally, for perfect graphs [9]. In these cases, special structures (patterns) have been found which identify those classes of graphs such that they also allow the design of polynomial-time algorithms on them.

We consider here the lowest value for k where the k -colouring is a NP-complete problem, that is, the 3-Colouring problem, and instead of using maximal independent sets of the input graph G , we consider here the CSP as a formulation for codifying proper 3-Colourings. We show that a subset of proper 3-Colourings of G exists which are codified as models of a 2-CF Boolean formula F_G , where the number of clauses in F_G depends on the number of basic cycles in G .

Then, to build the 2-CF F_G and to determine its models (if any exist) can be done in polynomial time on the number of basic cycles of the graph. And since 2SAT problem is polynomially solvable, our proposal changes the parameter for measure the time-complexity of the 3-Colouring problem: from the size of the input graph G to the number of basic cycles in G .

The main advantages of our proposal is to show how to build the 2-CF F_G whose models represent proper 3-Colourings of G , as well as to compute such 3-Colourings in polynomial time based on the number of basic cycles of G .

2 Preliminaries

Let $G = (V, E)$ be a simple graph (i.e., finite, undirected, loop-less and without multiple edges). $V(G)$ and $E(G)$ are also used to denote the sets V and E , respectively. Two vertices v and w are called *adjacent* if there is an edge $\{v, w\} \in E$, connecting them. Sometimes, the shorthand notation of $u v$ is used for denoting the edge $\{u, v\}$. The cardinality of a set A is denoted by $|A|$.

The neighborhood of a vertex $v \in V$ is the set $N(v) = \{w \in V : \{v, w\} \in E(G)\}$, and the closure neighborhood of v is $N[v] = N(v) \cup \{v\}$. The degree of a node v , denoted by $\delta(v)$, is the number of neighbors it has, that is $\delta(v) = |N(v)|$. The degree of a graph G is $\Delta(G) = \max_{x \in V} \{\delta(x)\}$.

A path from v to w is a sequence of edges: $v_0 v_1, v_1 v_2, \dots, v_{n-1} v_n$ such that $v = v_0$ and $v_n = w$ and v_k is adjacent to v_{k+1} , for $0 \leq k < n$. The length of the path is n . A simple path is a path where $v_0, v_1, \dots, v_{n-1}, v_n$ are all distinct.

A cycle is just a nonempty path such that the first and last vertices are identical, and a simple cycle is a cycle in which no vertex is repeated, except that the first and last vertices are identical. A simple cycle is even if it has an even number of edges while the cycle is odd when it has an odd number of edges. A graph G is acyclic if it has no cycles.

Definition 1. Let G be a graph. Then G is said to be connected if for each pair $u, v \in V$ there exists a path from u to v . A connected component of G is a maximal subgraph of G induced by this equivalence relation, that is, a connected component is not a proper subgraph of any other connected subgraph of G .

For example, a tree graph is an acyclic connected graph. Given an undirected connected graph $G = (V, E)$ to apply a depth-first search for traversing G produces a tree graph T_G , where $V(T_G) = V(G)$. The edges in T_G are called *tree edges*, whereas the edges in $E(G) \setminus E(T_G)$ are called *back edges*.

Let $e \in E(G) \setminus E(T_G)$ be a given back edge. The union of the path in T_G between the endpoints of e with the edge e itself forms a simple cycle, such cycle is called a basic (or fundamental) cycle of G with respect to T_G . Each back edge holds the maximum path contained in the basic cycle which is part of.

Let us consider $\mathcal{C} = \{C_1, \dots, C_k\}$ the set of basic cycles found during the depth first search on G . Given any pair of basic cycles C_i and C_j from $\mathcal{C}$, if C_i and C_j share any edges, they are called *intersecting* cycles; otherwise they are

called *independent* cycles. We say that a basic cycle $C_i \in \mathcal{C}$ is independent if C_i and $C_j \in \mathcal{C}$ are disjoint for each $j \neq i$.

It is known that any acyclic graph is 2-colourable and therefore, 3-colourable as well. Furthermore, if a graph G has only cycles of even length then G is also 2-colourable since to alternate two colours with respect to the levels of the tree T_G builds a proper 2-colouring.

Cycles of odd length require at least 3 colours to make proper colourings. Therefore, the subgraph structures which generate conflict for the 3-Colourings are the odd cycles. We show in the following subsection how to treat those structures for building 3-Colourings.

3 A Heuristic for Building Proper 3-Colourings

Let $G = (V, E)$ be a simple connected graph. Assume an order on the vertices V over which depth-first search is based. Then, the application of the depth-first search over G builds a new depth-first graph G' and a tree graph T_G . During the depth-first search, two sets C_o and C_e are formed: C_o contains the basic cycles of odd length in G and C_e the basic cycles of even length in G . The following procedure is a heuristic for reducing the problem of searching for a 3-Colouring of G to determine the satisfiability of a 2-CF F_G .

Procedure 3_Col(G)

1. We start colouring the nodes in T_G using two colors $\{G, R\}$: the same colour for all nodes at the same level, and with different colours for different levels, that is, alternating colours with respect to the levels of the tree T_G .
2. If $C_o = \emptyset$ then G is 2-Colourable.
3. If all the odd basic cycles are independent, then G is 3-colourable.

Two alternating colours on T_G and a third colour assigned to just one of the two nodes of each back edge of the cycles in C_o build a proper 3-Colouring of G .

4. Let $CE \subseteq C_e$ be the set of cycles of even length which are intersecting with any odd cycle in G' , i.e.,

$$CE = \{C_i \in C_e : \text{length of } C_i \text{ is even and } \exists C_j \in C_o, C_i \cap C_j \neq \emptyset\}.$$

5. We assume $k = |C_o|$, there are k odd basic cycles in G' . Let $De = \{be_1, \dots, be_k\}$ be the set of k back edges formed from C_o , such that each $be_j \in De$ is the back edge of $C_j \in C_o, j = 1, \dots, k$.
6. For each $be_j = \{x_j, y_j\} \in De, j = 1, \dots, k$ the two binary clauses are built: $A_j = (x_j \vee y_j) \wedge (\overline{x_j} \vee \overline{y_j})$.
7. For each pair of different back edges in De : $be_l = \{x_l, y_l\}$ and $be_j = \{x_j, y_j\}$, $l \neq j$ where one of its endpoints (x_j or y_j) is adjacent to (x_l or y_l), e.g. assuming that x_l and x_j are adjacent nodes, the following binary clause is built: $B_j = (\overline{x_l} \vee \overline{x_j})$.

8. For each back edge $e_l = \{x_l, y_l\}$ of a cycle in CE , the following binary clause is built: $E_l = (\overline{x_l} \vee \overline{y_l})$.
9. Let F_G be the 2-CF formed by the conjunction of the A'_i s and B'_j s and E'_l s built in the previous steps.
10. Determine if F_G is satisfiable or not. If F_G is satisfiable then G is 3-colourable and any model of F_G represents a proper 3-Colouring of G . Otherwise, our heuristic can not determine a proper 3-Colouring of G .

We show now that any satisfiable assignment of F_G determines a proper 3-Colouring for G . First, we explain in Table 1 what it means that the clauses in F_G are satisfied.

Table 1. Relationship between the third colour and the satisfiability of the clauses

Clauses	Intended meaning
A'_j s	For each back edge of an odd cycle, just one of its two nodes must be coloured with W
B'_j s	Any pair of nodes with colour W must not be adjacent
E'_j s	Both nodes of a back edge of an even cycle can not be assigned the colour W

Theorem 1. *Any satisfiable assignment of the 2-CF F_G formed by $3\_Col(G)$ determines a proper 3-Colouring for the input graph G .*

Proof:

If there is a satisfiable assignment s for F_G , the variables in s which are assigned the value True correspond to the nodes which are assigned the third colour W , while the variables in s which are assigned the value false remain with the same colour assigned in the step 1 of the procedure 3_Col .

The set of clauses in A'_j s guarantees that all odd cycles have the colour W assigned to one of their nodes and therefore, all cycle has been 3-colourable, while the set of clauses in B'_j s and E'_j s prohibit assigning the third color at two adjacent nodes. As all remaining substructures in T_G have been previously coloured with G and R then s represents a proper 3-Colouring of G .

On the other hand, if F_G is unsatisfiable then there is no way to assign colour W to one of the endpoints of a back edge of an odd cycle and at the same time to guarantee that two nodes coloured with W are not adjacent. So, 3_Col can not build a proper 3-Colouring for G .

4 Example: Generalized Petersen Graphs

The generalized Petersen graph $P(n, k)$ is made of vertices $a_0, \dots, a_{n-1}, b_0, \dots, b_{n-1}$ and edges $\{a_i, a_{i+1}\}$, $\{a_i, b_i\}$ and $\{b_i, b_{i+k}\}$, $i = 0, \dots, n-1$ where the subindex sum is taken inside of the additive abelian group $\mathbb{Z}_n$ of integers module n . The subgraph generated by $a_0, \dots, a_{n-1}$ is called the *outside subgraph* while the subgraph generated by $b_0, \dots, b_{n-1}$ is the *inside subgraph*.

In the following, we are assuming that depth-first search decides what node must be visited based on the following order on the vertices:

$$b_{n-1} > \dots > b_0 > a_{n-1} > \dots > a_0,$$

i.e, the greatest vertex is visited first.

Note that two vertices b_i, b_j are in the same connected component of the inside subgraph if and only if $j = i + ks \pmod{n}$ for some integer s , i.e., such connected components are the elements of the quotient $\mathbb{Z}_n k / \mathbb{Z}_n$, which has order g : the greatest common divisor (gcd) of n and k . Therefore, the number of connected components of the inner subgraph of $P(n, k)$ is g and each component has the same number of elements n/g .

Let us recall that when a depth-first search is applied, the pair of numbers called discovery and finishing time are quite useful (see [6]).

Lemma 1. *For each v vertex of $P(n, 2)$ let $d[v], f[v]$ the discovery time and the finishing time, respectively, after calling depth-first search on $P(n, 2)$. Then*

1. $d[v] + f[v] = 4n + 1$ and $d[v] \leq 2n$, for any vertex v of $P(n, 2)$;
2. the depth-first search tree $T_{P(n, 2)}$ is a chain.

Proof. Since g , the gcd of n and 2, is 1 or 2, then the number of connected components of the inner subgraph is 1 or 2.

1. By definition, $d[b_{n-1}] = 1$, then, since the vertices of the inside graph are greater than those in the outside graph, they are explored first. Then, the case $g = 1$ is straightforward. Otherwise, if $g = 2$: after the connected component of b_{n-1} in the inner subgraph is exhausted, at the vertex b_1 , a node in the outer subgraph is reached, namely a_1 , ($d[a_1] = (n/d) + 1$) then a_2 is explored, and then b_2 in the other connected component, different from the component of b_{n-1} , is explored. This second component is exhausted at b_0 ($d[b_0] = n + 2$), then the vertices explored are $a_0, a_{n-1}, a_{n-2}, \dots, a_3$ and $d[a_0] = n + 3, d[a_{n-1}] = n + 4, d[a_{n-2}] = n + 5, \dots, d[a_3] = 2n$. Now all the vertices have been explored, so $f[a_3] = 2n + 1$ and we have to trace back all the vertices, so $f[a_4] = 2n + 2, \dots, f[a_{n-1}] = 3n - 3$ and so on. Note that $f[v] + d[v] = 4n + 1$, for all vertices v of $P(n, k)$.
2. Let u, v be two different vertices of $P(n, 2)$. Then $d[u] < d[v]$ or $d[u] > d[v]$. In the former case, due to 1) we have $d[u] < d[v] < 4n + 1 - d[v] = f[v] < 4n + 1 - d[u] = f[u]$. It follows that v is a proper descendant of vertex u in the depth-first forest graph $T_{P(n, 2)}$ (see Theorem [6]). Similarly, in the latter case, we get that u is a proper descendant of v . Thus, for any u, v vertices in $T_{P(n, 2)}$, u is a proper descendant of v , or v is a proper descendant of u . This means that $T_{P(n, 2)}$ is a chain.

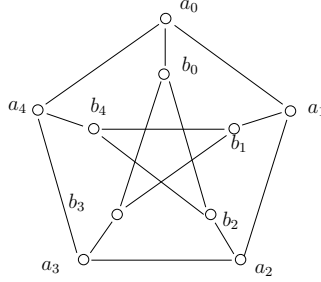
Proposition 1. *Let $n \geq 2$, the number of basic cycles of $P(n, 2)$ is $O(n)$.*

Proof. Since the degree of $P(n, 2)$ is 3 and $T_{P(n, 2)}$ is a chain, then there are at most, $6n$ back edges.

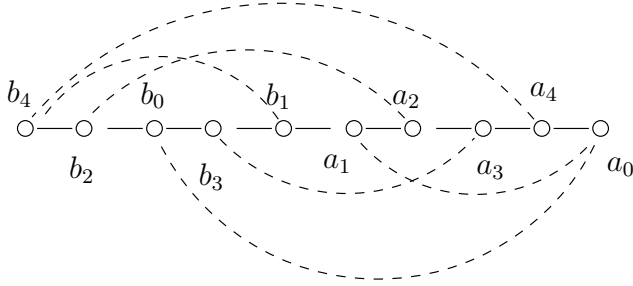
Then $3\_Col(P(n, k))$ determines 3-Colouring in polynomial time.

4.1 Applying the Heuristic on the Petersen Graph

The generalized Petersen graph $P(5, 2)$ is

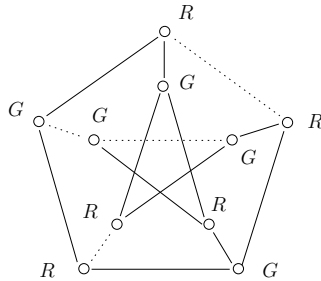


The depth-first search tree $T_{P(5,2)}$ represented by the bold lines, is:



where the dashed lines are the back edges. Then the basic cycles of odd length are $C_o = \{cycle(a_1, a_2, a_3, a_4, a_0), cycle(b_4, b_2, b_0, b_3, b_1, a_1, a_2, a_3, a_4), cycle(b_3, b_1, a_1, a_2, a_3), cycle(b_4, b_2, b_0, b_3, b_1)\}$; while the basic cycles of even length are $C_e = \{cycle(b_0, b_3, b_1, a_1, a_2, a_3, a_4, a_0), cycle(b_2, b_0, b_3, b_1, a_1, a_2)\}$.

Using $T_{P(5,2)}$ we put the alternate colours $\{G, R\}$ on $P(5, 2)$. Notice the conflicting colouring at the dashed line edges.



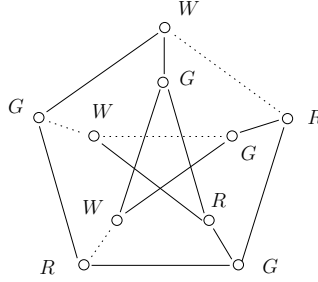
As the basic odd cycles are not independent, the sets $De = \{\{a_1, a_0\}, \{b_3, a_3\}, \{b_4, a_4\}, \{b_1, b_4\}\}$, $CE = \{cycle(b_0, b_3, b_1, a_1, a_2, a_3, a_4, a_0), cycle(b_2, b_0, b_3, b_1, a_1, a_2)\}$ are formed, $k = |C_o| = 4$.

Consider the clauses: $A_1 = (a_1 \vee a_0) \wedge (\overline{a_1} \vee \overline{a_0})$, $A_2 = (b_3 \vee a_3) \wedge (\overline{b_3} \vee \overline{a_3})$, $A_3 = (b_4 \vee a_4) \wedge (\overline{b_4} \vee \overline{a_4})$ and $A_4 = (b_1 \vee b_4) \wedge (\overline{b_1} \vee \overline{b_4})$. $B_1 = (\overline{a_0} \vee \overline{a_4})$, $B_2 = (\overline{b_1} \vee \overline{b_3})$, $B_3 = (\overline{b_1} \vee \overline{a_4})$. From CE we get $E_1 = (\overline{b_0} \vee \overline{a_0})$ and $E_2 = (\overline{b_2} \vee \overline{a_2})$.

$F_{P(5,2)}$ is $A_1 \wedge A_2 \wedge A_3 \wedge A_4 \wedge B_1 \wedge B_2 \wedge B_3 \wedge E_1 \wedge E_2$, i.e.,

$$F_{P(5,2)} : (a_1 \vee a_0) \wedge (\overline{a_1} \vee \overline{a_0}) \wedge (b_3 \vee a_3) \wedge (\overline{b_3} \vee \overline{a_3}) \wedge (b_4 \vee a_4) \wedge (\overline{b_4} \vee \overline{a_4}) \wedge (b_1 \vee b_4) \wedge (\overline{b_1} \vee \overline{b_4}) \wedge (\overline{a_0} \vee \overline{a_4}) \wedge (\overline{b_1} \vee \overline{b_3}) \wedge (\overline{b_1} \vee \overline{a_4}) \wedge (\overline{b_0} \vee \overline{a_0}) \wedge (\overline{b_2} \vee \overline{a_2}).$$

Thus, $F_{P(5,2)}$ is satisfiable, for instance, a model is $a_0 = 1, a_1 = 0, a_2 = 0, a_3 = 0, a_4 = 0$ and $b_0 = 0, b_2 = 0, b_3 = 1, b_4 = 1$. Hence, $P(5, 2)$ is 3-colourable, and the 3-Colouring is obtained by change the colors of a_0, b_3 and b_4 to W . The 3-Colouring for $P(5, 2)$ is shown in the following figure.



5 Algorithm's Time-Complexity

Let $G = (V, E)$ be a simple connected graph with $n = |V|$ and $m = |E|$. A depth-first search over G is of order $O(n + m)$ and it builds an equivalent depth-first graph G' . The set of basic cycles $C = C_o \cup C_e$ is built during the depth-first search. As every cycle has no more than n nodes, then to determine the size of the cycle during the depth-first search involves $O(n + m)$ basic operations.

If the number of basic cycles in T_G is not big, e.g., it is upper bounded by a polynomial function on n or m then C is formed in order $O(poly(n + m))$, $poly$ being a polynomial function. Otherwise, it is possible to consider topological graphs where their number of basic cycles grow as an exponential function over n , e.g., the class of graphs K_n (the complete graphs with n nodes), and for this class of graphs, C is built in exponential time-complexity on n .

Let us assume $nc = |C|$ as the number of basic cycles of the input graph. The time required to determine if there are intersecting cycles in a graph is of order $O(nc^2)$ since it consists of the following loop: for all $C \in \mathcal{C}$, for all $C' \in \mathcal{C}$, $C \neq C'$, test if $C \oplus C' \neq \emptyset$, $\oplus$ being the or-exclusive between the set of edges of the cycles. Steps 2, 3 and 4 in the algorithm 3.Col are done in polynomial time on nc , that is, $O(nc^2)$. Step 1 is of order $O(n)$ since it consists of assigning one of the two colours $\{G, R\}$ to each node of the graph. Step 5 runs in order $O(nc)$.

Steps 6,7 and 8 build a 2-CF F_G , the time-complexity for building F_G is related to the size $|F_G|$ which is the number of clauses in F_G . Step 6 builds $2 * |C_o|$ clauses since for each odd basic cycle there are two binary clauses in the A'_j s. Step 7 builds at most $|C_e|$ clauses, one clause for each intersecting even cycle, that is, the E'_j s have no more than $|C_e|$ clauses. Step 8 builds one clause

for each pair of adjacent nodes u and v which are endpoints from two different back edges. For this case, at most $n - 1$ clauses would be formed since at most each tree edge of T_G could hold that latter condition.

Thus $|F_G| \leq 2 * |Co| + |Ce| + (n - 1) < 2 * nc + n$. Of course, for graphs holding $nc \leq poly(n)$ (when the number of cycles are upper bounded by a polynomial function on the number of nodes) 3_Col runs in polynomial time on n . Furthermore, 3_Col builds proper 3 colourings on that polynomial time.

Otherwise, when $nc \gg n$ (the class of graphs where the number of cycles grows as an exponential function of n), we have that $|F_G| < 3 * nc$ and then, 3_Col determines the satisfiability of F_G in polynomial time based on the number of cycles that the input graph G has and so, a proper 3-Colouring of G .

6 Conclusions

We have designed an appropriate combinatorial pattern representing proper 3-Colourings of a graph. Such a pattern is codified by models of a 2-CF.

Thus, the problem of searching for proper 3-Colourings of a graph G is reduced to build satisfiable assignments of a formula F_G in 2-CF, where the number of clauses in F_G depends on the number of basic cycles in G .

The set of satisfiable assignments of F_G is a subset of all proper 3-Colouring of G . Thus, any model of F_G gives a way to 3-Colour G , although it could happen that G is 3-Colourable but F_G will be unsatisfiable.

Since 2SAT problem is polynomially soluble, our reduction shows that to build some proper 3-Colourings of a graph G can be done in polynomial time on the number of basic cycles in G .

References

1. Alon, N., Kahale, N.: A spectral technique for coloring random 3-colorable graphs. SIAM Jour. Comput. 26(6), 1733–1748 (1997), <http://www.research.att.com/kahale/papers/jour.ps>
2. Beigel, R., Eppstein, D.: 3-coloring in time $O(1.3289^n)$. Journal of Algorithms 54(2), 168–204 (2005)
3. Blum, A., Karger, D.: An $O(n^{3/14})$ -coloring algorithm for 3-colorable graphs. Inf. Proc. Lett. 61(1), 49–53 (1997), http://www.cs.cmu.edu/http://www.cs.cmu.edu/avrim/Papers/color_new.ps.gz
4. Bodlaender H.L., Kratsch D.: An exact algorithm for graph coloring with polynomial memory. Technical Report UU-CS-2006-015 (2006), <http://www.cs.uu.nl>
5. Byskov J.M.: Exact Algorithms for graph colouring and exact satisfiability. Phd thesis, University of Aarhus, Denmark (2005)
6. Cormen, T.H., Leiserson, C.E., Rivest, R.L., Stein, C.: Introduction to Algorithms, 3rd edn. MIT Press, Cambridge (2009)
7. Golumbic, M.C.: Algorithmic Graph Theory and Perfect Graphs, 2nd edn. North-Holland, Amsterdam (2004)
8. Greenhill, C.: The complexity of counting colourings and independent sets in sparse graphs and hypergraphs. Computational Complexity (1999)

9. Grötschel, M., Lovász, L., Schrijver, A.: The ellipsoid method and its consequences in combinatorial optimization. *Combinatorica* 1, 169–197 (1981)
10. William, K., Kreher Donald, L.: *Graphs, Algorithms, and Optimization*. Chapman & Hall/CRC, Boca Raton (2005)
11. Lawler, E.: A note on the complexity of the chromatic number problem. *Information Processing Letters* 5, 66–67 (1976)
12. Stacho, J.: 3-Colouring AT-Free Graphs in Polynomial Time. In: Cheong, O., Chwa, K.-Y., Park, K. (eds.) *ISAAC 2010, Part II. LNCS*, vol. 6507, pp. 144–155. Springer, Heidelberg (2010)
13. Vlasie, R.D.: Systematic generation of very hard cases for graph 3-colorability. In: *Proc. 7th-IEEE Int. Conf. Tools with Artificial Intelligence*, pp. 114–119 (1995)
14. Wilson, R.A.: *Graphs, Colourings and the Four-colour Theorem*. Oxford University Press, Oxford (2002)

Classifier Selection by Clustering

Hamid Parvin, Behrouz Minaei-Bidgoli, and Hamideh Shahpar

School of Computer Engineering, Iran University of Science and Technology (IUST),
Tehran, Iran

{parvin,b_minaei,shahpar}@iust.ac.ir

Abstract. This paper proposes an innovative combinational algorithm for improving the performance of classifier ensembles both in stabilities of their results and in their accuracies. The proposed method uses bagging and boosting as the generators of base classifiers. Base classifiers are kept fixed as decision trees during the creation of the ensemble. Then we partition the classifiers using a clustering algorithm. After that by selecting one classifier per each cluster, we produce the final ensemble. The weighted majority vote is taken as consensus function of the ensemble. We evaluate our framework on some real datasets of UCI repository and the results show effectiveness of the algorithm comparing with the original bagging and boosting algorithms.

Keywords: Decision Tree, Classifier Ensembles, Bagging, AdaBoosting.

1 Introduction

Although the more accurate classifier leads to a better performance, there is another approach to use many inaccurate classifiers specialized for a few data in the different problem spaces and using their consensus vote as the classifier. This can lead to a better performance due to the reinforcement of the consensus classifier in the error-prone feature spaces. In General, it is ever-true sentence that combining diverse classifiers usually results in a better classification [5].

This paper proposes a framework for development of combinational classifiers. In this new framework, a number of train data-bags are first bootstrapped from train data-set. Then a pool of weak base classifiers is created; each classifier is trained on one distinct data-bag. After that to get rid of similar base classifiers of the ensemble, using a clustering algorithm, here fuzzy k-means, the classifiers are partitioned. The partitioning is done considering the outputs of classifiers on train data-set as feature space. In each partition, one classifier, the head of cluster, is selected to participate in final ensemble. Then, to produce consensus vote, different votes (or outputs) are gathered out of ensemble. After that the weighted majority voting algorithm is applied over them. The weights are determined using the accuracies of the base classifiers on train dataset.

Decision Tree (DT) is one of the most versatile classifiers in the machine learning field. DT is considered as one of the unstable classifiers that can produce different

results in its successive trainings on the same condition. It uses a tree-like graph or model of decisions. The kind of representation is appropriate for experts to understand what classifier does [8]. Its intrinsic instability can be employed as a source of diversity in classifier ensemble. The ensemble of a number of DTs is a well-known algorithm called Random Forest (RF) which is one of the most powerful ensemble algorithms. The algorithm of Random Forest was first developed by Breiman [2]. In this paper, DT is totally used as base classifier.

Rest of this paper is organized as follows. Section 2 is related works. In section 3, we explain the proposed method. Section 4 demonstrates results of our proposed method against traditional ones comparatively. Finally, we conclude in section 5.

2 Related Work

Generally, there are two important challenging approaches to combine a number of classifiers that use different train sets. They are Bagging and Boosting. Both of them are considered as two methods that are sources of diversity generation.

The term Bagging is first used by [2] abbreviating for Bootstrap AGGREGatING. The idea of Bagging is simple and interesting: the ensemble is made of classifiers built on bootstrap copies of the train set. Using different train sets, the needed diversity for ensemble is obtained.

Breiman [3] proposes a variant of Bagging which it is called Random Forest. Random Forest is a general class of ensemble building methods using a decision tree as the base classifier. To be labeled a "Random Forest", an ensemble of decision trees should be built by generating independent identically distributed random vectors and use each vector to grow a decision tree. In this paper Random Forest algorithm which is one of the well known versions of Bagging classifier [6] is implemented and compared with the proposed method.

Boosting is inspired by an online learning algorithm called Hedge(β) [4]. This algorithm allocates weights to a set of strategies used to predict the outcome of a certain event. At this point we shall relate Hedge(β) to the classifier combination problem. Boosting is defined in [4] as related to the "general problem of producing a very accurate prediction rule by combining rough and moderately inaccurate rules of thumb." The main boosting idea is to develop the classifier team D incrementally, adding one classifier at a time. The classifier that joins the ensemble at step k is trained on a dataset selectively sampled from the train dataset Z . The sampling distribution starts from uniform, and progresses towards increasing the likelihood of "difficult" data points. Thus the distribution is updated at each step, increasing the likelihood of the objects misclassified at step $k-1$. Here the correspondence with Hedge(β) is transposed. The classifiers in D are the trials or events, and the data points in Z are the strategies whose probability distribution we update at each step. The algorithm is called AdaBoost which comes from ADaptive BOOSTing. Another version of these algorithms is arc-x4 which performs as a newer version of ADABOOST [6].

3 Proposed Framework

The main idea behind the proposed method is to use the most diverse set of classifiers obtained by Bagging or Boosting mechanism. Indeed a number of classifiers are first trained by the two well-known mechanisms: Bagging or Boosting. After that the produced classifiers partitioned according their outputs. Then a random classifier is selected from each of the produced clusters. Since each cluster is produced according to classifiers' outputs, it is highly likely that selecting one classifier from each cluster, and using them as an ensemble can produce a diverse ensemble that outperforms the traditional Bagging and Boosting, i.e. usage of all classifiers as an ensemble. Fig. 1 depicts the training phase of the Bagging method generally.

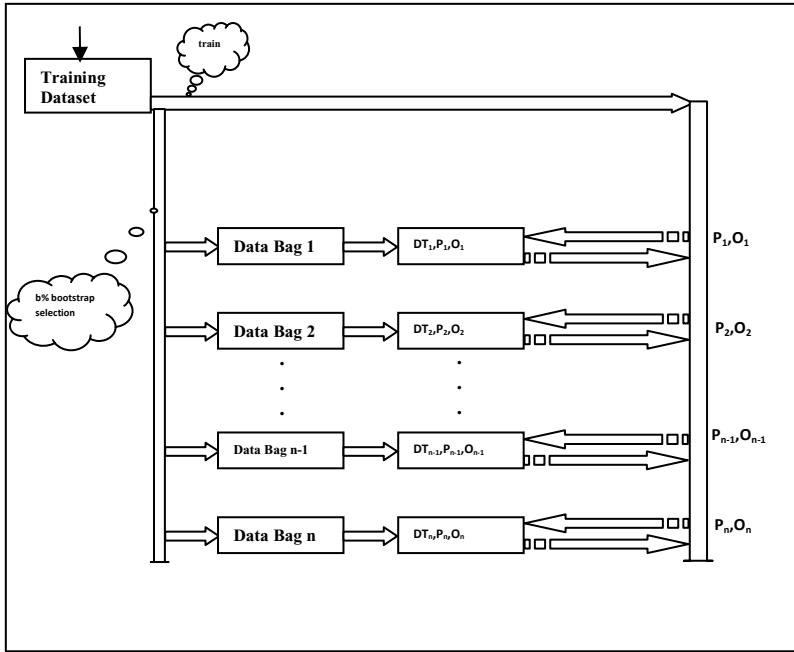


Fig. 1. Training phase of the Bagging method

As it is obvious from the Fig. 1, we bootstrap n subsets of dataset with b percent of the train dataset. Then a decision tree is trained on each of those subsets. We also then test each decision tree over the whole of train dataset and calculate its accuracy. The output of i th decision tree over train dataset is denoted by O_i and its accuracy is denoted by P_i . Fig. 2 depicts the training phase of the Boosting method. We again select a subset of dataset containing b percent of train dataset. Then the first decision tree is trained on this subset. After that the first classifier is tested on the whole train dataset which this results in producing the O_1 and P_1 . Using O_1 , the next subset of b percent of train dataset is obtained. This mechanism is continued in such a way that obtaining i th subset of b percent of train dataset is produced considering the $O_1, O_2, \dots, O_{i-1}$. For more information about the mechanism of Boosting, the reader can refer to Kuncheva [6].

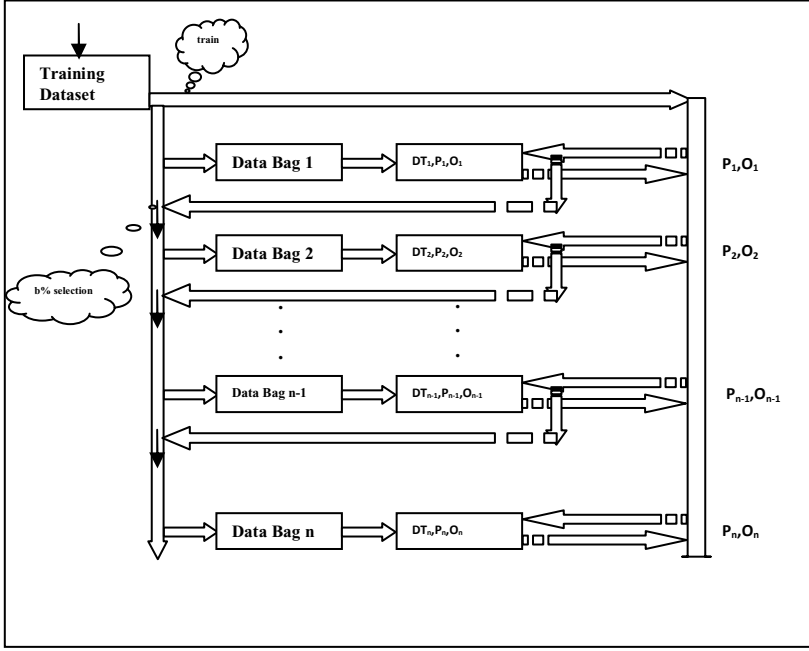


Fig. 2. Training phase of the Boosting method

The proposed method is generally illustrated in the Fig. 3. In the proposed method we first produce a dataset whose i th dataitem is O_i . Features of this dataset are real dataitems of under-learning dataset. Then we have a new dataset having n classifiers and N features, where n is a predefined value showing the number of classifiers produced by Bagging or Boosting and N is the cardinality of under-learning datasets. After producing the mentioned dataset, we partition that dataset by use of the clustering algorithm which results in some clusters of classifiers. Each of the classifiers of a cluster has similar outputs on the train dataset; it means these classifiers have low diversities, so it is better to use one of them in the final ensemble rather than all of them. For escaping from outlier classifiers, we ignore from the clusters which contain number of classifiers smaller than a threshold.

Let us assume that E is the ensemble of n classifiers $\{DT_1, DT_2, DT_3 \dots DT_n\}$. Also assume that there are m classes in the case. Next, assume applying the ensemble over data sample d results in a binary D matrix like equation 1.

$$D = \begin{bmatrix} d_{1 \ 1} & d_{1 \ 2} & \cdot & d_{1 \ n} \\ \cdot & \cdot & \cdot & \cdot \\ d_{m-1 \ 1} & d_{m-1 \ 2} & \cdot & d_{m-1 \ n} \\ d_{m \ 1} & d_{m \ 2} & \cdot & d_{m \ n} \end{bmatrix} \quad (1)$$

where $d_{i,j}$ is one if classifier j votes that data sample d belongs to class i . Otherwise it is equal to zero. Now the ensemble decides the data sample d to belong to class q according to equation 2.

$$q = \arg \max_{i=1}^m \left| \sum_{j=1}^n w_j * d_{i,j} \right| \quad (2)$$

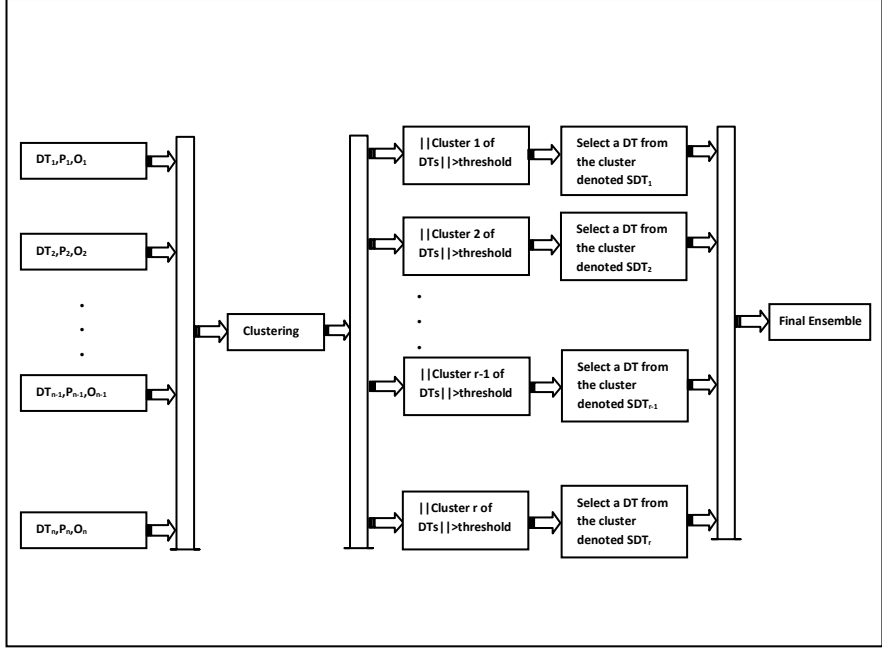


Fig. 3. Proposed method for selecting the final ensemble from a pool of classifier generated by Bagging or Boosting

where w_j is the weight of classifier j which is obtained optimally according to equation 3 [6].

$$w_j = \log \frac{p_j}{1 - p_j} \quad (3)$$

where p_j is accuracy of classifier j over total train set. Note that a tie breaks randomly in equation 2.

4 Experimental Results

This section evaluates the result of applying proposed algorithm on some real datasets available at USI repository [1] and one hand made dataset named half-ring. The details of half-ring dataset can be available in [7]. These dataset are summarized in the Table 1.

Table 1. Details of used dataset

Dataset Name	# of dataitems	# of features	# of classes
breast cancer	404	9	2
bupa	345	6	2
glass	214	9	6
galaxy	323	4	7
half-ring	400	2	2
heart	462	9	2
ionosphere	351	34	2
iris	150	4	3
test monk1	412	6	2
test monk2	412	6	2
test monk3	412	6	2
train monk1	124	6	2
train monk2	124	6	2
train monk3	122	6	2
wine	178	13	3

Measure of decision in each employed decision tree is taken as gini measure. The threshold of pruning is set to 2. Also the classifiers' parameters are fixed in all of their usages.

In all experiments n , r , b and threshold of accepting a cluster are set to 151, 11, 30 and 2 (i.e. only the clusters with one classifier is dropped down) respectively. All the experiments are done using 4-fold cross validation. Clustering is done by fuzzy kmeans with r (11) clusters. Table 2 shows the accuracies of different methods.

Table 2. Comparison of the results. * shows the dataset is normalized, and 4 fold cross validation is taken for performance evaluation. ** shows that the train and test sets are predefined.

	Arc-X4	Random Forest	Proposed Random Forest	Proposed Arc-X4
breast cancer*	96.18	95	96.47	95
bupa*	71.51	68.31	72.97	66.28
glass*	65.09	62.26	66.23	60.85
galaxy*	70.94	69.06	72.5	67.5
half-ring*	97.25	95.75	97.25	95.75
heart*	70.87	68.26	72.61	68.26
ionosphere*	92.24	90.52	91.54	90.52
iris*	95.95	95.95	95.95	95.27
monk problem1**	98.11	97.49	98.76	97.37
monk problem2**	97.01	86.64	97.62	86.73
monk problem3**	87.29	96.92	87.34	96.34
wine*	96.59	92.61	98.3	93.18
Average	86.59	84.9	87.3	84.42

While we choose only at most 7.3 percent of the base classifiers of Random Forest, the accuracy of their ensemble outperforms the full ensemble of them, i.e. Bagging. Also it outperforms Boosting.

Because the classifiers selected in this manner (by Bagging along with clustering), have different outputs, i.e. they are as diverse as possible, they are more suitable than their all ensemble. It is worthy to mention that the Boosting is inherently diverse enough to be an ensemble totally; and the reduction of ensemble size by clustering destructs their Boosting effect. Take it in the consideration that in Boosting ensemble, each member covers the drawbacks of the previous ones.

5 Conclusion and Future Works

In this paper, we have proposed a new method to improve the performance of classification. The proposed method uses Bagging as generator of the base classifiers. Then using fuzzy kmeans we partition the classifiers. After that we select one classifier per a validated cluster.

While we choose only at most 7.3 percent of the base classifiers of Bagging, the accuracy of their ensemble outperforms the full ensemble of them. Also it outperforms Boosting.

As a future work, we can turn to research on the variance of the method. Since it is said about Bagging can reduce variance and Boosting can simultaneously reduce variance and error rate.

References

1. Blake, C.L., Merz, C.J.: UCI Repository of machine learning databases (1998), <http://www.ics.uci.edu/~mllearn/MLRepository.html>
2. Breiman, L.: Bagging Predictors. *Journal of Machine Learning* 24(2), 123–140 (1996)
3. Breiman, L.: Random Forests. *Machine Learning* 45(1), 5–32 (2001)
4. Freund, Y., Schapire, R.E.: A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting. *J. Comput. Syst. Sci.* 55(1), 119–139 (1997)
5. Gunter, S., Bunke, H.: Creation of classifier ensembles for handwritten word recognition using feature selection algorithms. *IWFHR* (2002)
6. Kuncheva, L.I.: *Combining Pattern Classifiers, Methods and Algorithms*. Wiley, New York (2005)
7. Minaei-Bidgoli, B., Topchy, A.P., Punch, W.F.: Ensembles of Partitions via Data Resampling. In: *ITCC*, pp. 188–192 (2004)
8. Yang, T.: Computational Verb Decision Trees. *International Journal of Computational Cognition*, 34–46 (2006)

Ensemble of Classifiers Based on Hard Instances

Isis Bonet, Abdel Rodríguez, Ricardo Grau, and María M. García

Center of Studies on Informatics, Central University of Las Villas, Santa Clara, Cuba
{isisb, abdelr, rgrau, mmgarcia}@uclv.edu.cu

Abstract. There are several classification problems, which are difficult to solve using a single classifier because of the complexity of the decision boundary. Whereas, a wide variety of multiple classifier systems have been built with the purpose of improving the recognition process. There is no universal method performing the best. The aim of this paper is to show another model of combining classifiers. This model is based on the use of different classifier models. It makes clusters to divide the dataset, taking into account the performance of the base classifiers. The system learns how to decide from the groups, by a meta-classifier, who are the best classifiers for a given pattern. In order to compare the new model with well-known classifier ensembles, we carried out experiments with some international databases. The results demonstrate that this new model can achieve similar or better performance than the classic ensembles.

Keywords: multiple classifiers, ensemble classifiers, classification, pattern recognition.

1 Introduction

Supervised learning is a common task in pattern recognition and machine learning, but not completely solved. A lot of classification methods have been built, however, none of them have proven to perform the best. In fact, a research area has been created to develop methods for combining classifiers, which goal is to fuse different classifiers, hoping that the performance reached will be better than the single classifiers.

There are some reasons to justify the use of classifier ensembles, but the more general justification is when we are dealing with a dataset which decision boundary is too complex to be learned by a single classifier [1]. This also seems to happen in too large or too short data sets. A database with these characteristics is also sensitive to expect low performance levels with a single classifier. Multiple classifier systems precisely arise from these problems in several tasks to combine different classification models. Ideally, it should choose the appropriate base classifier for each case of the dataset.

The design of classifier ensembles can be divided in three important steps. The first one is the definition of the topology: the distribution and connection of components. The second step, very related with the first one, is the choice of the base classifiers,

which could be the key of success depending on the classifier diversity, a subject still open to research. The last step is the combination of classifier decisions, divided in two main strategies: selection and fusion.

There is a wide variety of these methods reported in the literature; indeed, some authors have made a good review and comparison of them [1], [2], [3], [4]. Even though there exist many classifier ensembles, the most popular ensembles are: Bagging [5], Boosting [6] and Stacking [7].

Bagging and Boosting are based on the manipulation of the training base to achieve the diversity of base classifiers. The diversity is obtained by using bootstrapped replicas of the training data. All classifiers are instances of the same model and the output combination is obtained by the majority vote.

Boosting is similar to Bagging, but once a classifier is trained, it uses its mistakes to weight the samples for the training set of the next classifier, reinforcing misclassified samples. AdaBoost [8] is a more general version of this algorithm and actually more known than Boosting. In this paper when we refer to Boosting we are actually talking about this version: AdaBoost.

On the other hand, Stacking uses different classification models with the same training sets and has a second classifier level provided for a meta-classifier that works as the outputs combiner. This combiner is trained on a data set obtained from the relation between the classifiers outputs and the real value of the objective feature for each pattern.

In short, we can divide the classifier ensembles according to if they use the same classifier model and only change the parameters for training: datasets, features, model parameters; or if they use different classifier models. Bagging and Boosting are examples of using the same model, whereas Stacking is an example of the second type of ensembles.

Despite the classifier ensembles have been successfully applied in complex real-world pattern recognition problems, some complex real-problems are not solved and the searching for new models of combined classifiers is still an open research challenge.

The aim of this paper is to present a new way of combining classifiers based on the ensemble with different classifier models. The research is focused on the topology of the model and the learning process, not on how to guarantee diversity. In order to test the presented model we carried out experiments on 34 data sets and also a comparison is shown with some popular classifier ensembles.

2 Ensemble of Classifiers Using a Meta-classifier by Clusters of Hard Instances

The proposed model in this paper is based on the philosophy of stacking. It uses different classification models and a meta-classifier to combine the decisions, as Stacking does. Moreover the training data to the meta-classifier is built from the outputs of the base classifiers. The principal difference is on the way to train the meta-classifier. In stacking the original representation of patterns changes, now the features are the classifiers, where the values are the outputs of them, keeping the same objective feature. On the other hand, the proposed model also build a database to train

the meta-classifier, but the patterns keep the same features of the original data and the objective feature now represent the classifier that provides good classification for the pattern.

In order to give an insight into the workings of the proposed model its different parts will be defined below.

2.1 Topology

The general objective function is defined for classification problems (equation 1).

$$FO: RD \rightarrow O \quad (1)$$

Where RD is the features set and $O=\{1,2,\dots,N\}$ is the set of N possible values of the objective feature (classes).

For a given classification problem the use of different classifier models can provide different decision boundaries. Hence, the adequate selection of classifiers is the key for the success [9].

Once the selection of potential classifiers is done, the question is: “Which of them are the best for classifying a given pattern?” That process can be very complex because each classifier can define a different decision boundary to separate the classes. This can result in different outputs from the classifiers for a given instance. What classifier should we select? Ideally, we know what each classifier has learned. To take into account this idea, the model developed in this work is based on the correct classified patterns for each classifier, and the misclassified patterns will be dismissed, they will produce an error anyway.

Now the problem is to define the method for combining the outputs, but in this model this is also related with the design of the topology. In order to do it, we consider the best classifiers of the ensemble for a given pattern by associating them with the group of correctly classified patterns. Once the classifiers are trained, each one will have a group of correct patterns associated, that we will call “hard instances”. An output combination could be useful to find what patterns each classifier has learned. Afterwards, we could separate the data instances in clusters according to the classifier outputs; suppose we have C base classifiers, so we would be in presence of C clusters.

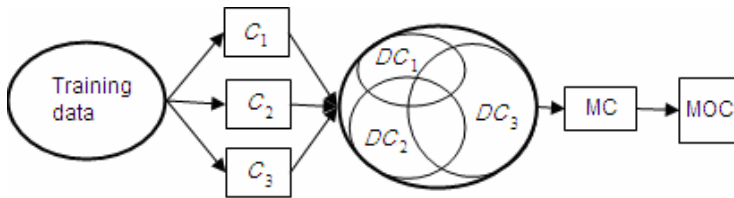


Fig. 1. Example of the multiclassifier using three base classifier models

Figure 1 depicts an example of the proposed model using $C=3$ base classifiers, where classifiers C_1 , C_2 , and C_3 are trained from the training data. Afterwards, as it can be seen in the figure, the result can be interpreted as a clustering process of the

dataset. Each classifier will have a subset of *hard instances*, these are represented by circle inside the data (DC_1, DC_2, DC_3), to give a better understanding of the problem. Usually these subsets have common patterns.

The figure shows a subset of patterns which are out of the three clusters; these are misclassified by all classifiers. There is no way to combine the base classifiers in order to obtain another result for the last set of misclassified patterns. Hence, those patterns can be excluded of the training data for the meta-classifier.

This system is based on adding a meta-classifier that learns how to separate the patterns according to the C groups. In many data sets we will have a high significance percentage of patterns that can be easily classified by any classifier, and in this case, we would have this patterns repeated in all C clusters, that means that the region in the interception of all clusters of hard instances can be too big. This is another patterns subset to be taken into account. In the figure we can see it in the interception of the three circles; this represents the patterns that are correctly classified by all classifiers in the model. Then we create a $C+1$ cluster with the patterns in the interception of all previous ones in order to reduce the noise caused by repeated patterns in all groups. The creation of this new group does not eliminate all repeated cases associated to different classes because we also have the interception of clusters by pairs, but these repeated instances should not be so significant. Anyway, it is important to keep in mind selecting a meta-classifier that can use a continuous valued output, in order to use the probability associated to every class (in this case the class represents the classifiers) as the measure to combine the outputs of the different classifiers.

In short, in order to balance the classes in the new database, we select a threshold to decide when the case is kept in the $C+1$ group or added to any other group. Usually, when the classifiers produce an outputs' vector, a threshold is establish to decide what the resulting class is. In the case of two classes this threshold is 0.5 by default. The variety of this threshold can drastically change our clusters, but also can redistribute better the patterns in the different sets. Of course, this threshold depends on the base, for this reason it must be considered as a parameter of the model.

We build a new dataset from the formed clusters of *hard instances* to train the meta-classifier. The difference between the objective function for the base classifier and the meta-classifier is given by the objective feature. Given a problem which training data has N possible values the objective feature is $O=\{1,2,\dots,N\}$ for the base classifiers. The objective feature for the meta-classifier is $O=\{1,2,\dots,C+1\}$. Notice that now the objective feature is composed by values corresponding with C classifiers and an extra $C+1$ that is the subset of patterns that are correctly classified by all base classifiers.

We can find some problems in the new database, created for the meta-classifier. A classifier has a small cluster of correct classified instances, resulting in unbalanced groups. Sometimes that problem can be solved selecting other base classifiers or changing the threshold value.

Another problem is found when a large number of examples are misclassified by all base classifiers. In that case, the meta-classifier does not have any possibility to change the results, for this reason these cases are not important for our model.

Some authors have developed many diversity measures that help us to select base classifiers [10], [11], [12]. Nevertheless, in this paper we do not pretend to optimize the search of the classifiers, just to propose a different paradigm, as a different way for finding the decision regions.

Ideally, for an specific data we may select the appropriate classifiers using any diversity measure [10].

2.2 Combining Outputs

In this work we propose two ways to combine the outputs, one based on selection and another based on fusion.

The combination by selection is the simplest, consisting on the selection of the classifier with the bigger probability, taking into account the outputs vector of meta-classifier. It can also be seen as an alternative approach to the classifier combination, called dynamic classifier selection [13].

The fusion combination is based on weighting the outputs of the classifiers with the output of the meta-classifier. The meta-classifier is trained with this new database, explained in the topic 2.1, so it will have a vector of probabilities associated to the clusters as output. Then, the module of outputs combination (MOC) takes it to weigh the classifiers, as it is shown in the equation 2.

$$S = SC * SMC \quad (2)$$

Where, SC is a matrix $C \times N$, C is the number of classifiers and N the amount of classes. SC is the matrix of classifier outputs, that is, $SC(i, j)$ is the probability given by the classifier i for class j . SMC is the probabilities' vector given by the meta-classifier. S is the vector of probabilities, with dimension N , which results from the system.

The proposed method will be called "Multi-Expert by Hard Instances" (MEHI). It was implemented by using Weka (created in the University of Weikato in New Zealand) [14]. This software offers a great number of classification methods and facilities to add new models and to compare with others.

3 Result and Discussion

To validate this new model, 34 bases of general classification problems were chosen from the UCI Repository [15]. The databases are miscellaneous, 10 of them have discrete features, 11 have continuous features and 13 have mixed features. With respect to the objective feature they also present diversity, the amount of value ranges is between 2 and 24. Also the amount of samples varies between 23 and 3772.

Several topologies were used in this work to choose the base classifiers and the meta-classifier. Finally, 3 base classifiers were selected: decision tree (J48), Bayesian Network (BayesNet) and Support Vector Machine (SVM). Multilayer Perceptron (MLP) was used as meta-classifier. A study of the threshold was made to build the clusters by classifiers in correspondence with each base, these thresholds oscillated between 0.6 and 0.9. We used 10-fold cross-validation, in order to validate the results and to compare them with other methods.

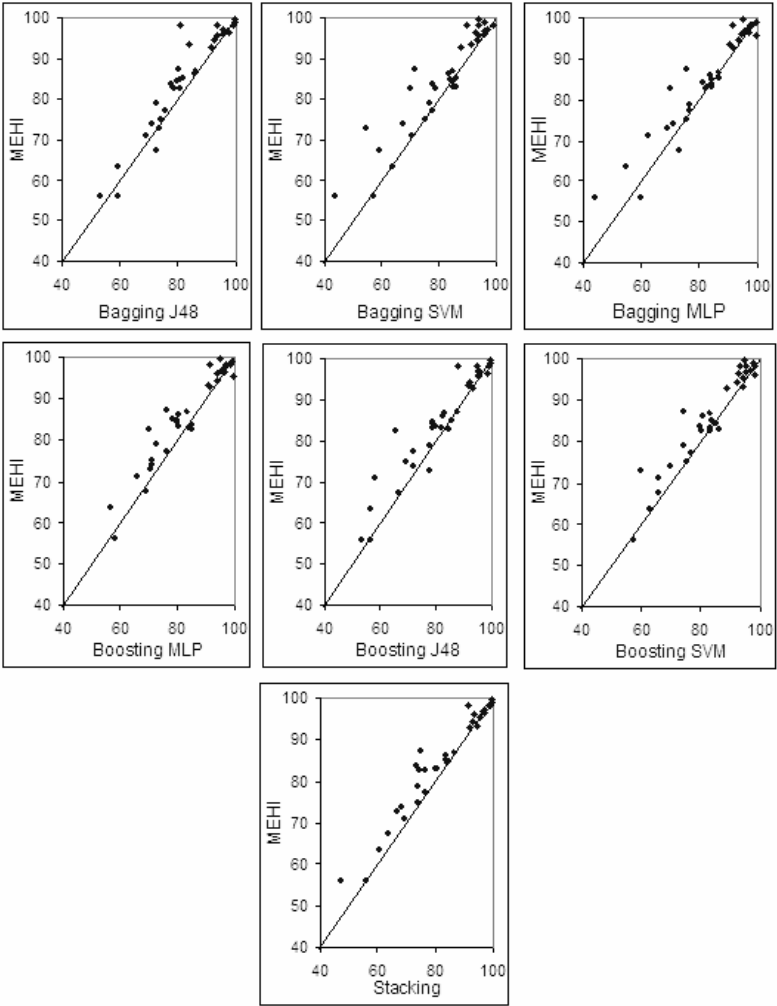


Fig. 2. Comparison of the results from the 11 databases using Bagging, Boosting, Stacking and the new model: MEHI

To validate the new method, Bagging, AdaBoost and Stacking were chosen. In the case of Bagging and AdaBoost they were tested with three base classifiers: J48, SVM and MLP. Bayesian Network was not used because of the stability of this method. On the other hand, Stacking was tested with the same classifier models used in the proposed method.

The results are shown in figure 2 based on the accuracy of each method. Each graph represents the comparison between MEHI and each of the ensembles selected: Bagging, Boosting and Stacking. Each dot represents a database. If the point lies above the line means that MEHI achieved higher accuracy than another ensemble.

The three graphs in the top of the figure show the results of MEHI versus Bagging with different models. The first one is with Bagging using J48 as base classifier, the second one using SVM and the last one using MLP. In the middle of the figure the results of Boosting are shown in the same way. In both cases the results are similar; the proposed model is superior in most of the databases. But, it is also true that many of them are on the line or close to it, this means that Bagging and Boosting can be better for some databases.

The comparison with Stacking is in the bottom of the figure, showing again that MEHI has better results.

Both ways of combination: selection and fusion were used in the experiments. The results shown in figure 2 are related with de combination by fusion. The performance obtained using combination by selection, is similar, although a little lower, than the combination by fusion shown above. For this reason we select combination by fusion to show in this experiment.

Although we do not intend to propose our method as the best ensemble for classification database, we suggest it as an alternative model to be taken into account when the ensembles are needed to solve problems of complex decision boundaries.

In general, the results show that in most of the bases the new model are superior or at least similar to those obtained by the other classifier ensembles.

A statistical study with Wilcoxon signed rank test could conclude that the differences in performance are significant in all cases, with MEHI ahead.

4 Conclusions

In this work, a model of combination of classifiers based on their local performance was designed and implemented. This result was obtained by a meta-classifier that learns how to separate the correctly classified instances by each base model. This method was compared with well-known classifier ensembles and its accuracy was significantly superior for a sample of 34 international databases of UCI Repository. Therefore, we conclude that the proposed classifier ensemble improves the results of accuracy and should be tested in real-world databases. It is important to remark that the methods do not replace none other ensemble, it is just another way to combine multiple classifiers.

References

1. Polikar, R.: Ensemble based systems in decision making. *IEEE Circuits and Systems Magazine* 6, 21–44 (2006)
2. Dietterich, T.G.: Ensemble methods in machine learning. In: Kittler, J., Roli, F. (eds.) *MCS 2000. LNCS*, vol. 1857, pp. 1–15. Springer, Heidelberg (2000)
3. Ghosh, J.: Multiclassifier systems: Back to the future. In: Roli, F., Kittler, J. (eds.) *MCS 2002. LNCS*, vol. 2364, pp. 1–15. Springer, Heidelberg (2002)
4. Kuncheva, L.I.: *Combining Pattern Classifiers, Methods and Algorithms*. Wiley Interscience, New York (2004)
5. Breiman, L.: Bagging predictors. *Machine Learning* 24, 123–140 (1996)
6. Schapire, R.E.: The strength of weak learnability. *Machine Learning* 5, 197–227 (1990)

7. Wolpert, D.: Stacked generalization. *Neural Networks* 5, 241–259 (1992)
8. Freund, Y., Schapire, R.E.: Decision-theoretic generalization of on-line learning and an application to boosting. *Journal of Computer and System Sciences* 55, 119–139 (1997)
9. Canuto, A.M.P., Abreu, M.C.C., Oliveira, L.D., Xavier, J.C., Santos, A.D.: Investigating the influence of the choice of the ensemble members in accuracy and diversity of selection-based and fusion-based methods for ensembles. *Pattern Recognition Letters* 28, 472–486 (2007)
10. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning* 51, 181–207 (2003)
11. Banfield, R.E., Hall, L.O., Bowyer, K.W., Kegelmeyer, W.P.: Ensemble diversity measures and their application to thinning. *Information Fusion* 6, 49–62 (2005)
12. Tang, E.K., Suganthan, P.N., Yao, X.: An analysis of diversity measures. *Machine Learning* 65, 247–271 (2006)
13. Giacinto, G., Roli, F.: Dynamic classifier selection. In: 1st International Workshop on Multiple Classifier Systems, pp. 177–189 (Year)
14. Witten, I., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*. Diane Cerra, San Francisco (2005)
15. University of California, Irvine, School of Information and Computer Sciences, <http://www.ics.uci.edu/~mlearn/MLRepository.html>

Scalable Pattern Search Analysis

Eric Sadit Tellez, Edgar Chavez, and Mario Graff

Universidad Michoacana de San Nicolas de Hidalgo, México

{sadic, mgrafig}@lsc.fie.umich.mx,
elchavez@fismat.umich.mx

Abstract. Efficiently searching for patterns in very large collections of objects is a very active area of research. Over the last few years a number of indexes have been proposed to speed up the searching procedure. In this paper, we introduce a novel framework (**the K-nearest references**) in which several approximate proximity indexes can be analyzed and understood. The search spaces where the analyzed indexes work span from vector spaces, general metric spaces up to general similarity spaces.

The proposed framework clarify the principles behind the searching complexity and allows us to propose a number of novel indexes with high recall rate, low search time, and a linear storage requirement as salient characteristics.

1 Introduction

Proximity search, often generalized as similarity search, is present in many fields of computer science such as: pattern recognition, textual and multimedia information retrieval, query by content and classification, machine learning, lossless/lossy data streaming and compression, security (e.g. criminal record databases, biometric identification, biometric authentication, etc.) and bioinformatics, followed by a very large etcetera.

We need a formalization of the problem to continue the discussion. A metric workload is a triple (U, d, S) with U a domain, $S \subseteq_F U$ a finite subset of U , named the database and $d : U \times U \rightarrow \mathbb{R}^+$ a distance function. The distance function obeys the following properties $\forall x, y, z \in U$ (i) $d(x, y) \geq 0$ and $d(x, y) = 0 \iff x = y$, (ii) $d(x, y) = d(y, x)$, and (iii) $d(x, z) + d(z, y) \geq d(x, y)$. These properties are known as strict positiveness, symmetry, and the triangle inequality, respectively.

The two most common search queries in S are *range* and *nearest neighbor* (NN) queries. This paper focus on a natural extension of the later, the *k*-Nearest-Neighbors (kNN). Informally speaking the $\text{kNN}_d(q, S)$ retrieves the k closer objects to the query q in the database S .

Searching kNN have a well known linear worst case [1][2][3]. This problem gets worst in the case of high intrinsic dimensional dataset. Under this circumstance, traditional indexing techniques using the triangle inequality such as the families of indexes: *compact partition* and *pivot based* suffer from a condition known as the *curse of the dimensionality* (CoD) [2]. The problem comes from the phenomenon of *concentration of the measure*, informally characterized by an histogram of distances with small standard deviation and a large mean. We are not aware of any exact index able to deal with the curse of dimensionality.

A common approach to alleviate the CoD is to use *approximate* proximity search algorithms trading speed for accuracy. One way is to convert an exact algorithm to an approximate one using the procedure described in [4] which consists in aggressively reduce the radius by a stretching constant.

In this work, we present a new framework where a number of approximate indexes can be analyzed and understood. Our framework is simple, yet powerful enough to analyze different approximate indexes that work on different spaces such as: vector spaces, general metric spaces, and similarity spaces.

Furthermore, the simplicity of our approach allow us to propose novel approximate indexes that have a high recall rate, low search time, and a linear storage requirement, among other characteristics. As we will see, these novel indexes are competitive to the point that in the majority of cases tested they have a better performance¹ than previous approaches.

The rest of the paper is organized as follows. In Section 2 we introduce our framework and describe those indexes that can be analyzed under this framework. Section 3 shows the experimental results. A number of conclusions and possible directions for future work are described in Section 4.

2 K Nearest References

Our framework is composed by two functions: an encoding function and a similarity function. Each object u in the database is encoded using its K closer objects from a set of references R , i.e., $\mathbf{KNN}_d(u, R)$ where R is a small subset of objects randomly selected from the database. We call this representation the K Nearest References (**KNR**). Proximity between objects is approximate by proximity between encodings of the objects using a similarity function.

Let **encode** be an encoding function, the domain U is converted to the **KNR** space as: $\hat{U} = \{\hat{u} \leftarrow \mathbf{encode}(\mathbf{KNN}_d(u, R)) : u \in U\}$. Similarly $\hat{S}$ denote the encoded database.

Each object $r_j \in R$ can be unequivocally identified using j . We will use j and r_j without distinction when there is no confusion. Also, for notation convenience, we define $\mathbf{KNR}(u) = \hat{u}$. In general, the **KNR** sequences can be encoded as vectors, strings or sets. Each encoding gives a tradeoff between space, search time and accuracy. To fix ideas, think on a string representation and the amount of space needed for the whole database S . We will need at most $O(nK \log |R|)$ bits to store $\hat{S}$.

To complete our framework we need a distance function $d' : \hat{U} \times \hat{U} \rightarrow \mathbb{R}^+$. This function accepts two **KNR** sequences encoded as $\hat{u}$ and $\hat{v}$ and if the corresponding objects are close under d , then the encoded sequences will be close under d' .

To search with a **KNR** method we complete three steps: a) map the query to the **KNR** domain, b) search for the closest γ candidates under d' in the mapped space, and finally, c) verify the candidates using the original distance. Typically, γ is a tuning parameter that optimize the desired performance. That is, it is optimized to a particular data set and **kNN** queries in order to maximize performance.

¹ Here performance is measured as the tradeoff among storage, recall and search time.

2.1 Describing Existing Proximity Indexes Using KNR

The first use of our framework is in describing and analyzing existing indexes. We are aware of four proximity indexes that can fit under our framework. In the next paragraphs, we briefly describe them.

Permutation Based Index (PBI) is the first KNR index reported in the literature (see [5]). The idea behind this is to describe an object by capturing the perspective of it. That is, it measures how a set of references (R), called permutants, are seen by each object in the database. The proximity is computed using the relative movements of the permutants. We must notice that $K = |R|$ in PBI. Here $|R|$ is very small, this makes it particularly efficient for expensive distances.

The index uses the inverse of the permutation to create a vector space as follows: given a $\mathbf{KNR}(u) = x_1x_2 \cdots x_{|R|}$ where $x_i \in R$. We define $\hat{u}$ as the permutation of u . The inverse of $\hat{u}$ is defined as $\hat{u}^{-1}[x_i] = i$, and they are measured with Minkowski's L_1 or L_2 distances (i.e. Spearman Footrule and Spearman ρ , respectively) [5]. Please note that each permutation requires $|R|$ distances, a linear sort $O(|R|)$ [8] and a linear pass to find the inverse. In order to find the candidate list, one needs to perform n permutation distance comparisons (L_1 or L_2) each one of them requires $O(|R|)$ basic arithmetic operations. This yields to a total cost of $O(n|R|)$ basic operations. The space complexity is $n|R| \log |R|$ bits.

Metric Inverted File. It is based on a simplification of the Spearman Footrule [5,6], using only the first K references closer to an object and its positions in the full permutation. This information is used to compute approximately the Spearman Footrule (L_1) distance of the permutations. Indexes are small and fast since $K \ll |R|$.

Since many permutants cannot be found to compute L_1 , the blank positions should be filled with a penalization constant ω (e.g. $\omega = |R|/2$). To provide a scalable representation, the authors use an inverted file to represent R (as the *thesaurus*), and list of tuples (*object, position*) as *posting lists* [6]. A detailed explanation about inverted indexes can be found in [7,8].

The computational cost to represent each object is equivalent to the permutation based index; however, in this case the plain mapping (without inverted index representation) requires $Kn \log K |R|$ bits i.e., each object requires K tuples of (*referenceId, position*). These tuples are sorted by *referenceId* (adding an additional sort over K items). The candidate list requires n evaluation of the partial Spearman Footrule distance which requires in the worst case $O(K)$ and in the best scenario $O(1)$. Please note that when the objects are not related the best scenario (or something close to it) is frequently found. This yields to a worst case of $O(nK)$ and a smaller average worst case for obtaining the candidate list. Using an inverted index, requires $Kn \log(Kn)$ bits of space, and the total cost is driven by the cost to obtain the candidate list and $O(\gamma)$ distance computations.

The Brief Permutation Index. As the original PBI, the brief permutation index [9] uses $K = |R|$, but it speeds up searches, in total time, while reducing the space needed

² Another option is Kendall τ [5], but its usage it is limited by the high cost of the computation.

³ In general, if distances cannot be discretized, we need a comparison based sorting, i.e. $O(|R| \log |R|)$.

for the index. The main idea behind the *brief index* is to lossy encode the inverse of the permutation with a single bit per permutant, using the information about how much it moves from its position in the identity permutation. The encoding yields a binary Hamming space of dimension $|R|$.

The codification of each object requires $O(|R|)$ distances plus $O(|R|)$ basic operations in the same way as the permutation based index. $|R|$ bits are needed to represent each object in the mapped space. The total space is then $n|R|$ bits. Assume that w is the size of the computer word then the computational cost of each distance d' is $O(\frac{|R|}{w/2})$ using an additional table of size $2^{w/2}$ to pre-compute hamming [4].

Prefix Permutations Index: PP-Index. The mapping used by the PP-Index [10] is as follows: for every object $u \in U$ we compute $KNR(u)$. The proximity between objects is measured by the length of the shared prefixes of the corresponding strings, larger shared prefixes means high proximity and short or zero length prefixes reflects low proximity. The strict notion of closeness yields to low recall (ranging from 0.3 to 0.5) [10]. In contrast, it is really fast and can be represented efficiently using a compressed trie data structure [7,8].

In order to overcome the low recall several strategies are possible, increasing search time and memory usage (see [10]). While some of these enriching strategies are common to all **KNR** indexes and we discuss and generalize these strategies in section 3.3 for all **KNR** methods. The basic implementation used in the PP-Index will be referred as **KNR** prefixes.

The computational cost of this index is similar to the metric inverted file explained above; however, the constants involved in this index are lower and the best scenario (for d') is even more frequent in the PP-Index than in the Metric Inverted File.

2.2 Using the KNR Framework to Create Proximity Indexes

Using the described framework, we create seven new indexes. We present them in a classification based on the mapped space, obtaining three main **KNR** mappings: vector spaces, strings, and sets. In general the computational cost of computing the mapping per object is composed by the number of distances to obtain the K nearest references, plus the cost of the **encode** function. The search process has a computational cost of n evaluations of d' (using the plain mapping without an index) plus at most γ evaluations of d . The storage requirements are linked to the particular **encode** function, as well the specific cost of d' .

Vector Space Mappings. In this class of mappings, the target is to obtain vectors from the sequences $KNR(u)$. One way to do so is by fixing the attention to the positions of the references in $KNR(u)$, or the actual distances to the references u_i . The permutation index [5], the brief index [9], and the metric inverted index [6] using the Spearman Footrule measure (L_1) are **KNR** vector space mappings. We add the vector space mapping for Spearman ρ (L_2), as a natural variation to L_1 . Another possibility is to use the cosine between the $|R|$ -dimensional **KNR** vectors having the distances to the K closer

⁴ When $w = 32$ this scheme produces tables of 2^{16} entries of 5 bits each one, i.e. $\log(w/2 + 1)$ bits in general. For very large w one needs to divide w in smaller pieces.

references. These methods would have the same computational cost than the Spearman Footrule. The space complexity is $nK(\log |R| + w)$ bits for the **KNR** cosine.

String Mappings. **KNR** sequences are strings in R^K (using R as an alphabet), where $K \ll |R|$. So, each $\hat{u}$ is used as a short string, then objects are compared using distances defined in the string domain. The main point here would be to measure the order of shared references, and how much effort it must be done to convert one string into another.

The PP-Index [10] is a string mapping. We augment the list showing the performance for Levenshtein (edit distance), longest common subsequence (*LCS*), both distances require $O(K^2)$ operations, for detailed description of the distances the reader should see [11]. The **KNR** string mappings uses only the information found by the **KNR** sequence, and a string distance function to predict proximity. The space complexity is $nK \log |R|$.

Set Mapping. Up to the best of our knowledge, there are no **KNR** indexes based on sets in the literature. Our idea is to use the **KNR** sequence as a set. We will keep the same notation for the **KNR**(u) and the encoded version $\hat{u}$. Since each reference can appear only once in **KNR**(u), the main difference with respect to string distances is the lack of order in the representation.

For set mappings we studied three distances, the Jaccard coefficient, the Dice coefficient, and the cardinality of the intersection as similarity measures. We found that set **KNR** mappings produces small and fast indexes with excellent recall.

The Jaccard distance is computed as $d_J(\hat{u}, \hat{v}) = 1 - \frac{|\hat{u} \cap \hat{v}|}{|\hat{u} \cup \hat{v}|}$, the distance gives values between $[0, 1]$ where 0 means equality and 1 means disjointness. The Dice coefficient, $d_D(\hat{u}, \hat{v}) = \frac{2|\hat{u} \cap \hat{v}|}{|\hat{u}| + |\hat{v}|}$, is used in many information retrieval tasks [12, 7]. For similarity functions, a zero value means no closeness.

The computational cost is in the same order than the metric inverted file, but simpler, since it does not requires additional operations than the union, i.e., the metric inverted file computes $O(K)$ arithmetic operations to partially compute the L_1 , then the constants involved are smaller for this case. The space complexity is smaller too, it is $O(nK \log(|R|))$ bits. As we experimentally prove in the experimental section below, these facts and a higher recall makes the set mappings a better option.

3 Experimental Results

In order to study the behavior of the different **KNR** mapping methods, we performed a series of experiments using as benchmarks three real datasets. The first one (**documents**) is a collection of 25157 short news articles in the *TFIDF* format from Wall Street Journal 1987 – 1989 files taken from TREC-3 collection. We use the angle between vectors as distance measure [7] and extracted 100 random documents from the collection as queries (note: these documents were not indexed). Each query searches for 30NN. The objects are vectors of thousand of coordinates. The second benchmark (**vectors**) is a set of 112544 color histograms (112-dimensional vectors) from an image database⁵. We choose randomly 200 histogram vectors and applied a *perturbation*

⁵ The original database source is <http://www.dbs.informatik.uni-muenchen.de/~seidl/DATA/histo112.112682.gz>

of ± 0.5 on one random coordinate. The search consists on finding 30NN under L_2 distance. The third one (**CoPhIR**) consists of 10 million of objects selected from the CoPhIR project [13]. Each object is a 208-dimensional vector and we use the L_1 distance. Each vector was created in a linear combination of five different MPEG7 vectors as described in [13]. We choose the first 200 vectors from the database as queries. Searches for the 30NN were performed.

The algorithms were implemented in the C# programming language, running under the Mono framework⁶. The experiments were performed in an Quadcore Intel Xeon 2.33 GHz workstation with 8GiB of RAM, running Ubuntu Linux 8.04. We kept the entire database and indexes in main memory and without exploiting parallel capabilities of the workstation.

3.1 Recall of the Indexes

Figure 1 shows the recall rate when the number of references is varied. The Figure presents three **KNR** mappings: vector, string and set mappings (rows) on two different data sets, namely **documents** and **CoPhIR** (columns). As we can see on figure 1(b) the permutation index and the brief index have a perfect recall for a small $|R|$. The other indexes performed below these two. On the other hand, for **documents** dataset, the lowest performance is presented by the permutation index and the brief index (figure 1(a)). This behavior is consequence of the high dimensionality of the **documents** data set.

String mappings (Figures 1(c) and 1(d)) show a diversity of performance. That is, **KNR LCS** has the highest recall rate in both data sets, this is followed by **KNR Levenshtein**. The worst performance obtained by **KNR** prefixes, note that for this method the recall get worst as the number of references increases.

In the set mappings, namely Figures 1(e) and 1(f), all the indexes share almost an identical recall rate. **KNR** set methods need larger values of $|R|$ to achieve its optimal value. This is a good characteristic, because larger $|R|$ means faster inverted indexes (which is the underlying data structure for set mappings, Section 3.3).

3.2 Increasing Recall Using Several **KNR** Indexes

A general technique to increase the recall in **KNR** methods is the usage of several indexes, as reported by [10,6]. Unfortunately, as expected, the recall increases but also the time and the storage requirements.

The increasing in the recall is produced by the diversity of several R sets. Each index retrieves a subset of the complete result, note that the size of the partial result depends on the particular **KNR** method. The union of partial results is our final result set.

Note that joining results is not tight in the length of different R sets or the particular **KNR** method, nor the mix of them. In order to show this behavior, Figure 2(a) shows the recall in each index. In the right side, Figure 2(b) shows the corresponding *cumulative* recall. The cumulative recall is obtained by using the union of the current and all the smaller results. For example the cumulative point for $|R| = 512$ needs the union of result from indexes working with $|R|$ as 64, 128, 256, and 512. For $|R| = 512$ four indexes are needed to work.

⁶ <http://www.mono-project.org>

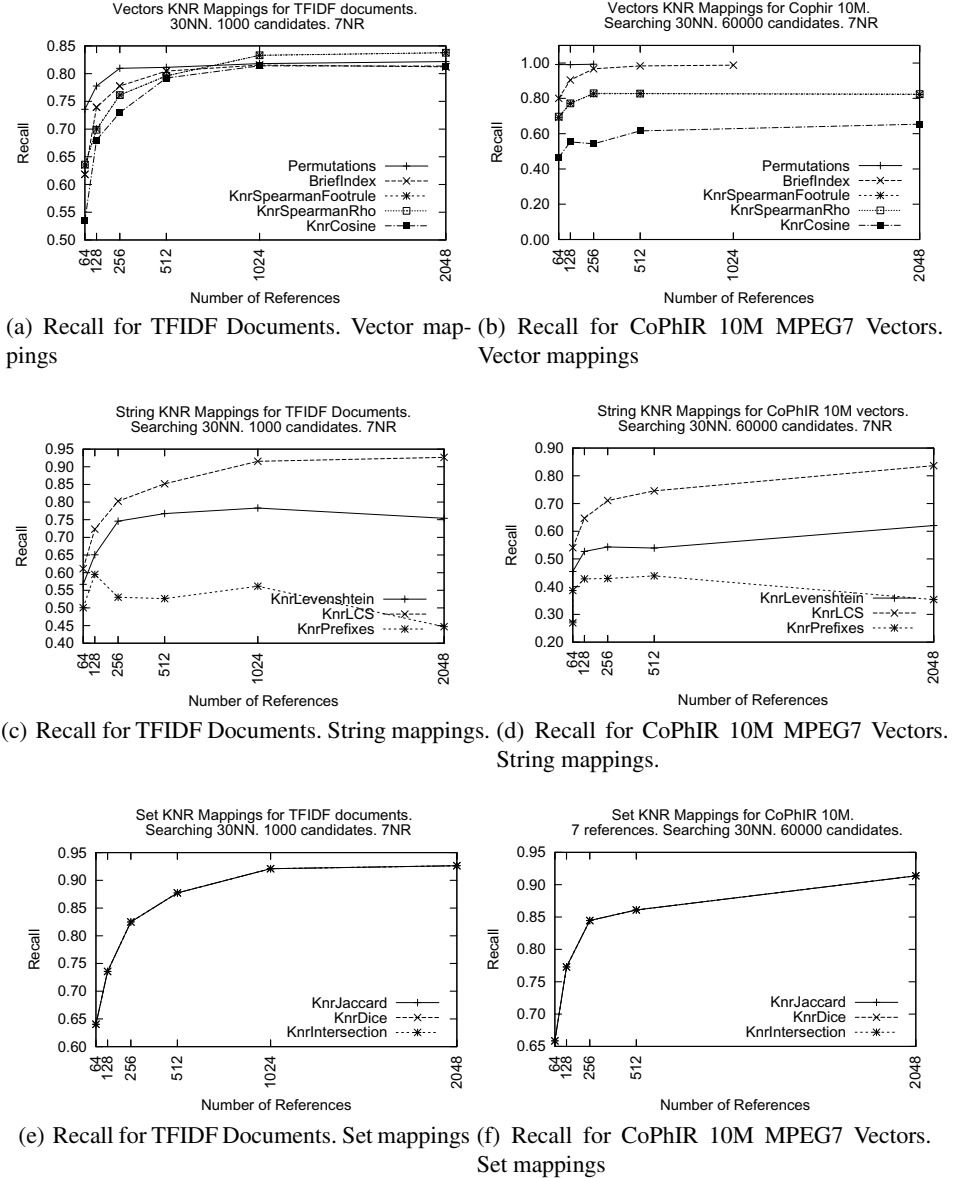


Fig. 1. Recall behavior of **KNR** mappings

This solution can be expensive, but it is a simple and effective solution to the problem of low recall of some **KNR** indexes and it can be used to index *very* large databases.

As shown in Figure 2(a), the best single recall is smaller than the corresponding point in 2(b). This is evident, since partial results are joined. A particular large improvement was found for **KNR** Prefixes (PP-Index), which improves dramatically the recall,

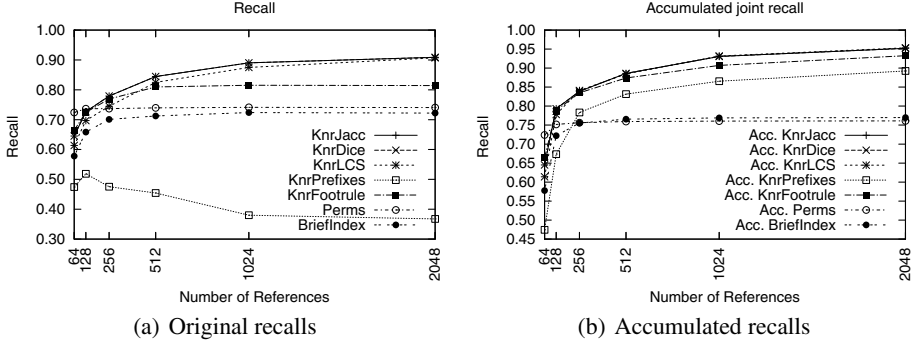


Fig. 2. Joining **KNR** results for 112544 color's histograms, searching 30NN. The *accumulated* curves uses the *union* of current and smaller $|R|$ results.

transforming the index into an appealing option for high quality requirements. Other indexes present a gain of 5 – 15%, which is still very important.

The same strategy is valid to speed up searches, using a partition of the database (a disjoint collection of subsets) across several search servers. As usual, hybrid approaches can be used to achieve both recall and speed enhancements.

3.3 Improvements Using K and γ Variations

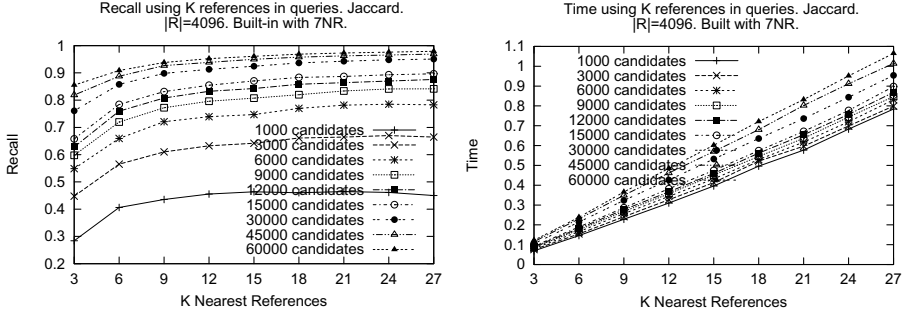
The optimization of K and γ parameters can be used to control the tradeoff between time and recall.

The parameter K has two roles, for building and for searching. At building time K modifies the size of the index, because it increase or decrease $|\hat{u}|$. When searching, increasing K allow to increase the recall without growing the index size and using the same index. The cost of searching- K shows up in increasing the number of set operations in the inverted index, which impacts the real time used for searching. Due to these characteristics, our experiments are focused on searching- K [\[10\]](#).

About the Implementation of this Experiment. In this experiment we used a simple inverted index to index the **KNR** Jaccard mapping. References are the thesaurus, and objects identifiers are inserted in the posting lists. The search algorithm consists in computing the set *union* and counting the cardinality of the union of the inverted lists (which is in fact the cardinality of the intersection of corresponding **KNR** sets).

Using an inverted index increases the space complexity from $nK \log |R|$ bits to $nK \log n$ bits, but reduces the cost of computing the candidate list. Let us define γ' as the maximum number of possible candidates, $\gamma' = |\bigcup_{x_i \in \mathbf{KNR}_d(q \in U, R)} \{\hat{v} \in \hat{U} : x_i \in \hat{v}\}|$, then we perform $O(\gamma')$ implicit evaluations of d' . The union uses $O(\log K)(\sum_{x_i \in \mathbf{KNR}_d(q \in U, R)} |\{\hat{v} \in \hat{U} : x_i \in \hat{v}\}|)$ comparisons. Finally we perform $\min\{\gamma', \gamma\}$ distances d . Notice that all these costs are not directly dependent of n .

⁷ Increasing searching- K is meaningless for **KNR** prefixes, permutations and brief permutations.



(a) Recall changing K ($|\hat{u}|$) to compute queries mappings (b) Time changing K ($|\hat{u}|$) to compute queries mappings

Fig. 3. γ and K strategies enhancing **KNR** methods. CoPhIR 10M 30NN.

This analysis is important since we present real time results along the recall in the CoPhIR 10M database (which can be considered a large database). The index construction uses seven nearest references ($K = 7$).

Time and Recall Tradeoff Induced by Searching- K . Figure 3(a) shows the behavior for **KNR** Jaccard for different γ and K values. These results show that a large K value produce higher recalls, even in moderate γ values (e.g. 15000 candidates). The improvements by γ are rapidly stabilized and gives low increments for $\gamma \leq 30000$ candidates, under this configuration, recall values are up to 0.9 for $K = 6$, an being close to perfect for larger configurations. In the other hand, Figure 3(b) shows the impact of the K and γ variations in the search time. On the same Figure, large K values imply higher cost than increasing the number of candidates, i.e. there are more inverted lists (and objects) to compute the union operation.

4 Conclusions and Future Work

In this work, we presented a novel framework for approximate proximity search algorithms called K Nearest References (**KNR**) methods. This framework consists in mapping spaces from a general metric space or similarity space to a simpler space using **KNN** queries in a set of references R . The produced mapped spaces have a simple and well defined structure, allowing the creation of string indexes and inverted indexes [7][12].

As part of our study, we described and analyzed previous methods and explained how they belong to **KNR** indexes. Furthermore, we presented several enhancements for **KNR** methods based on parallelization, distribution and parameter optimizations. These enhancements reduce the search time, increase the recall, and highlight the scalability properties of the **KNR** mappings.

Notice that the set of references R can be indexed to search for the K nearest references. Since R is relatively small, we can use an exact index like AESA [2] for the searching. Using an approximate index to search for the **KNR** should be also consider, but may lower the recall of the subsequent index.

Finally, the present work focus on static collections, as a consequence a deep study on dynamic collections would be interesting. That is, **KNR** algorithms supporting efficient updates, inserts and deletion of items.

Acknowledgements

The first author acknowledges support from the National Council of Science and Technology (CONACyT) of Mexico to pursue graduate studies at the Universidad Michoacana de San Nicolas de Hidalgo. Also, part of this work was developed during his stay at the DCC of the University of Chile.

References

1. Samet, H.: Foundations of Multidimensional and Metric Data Structures. Morgan Kaufmann, San Francisco (2006)
2. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in metric spaces. *ACM Comput. Surv.* 33(3), 273–321 (2001)
3. Böhm, C., Berchtold, S., Keim, D.A.: Searching in high-dimensional spaces: Index structures for improving the performance of multimedia databases. *ACM Computing Surveys* 33(3), 322–373 (2001)
4. Chávez, E., Navarro, G.: Probabilistic proximity search: Fighting the curse of dimensionality in metric spaces. *Information Processing Letters* 85, 39–46 (2003)
5. Chavez, E., Figueroa, K., Navarro, G.: Effective proximity retrieval by ordering permutations. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 30(9), 1647–1658 (2008)
6. Amato, G., Savino, P.: Approximate similarity search in metric spaces using inverted files. In: *InfoScale 2008: Proceedings of the 3rd international conference on Scalable information systems, ICST, Brussels, Belgium, Belgium, ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering)*, pp. 1–10 (2008)
7. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: *Modern Information Retrieval*. ACM Press / Addison-Wesley, New York (1999)
8. Witten, I.H., Moffat, A., Bell, T.C.: *Managing Gigabytes: Compressing and Indexing documents and images*, 2nd edn. Morgan Kaufmann Publishing, San Francisco (1999)
9. Téllez, E.S., Chávez, E., Camarena-Ibarrola, A.: A brief index for proximity searching. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) *CIARP 2009. LNCS*, vol. 5856, pp. 529–536. Springer, Heidelberg (2009)
10. Esuli, A.: Pp-index: Using permutation prefixes for efficient and scalable approximate similarity search. In: *LSDS-IR Workshop* (2009)
11. Navarro, G.: A guided tour to approximate string matching. *ACM Computing Surveys* 33(1), 31–88 (2001)
12. Grossman, D.A., Frieder, O.: *Information Retrieval: Algorithms and Heuristics*. Springer, Heidelberg (2004)
13. Bolettieri, P., Esuli, A., Falchi, F., Lucchese, C., Perego, R., Piccioli, T., Rabitti, F.: Cophir: a test collection for content-based image retrieval. *CoRR* abs/0905.4627v2 (2009)

Application of Pattern Recognition Techniques to Hydrogeological Modeling of Mature Oilfields

Leonid Sheremetov, Ana Cosultchi, Ildar Batyrshin, and Jorge Velasco-Hernandez

Mexican Petroleum Institute, Eje Central Lazaro Cardenas 152,
San Bartolo Atepehuacan, 07730, Distrito Federal, Mexico
{sher, acosul, batyr, velascoj}@imp.mx

Abstract. Several pattern recognition techniques are applied for hydrogeological modeling of mature oilfields. Principle component analysis and clustering have become an integral part of microarray data analysis and interpretation. The algorithmic basis of clustering – the application of unsupervised machine-learning techniques to identify the patterns inherent in a data set – is well established. This paper discusses the motivations for and applications of these techniques to integrate water production data with other physicochemical information in order to classify the aquifers of an oilfield. Further, two time series pattern recognition techniques for basic water cut signatures are discussed and integrated within the methodology for water breakthrough mechanism identification.

Keywords: Principal component analysis, clustering, time series pattern, oilfield.

1 Introduction

In engineering practice data-driven approaches usually help to identify unknown facts and inspire further research of the discovered phenomena. In this study the application of pattern recognition techniques (PRT) is illustrated in the context of hydrogeological modeling of mature oilfields. Water associated to hydrocarbon – known as natural brine or formation water – is water trapped in underground formations that is brought to the surface along with oil or gas. It is one of the most effective driving mechanisms for oil production, but when the amount of produced water becomes excessive, oil production drops down reducing the lifespan of most hydrocarbon wells [1]. It is then not surprising that the nature, chemical properties and quantity of produced or injected water during recovery process all have a direct impact on oilfield development. A hydrogeological model deals with the determining the dynamic behavior of fluids within the formation, the identification of the number of aquifers and their impact on oil production, i.e. the origin of the produced water, the mechanism of intrusion and the flow path directions.

Most of physically-based approaches used to build up a dynamic model of the reservoir, such as geochemical models, pressure tests, tracing tests, require expensive and time-consuming procedures and are not risk-free since sometimes may result in the well shut-in. Moreover, domain experts usually use data-driven procedures

analyzing the results of these tests in order to suggest features for complex heterogeneous reservoirs. When studying directly numerical data, different associations between water-related parameters can simply escape because of the amount of information, which makes the manual analysis impractical. In order to solve these difficulties, the data analysis methods and PRT should be used [2].

Data analysis methods can comprise a necessary alternative and an additional tool for hydrogeological modeling. PRT have been widely used for production and pressure data analysis in different studies of reservoir behavior [3]. Nevertheless, to our knowledge, they have not been systematically used to evaluate the dynamic behavior of fluids. The paper discusses the motivations for and applications of PRT in order to classify the aquifers of an oilfield and identification of water behavioral patterns necessary for establishing water control methods.

The proposed methodology consists of several sequential steps of data analysis to describe the dynamic behavior of formation water in an oilfield. First, the principal component analysis (PCA) of various parameters characterizing oilfield formation water was applied. Second, based on this analysis two clusters of petroleum wells were determined applying hierarchical cluster analysis (HCA). Third, a description of these clusters by parameter values discriminating these clusters one from another was given. Forth, spatial representation of oilfield wells showed that obtained clusters had good spatial grouping. Finally, using new technique of temporal-spatial visualization proposed in this paper, the dynamics of water invasion in petroleum wells related to the spatial clustering of wells was shown.

In the rest of the paper a methodology, which makes use of several PRT is presented and exemplified with a case study of a Mexican oilfield located in the coastal swamps of the Gulf of Mexico.

2 Multivariate Statistical and Pattern Recognition Techniques for Hydrogeological Modeling

The aquifer is described by physicochemical properties of formation water (brine), which also can be used to identify the water likely migration trends. Conventional practice in production engineering usually assumes that for any formation there is only one common and huge aquifer associated. Although that hypothesis may be true, some oilfields have revealed the existence of various aquifers [4]. Aquifer identification methodology described here roughly consists in the application of multivariate statistical techniques to fluid properties and production data.

Dimensionality reduction is the first step. There are two main reasons to keep the dimensionality of the pattern representation as small as possible: measurement cost and classification accuracy. In many practical cases, not all the measured variables from a high-dimensional datasets are important for understanding the underlying phenomena. The best known unsupervised feature extractor is the PCA, which is the most commonly used technique for allowing high-dimensional representations to be compressed into a lower dimensionality without implication of statistical significance testing [5]. In the proposed method, PCA is applied with a Varimax rotation, extracting the factors or principal components. Kaiser criterion was also used (eigenvalue > 1).

The recognition problem is considered as a classification or categorization task, where the classes may be defined by the domain expert (supervised classification) or are learnt based on the similarity of patterns (unsupervised classification). The object is represented by a d -dimensional vector of attributes. To make a classification, the space spanned by these vectors, is subdivided using decision boundaries.

Unsupervised classification method, HCA is used to classify entities with similar properties based on their nearness or similarity. This method has been used to identify aquifers as zones with similar or distinct hydrochemical regimes of ground or superficial water. To establish decision boundaries, concepts from statistical decision theory are utilized.

Once aquifers are identified, the preferential flow directions can be inferred from the water production historical data while the water breakthrough mechanism can be evaluated from the logistic curve parameters of the water-cut curve. Water-to-oil ratio (WOR) or water cut data plots (visual patterns) help to understand the type of flow and the water breakthrough mechanism [6]. Several behavior patterns identified from the basic signatures of WOR are defined in the literature. In [7] three patterns are defined: rapid increase, rapid increase followed by a straight line and gradual increase. Each of these patterns is associated with water invasion problems. For instance, a high water cut following an S-shape curve with a high and positive slope indicates flow behind pipes, fractures or fracture-lake features and specially edge water drive associated to linear flow as defined in [1].

The real behavior usually makes hard to relate the water production curve with these patterns [7,8]. Two methods based on moving approximation (MAP) transform and logistic growth curve (LGC) respectively were developed and tested [9, 10]. A LGC can be used to model functions that increase gradually at first, more rapidly in the middle, followed by a slow increase at the end. LGC is proposed as a mathematical tool to be applied to WOR data in order to determine the type of flow and the water invasion mechanism. The formula for the logistic function follows Gompertz curves:

$$y = ae^{-e^{-k(x-x_c)}} \quad (1)$$

where y represents the water cut in %; x is the time of production in months; a is the upper asymptote, x_c is the time of maximum growth and k is the growth rate. For water influx behavior, k is positive and the logistic function will always increase.

In the second pattern recognition method, the MAP-transform replaces values of the WOR time series $y = (y_i, y_{i+1}, \dots, y_n)$ with the sequence of slope values $MAP(y) = (a_1, \dots, a_N)$, where $N = n - k + 1$, obtained as a result of moving approximations of time series in a moving window of size k . A window W_i of a length $k > 1$ is a sequence of indexes $W_i = (i, i+1, \dots, i+k-1)$, $i \in \{1, \dots, n-k+1\}$. The slope values $a_1, \dots, a_{n-k+1}$ are called local trends of a linear function $f_i = a_i t + b_i$ with parameters $\{a_i, b_i\}$ minimizing the criterion:

$$Q(f_i, y_{W_i}) = \sum_{j=i}^{i+k-1} (f_i(t_j) - y_j)^2 = \sum_{j=i}^{i+k-1} (a_i t_j + b_i - y_j)^2 \quad (2)$$

This function is called a moving (least squares) approximation of y and approximates the data of the time series using a sliding window. Being based on the measurement of a distance between sequences of slope values, the method looks for the pattern $(a_i, \dots, a_{i+m})$, where m is a size of the pattern given by the sequence of the slopes, in $MAP(y)$ closest to the searched pattern [9].

3 Application of Pattern Recognition Techniques to the Oilfield Data

Pattern recognition techniques were applied to a data set collected from a Mexican oilfield. All mathematical and statistical computations were made using ExcelTM, OriginProTM and STATISTICATM software while visualization of some maps have been done with VMDPetro software developed in the IMP.

3.1 Cluster Analysis of Physic-chemical Water Properties for Aquifer Identification

First, the analysis of physic-chemical water properties was carried out. Water from 16 wells was sampled and analyzed in the laboratory to fulfill a dataset with up to 147 analyses (observations). The data of each analysis were examined, missing values were imputed and outliers were excluded from the dataset. From 16 properties after correlation analysis, the following ones are kept for the following study: TDS, Ca/Mg, Na/Cl, SO_4 , HCO_3 , Fe and SiO_2 . Oil properties were measured and reported only at the beginning of the well production. These properties include fluid density or API, dynamic or absolute viscosity related to internal resistance of the fluid to flow and cinematic viscosity, a physical property which depends on fluid density and also the carbon content measured by two techniques: Conradson and Ramsbottom.

The PCA showed interesting results. While using water and oil properties three principal components were identified: two of them describing water properties and one – for oil, with a total loadings of 70.93%. Nevertheless, results using only water variables show that the system variance can be explained with two PC's with a total loading of 81.72%. It means that oil properties can be excluded from this analysis. Table 1 shows the PC's compositions and loadings after Varimax normalization. PC_i refers to the i -th principal component. PC_1 explains 64.27% (after Varimax rotation - 59%) of the total variance and the loadings are strong and positive for well depth, SO_4^{2-} and SiO_2 , and strong and negative for TDS and Ca/Mg ratio variables. This PC relates to the mineral characteristics of water and is called "water mineralogy factor". PC_2 explains 17.45% (after Varimax rotation - 23%) of the total variance and shows strong and positive loading for Fe ion.

For HCA the set containing the well depth and six water properties – TDS, SO_4^{2-} , SiO_2 , Fe, Ca^{2+}/Mg^{2+} and Na^+/Cl^- – arranged by wells was used. Cluster analysis rendered a dendrogram (Fig. 1) where all samples of the well parameters were grouped into two statistically significant clusters at $(D_{link}/D_{max}) \cdot 100 < 70\%$. The clustering procedure highlighted two groups of wells in a very definite way because the wells in these groups have similar characteristics. Cluster A contains 10 wells, half of them producing from Upper Jurassic Kimmeridgian (UJ_{Kim}) and the other half - from Lower Cretaceous (LC) formation; and cluster B contains 6 wells, all of them producing from LC formation.

Table 1. Results of PCA (Eigenvalues and Factor Loadings after Varimax normalization, Marked loadings are > .7) and using selected water variables.

	Eigenvalue	% Total	Cumulative	Cumulative
PC1	4.50	64.27	4.50	64.27
PC2	1.22	17.45	5.72	81.72

Variables	PC1	PC2
Well Depth	0.76	0.18
TDS	-0.91	0.35
SO ₄ ²⁻ , meq/l	0.77	-0.46
Fe, mg/l	0.01	0.96
SiO ₂ , mg/L	0.86	0.02
Ca/Mg %	-0.85	0.13
Na/Cl %	0.81	-0.55
Expl.Var	4.10	1.62
Prp.Totl	0.59	0.23

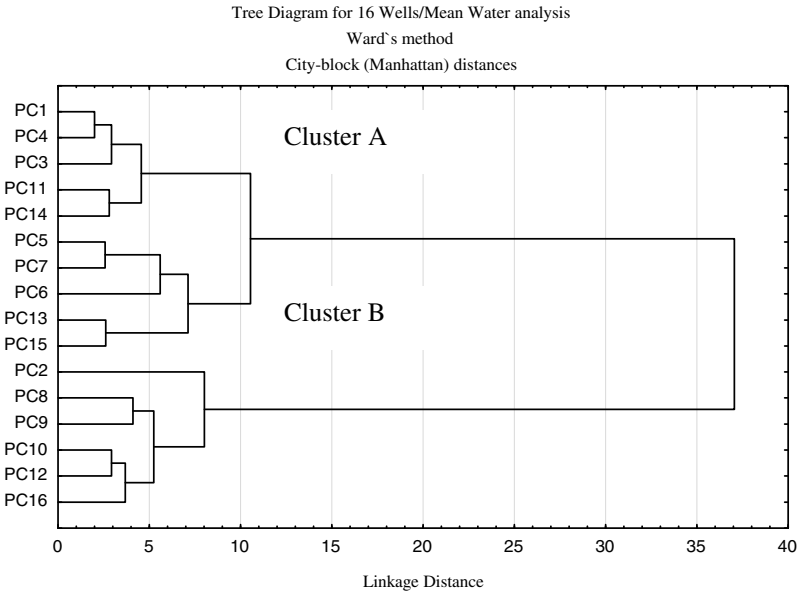


Fig. 1. The hierarchical clustering dendrogram of 16 wells of the PC oilfield based on water properties: Ward's method, Manhattan (city-block) distance

Such classification has a clear physical interpretation. Comparison of water properties (not shown here for space limits) shows that Salinity, Electric Conductivity, TDS mean values of Cluster A aquifer are 2.5 times higher than those of Cluster B aquifer and such difference is sustained by the concentration of Cl and Mg ions. Total hardness mean is 5.6 times greater for Cluster A compared to Cluster B and so the concentration of Ca cations. Concentration of SO_4 anions for Cluster B is 2.6 times the concentration of this anion in Cluster A. This difference may be related to the burial conditions of the aquifer or to geochemical process of calcium sulfate mineral dissolution, which is commonly associated to petroleum formations.

The differences between properties are also related to the well depth: Cluster B is grouping deeper wells than Cluster A. Table 2 contains the mean values of all measured variables calculated for each cluster including those, which were not involved in clustering procedure.

Table 2. Statistics of each cluster

Variables	Cluster A		Cluster B	
	Mean value	Std. Desv.	Mean value	Std. Desv.
Well Depth	5889.20	278.31	6363.67	261.03
Water properties				
Density, g/cm ³	1.21	0.02	1.08	0.01
Salinity, mg/L	300750.74	34833.48	116571.30	12250.80
El. Conductivity, mS/cm	605.02	48.97	239.54	28.62
Total hardness, mg/L	137531.58	27510.47	24737.04	3553.79
TDS mg/L	309348.91	21364.85	119294.03	14383.69
Ions, meq/L				
Na ⁺	59450.45	5547.65	34902.48	3451.45
Ca ²⁺	47913.12	9894.68	7610.37	1673.34
Mg ²⁺	4006.76	1289.74	1418.67	433.82
Cl ⁻	186302.42	16292.35	70947.99	7443.87
SO ₄ ²⁻	184.31	97.28	478.38	130.94
HCO ₃ ⁻	330.06	57.53	397.81	128.44
% Ca/ %Mg	7.63	1.80	3.66	1.66
Na%/Cl%	0.49	0.06	0.76	0.02
Solids, mg/L				
Fe	3.94	8.55	0.76	0.80
SiO ₂	46.38	12.72	88.48	37.06
Crude Oil properties				
API	31.36	1.32	31.11	1.20
Dyn. Visc., cP	5.73	1.54	5.81	1.29
Say. Visc., SSU	51.28	6.99	52.67	7.41
RCR, wt%	2.50	0.69	2.77	0.89
RCC, wt%	3.29	0.87	3.23	0.63

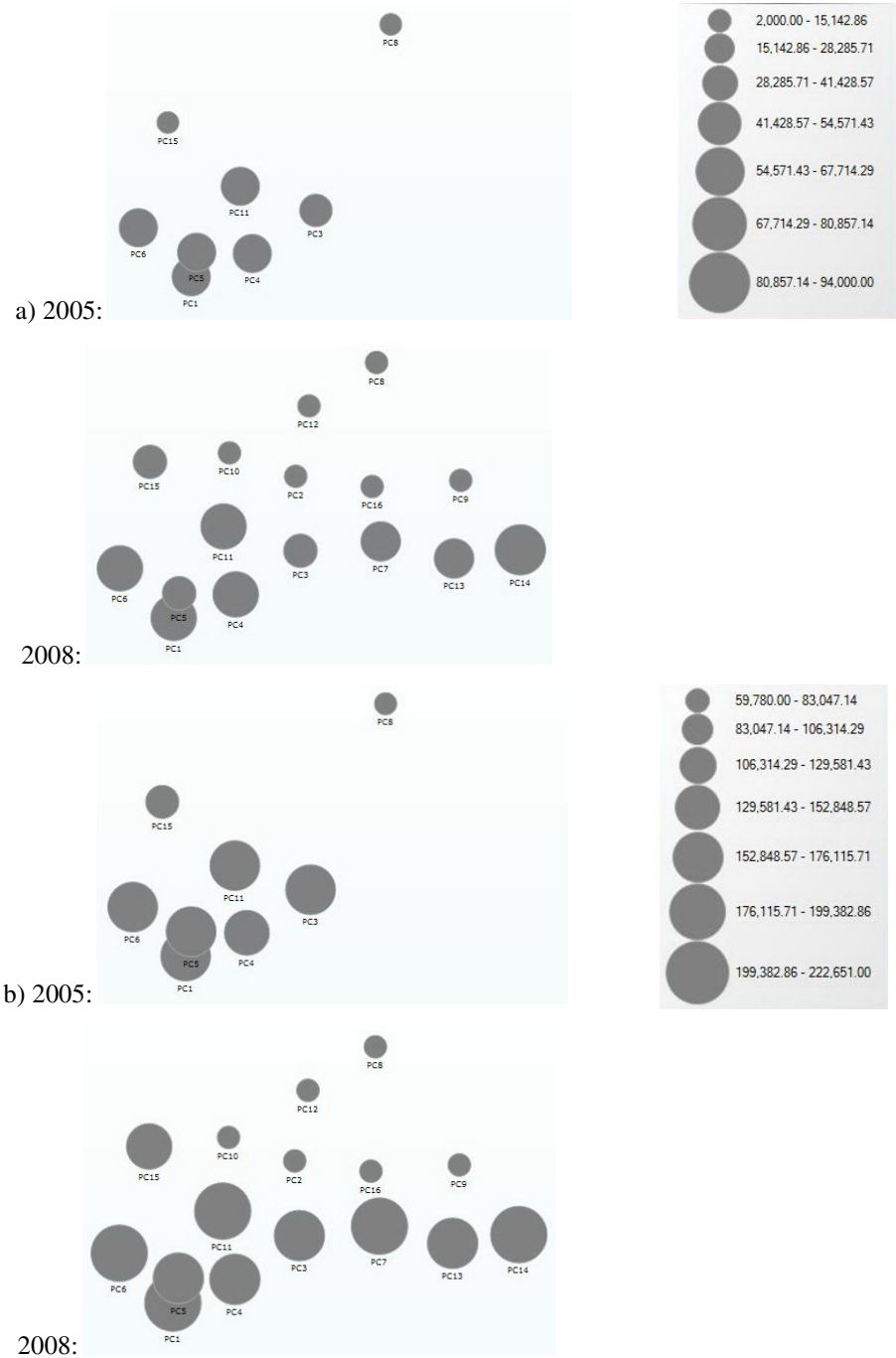


Fig. 2. Spatial and temporal distribution of a) Ca^{2+} and b) Cl^- water ion concentrations corresponding to PC wells water analysis dated 2005 and 2008 respectively

For the further interpretation of the HCA results, a geographical distribution of wells was analyzed, showing a clear coincidence with their location dividing the oilfield into two parts: Southern and Northern. For illustrative purposes, only water ion concentrations for Ca^{2+} and Cl^+ measured during the autumn of 2005 and after four years of production (2008) are presented in Fig. 2.

3.2 Pattern Recognition of Basic WOR Signatures for Water Breakthrough Mechanism Identification

The finding of two aquifers raises two important questions: if there is oil migration between formations through fractures or conductive faults and if water influx is the mechanism connecting these formations? The parameters of the logistic curve, the approximated date of the starting date of water production and the maximum value of water cut are useful for the interpretation of water influx mechanism within each well. For illustrative purposes three most typical identified patterns are shown in Fig. 3. Water cut historical behavior of the wells shows a good fitting of the LGC (Table 3). For the definition of the typical patterns in the form of fuzzy rules [8], the growth rate k and the upper asymptote value a were learnt from the observation data. These parameters can be useful for the field operators and give them a most clear idea of the water problem. Parameter a is very important when compared to the critical water cut calculated from Nodal Analysis used for the analysis of well performance, while k indicates how stressful is the influx problem and what solution should be applied.

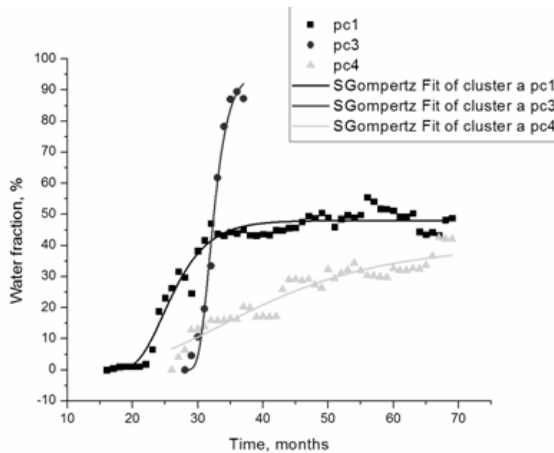


Fig. 3. Three typical behavior WOR patterns with associated logistic Gompertz curve fitting

Additionally to the analysis of the type and distribution of oilfield formation water the dynamics of water invasion of wells has also been studied. For the identification of hydraulic water flow paths within the formation, a new form of representation of spatial-temporal events is proposed, where the earlier events are represented by larger circles (Fig. 4). The spatial distribution of events for each well clearly shows the sequence of water breakthrough within the clusters obtained by HCA. It can be seen

that the size of circles monotonically decreases from wells PC6, PC1 to PC7, PC,13, PC14 for cluster A and from PC8 and PC16 to PC10, PC2 and PC 9 for cluster B. So in these clusters the water invasion is coming from the opposite directions.

Table 3. Parameters of the LGC fitting the water cut data. The parameters are calculated according to the logistic function formula (1).

Model	SGompertz		
Equation	$y = a * \exp(-\exp(-k * (x - xc)))$		
Reduced Chi-Sqr	10.85888	21.5744	10.69689
Adj. R-Square	0.9624	0.98446	0.89447
		Value	Standard Error
pc1	a	48.04807	0.59964
pc1	xc	24.65875	0.28874
pc1	k	0.26092	0.0249
pc3	a	95.53129	5.10802
pc3	xc	31.76059	0.1672
pc3	k	0.62707	0.10336
pc4	a	39.8968	2.63364
pc4	xc	33.91353	1.18578
pc4	k	0.072	0.01226

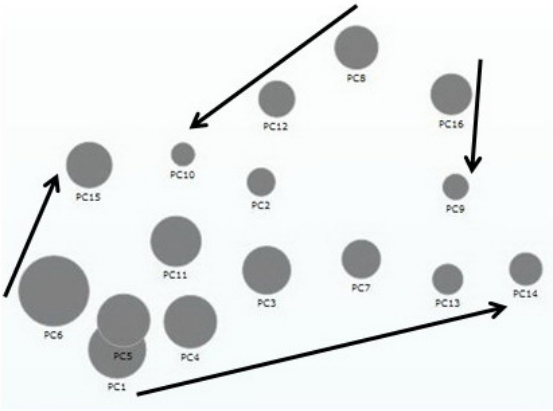


Fig. 4. Spatial-temporal distribution of wells' water breakthrough: the earlier events are represented by larger circles. The arrows show the sequence of these temporal events.

4 Conclusions

The paper shows that PRT can be considered as a powerful exploratory tools to evaluate the dynamic behavior of fluids within a formation; these methods can be

particularly useful when applied previous to any other technique since they can provide important clues not evident in the raw data. The distinctive feature of the proposed approach is the integration of PRT in order to advance with the reservoir hydrogeological modeling. A well-defined and sufficiently constrained recognition problem led to a compact pattern representation and a simple decision making strategy. A varimax rotation of the PCs led to a reduced number of PCs, each of them related to two groups of experimental variables with a respective meaning: water mineralogy factor PC1 and Fe-factor PC2, respectively. HCA has found spatial similarities in wells variations across the oil camp, reducing the number of groups to two although not all the members of the clusters are producing from the same formation. Water behavior PRT resulted in generation of fuzzy rules for diagnostic expert system called SMART-Agua [8].

Existence of large volumes of measurement data is a rather new issue in the oilfield practice, long tradition of using data analysis techniques in that field does not yet exist. The application of the proposed methodology can significantly reduce the time and the costs of petroleum engineering tasks, since the collected data are used for the analysis not requiring additional tests and well shut-ins.

Acknowledgments. The partial support for this work has been provided within the IMP projects D.00507, Y.00102 and Y.00122.

References

1. Seright, R.S., Lane, R.H., Sydansk, R.D.: A strategy for attacking excess water production, paper SPE 84966 (2003)
2. Sheremetov, L., Alvarado, M., Bañares-Alcántara, R., Aminzadeh, F.: Intelligent Computing in Petroleum Engineering (Editorial). *J. of Petroleum Science and Engineering* 47(1-2), 1–3 (2005)
3. Gaskari, R., Mohaghegh, S.D., Jalali, J.: An Integrated Technique for Production Data Analysis with Application to Mature Fields, SPE 100562 (2006)
4. Birkle, P., Martínez-García, B., Milland-Padrón, C.M., Eglington, B.M.: Origin and evolution of formation water at the Jujo–Tecominoacán oil reservoir, Gulf of Mexico. Part 2: Isotopic and field-production evidence for fluid connectivity. *Applied Geochemistry* 24, 555–573 (2009)
5. Lim, J.S.: Multivariate Statistical Techniques Including PCA and Rule Based Systems for Well Log Correlation. In: Nikraves, M., Aminzadeh, E., Zadeh, L.A. (eds.) *Developments in Petroleum Science*, pp. 673–688. Elsevier, Amsterdam (2003)
6. Yortsos, Y.C., Choi, Y., Yang, Z., Shah, P.C.: Analysis and Interpretation of Water/Oil Ratio in Waterfloods. SPEJ 4, 413–424, paper SPE 59477 (1999)
7. Bailey, B., Crabtree, M., Tyrie, J., Elphick, J., Kuchuk, F., Romano, C., Roodhart, L.: Water control. *Oilfield Review*, Schlumberger (2000)
8. Sheremetov, L., Batyrshin, I., Cosultchi, A., Martínez-Muñoz, J.: SMART-Agua: a Hybrid Intelligent System for Diagnostics. In: INES 2006, London, pp. 238–243 (2006)
9. Batyrshin, I., Sheremetov, L.: Time Series Pattern Recognition Based on MAP Transform and Local Trend Associations. In: Martínez-Trinidad, J.F., Carrasco Ochoa, J.A., Kittler, J. (eds.) *CIARP 2006. LNCS*, vol. 4225, pp. 910–919. Springer, Heidelberg (2006)
10. Cook, L.M.: Oscillation in the simple logistic growth model. *Nature* 207, 316 (1965)

On Trend Association Analysis of Time Series of Atmospheric Pollutants and Meteorological Variables in Mexico City Metropolitan Area

Victor Almanza and Ildar Batyrshin

Mexican Institute of Petroleum
Eje Central Lazaro Cardenas 152, Mexico City, Mexico
{vhalmanz,batyr}@imp.mx

Abstract. The paper studies trend associations between atmospheric pollutants and meteorological variables time series of Mexico City Metropolitan Area (MCMA) by applying the Moving Approximation Transform (MAP). This recently introduced technique measures and visualizes associations of the dynamics between different time series in the form of an association network. The paper studies associations between 5 atmospheric pollutants (SO_2 , O_3 , NO_2 , NO_x and $\text{PM}_{2.5}$) and 7 meteorological variables (mean wind velocity, minimum, average and maximum values of both temperature and relative humidity) measured daily during one year in three meteorological stations located in different zones of MCMA. These associations were studied for 4 seasons characterized by different meteorological conditions. For considered stations atmospheric pollutants and meteorological variables for different seasons positive and negative associations have been found and explained.

Keywords: Time series data mining, trend associations, MAP transform, atmospheric pollutants.

1 Introduction

Air pollution in urban areas is an issue of mayor concern due to its undesirable effects such as the public health impact, environmental damage and climate changes among others. For instance, there are epidemiological studies based on time series analysis to infer respect to the contribution of particle matter less than 10 and 2.5 micrometers to respiratory diseases in hospital admissions [1-3]. Methods of Soft Computing and Nonlinear Dynamical Systems have also been applied. For example, in [4] both fuzzy and neural networks (NN) were applied to develop a system which forecast concentrations of ozone daily maximums employing registers of four monitoring sites in Seoul, where patterns in the time series were fundamental for preparing the datasets for this system. In a similar way, Kukkonen et.al, [5] and Chaloulakou et.al [6] found that NN yielded better estimates for predicting pollutants concentrations. Also, Liu [7] developed a computational model in order to infer information the dynamics of chemical reactions that produce air pollution, and Cheng [8] improved the pollutant standard index by means of applying an entropy function. Nevertheless, Dillner [9]

and Hyvönen, et. al [10] deepens in the understanding of the pollution phenomena since they focus on the source of the particles and aerosol formation respectively. They used cluster analysis of aerosol time series.

Since Mexico City is considered a megacity, two main pollutants are of particular concern in the MCMA, ozone and particulate matter (PM), because of frequent exceedance to air quality standards several days a year. Long-range transport could influence air quality, and hence effects could be felt in regions far from their sources [11]. This is why it is important to infer about the physical and chemical behavior of air pollution.

The objective of the paper is to apply novel time series data mining techniques based on local trend associations [12] to pollutant time series in order to obtain information about possible associations between meteorological conditions and air pollution for different year seasons in three meteorological stations located in MCMA. Meteorological conditions near these stations differ one from another depending on wind strength and wind direction. The information obtained with this method, can confirm expected relations between meteorological parameters and air pollutants as well as new insights about such relations in different seasons in MCMA.

The paper has the following structure. Section 2 describes the data used in analysis. The Section 3 gives short description of the method of local trend associations of time series. Section 4 presents obtained results and their discussion. Section 5 contains conclusions.

2 Data Used for Analysis

Air pollution and meteorological time series of three stations of the Atmospheric Automatic Monitoring Network (RAMA by its Spanish initials) were considered for the analysis (Fig.1). These are Pedregal (PED), Tlalnepantla (TLA) and Merced (MER) stations. They were chosen because they represent different pollution zones in the MCMA. MER is located in the center of the Mexico City near heavily, paved and curbed surface streets with light-duty vehicles and modern heavy-duty diesel buses. PED is in a suburban neighborhood near clean, paved residential roads, lightly traveled and presents no nearby industries. TLA is both an industrial and residential area with nearby electronics manufacturing, corn milling, and metal fabricating facilities [13].

The pollutants considered were ozone (O₃), sulfur dioxide (SO₂), nitrogen dioxide (NO₂), nitrogen oxides (NO_x), and particulate matter less than two micrometers (PM_{2.5}); and the meteorological variables were wind direction (WD), wind velocity (WV) temperature (T) and relative humidity (RH). The raw time series consisted of hourly data for year 2004, but in this analysis only the daily maximum were considered since for pollutants the maximum concentration achieved in the day is more representative than the average since the latter could diminish the level of exposure of the population. In the case of meteorological variables, minimum, average and maximum values time series were constructed. Missing data were handled by simple interpolation.

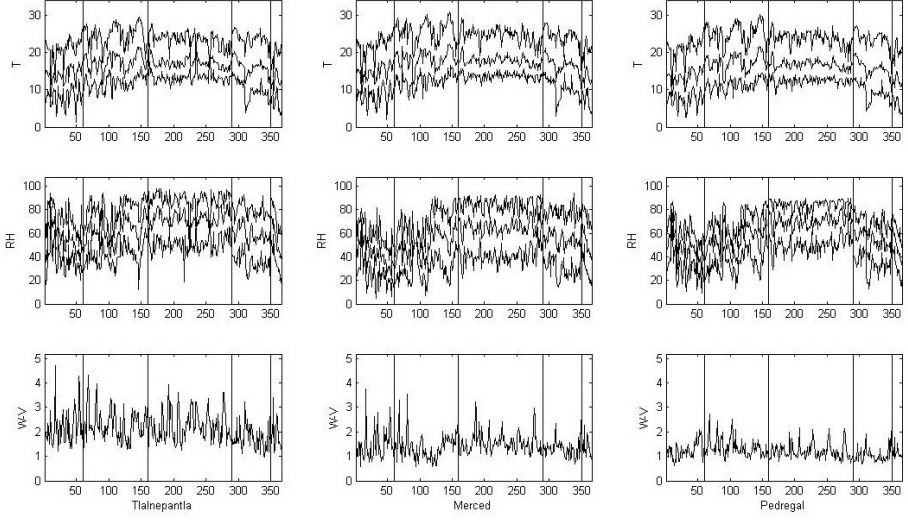


Fig. 1. Meteorological time series of three stations separated on 4 seasons (vertical lines denote borders between seasons)

3 Basic Notions of MAP Transform and Trend Associations

A time series (y, t) is a sequence $\{(y_i, t_i)\}$, $i \in I = (1, \dots, n)$, such that $t_i < t_{i+1}$ for all $i = 1, \dots, n-1$, where y_i and t_i are real numbers called time series values and time points, correspondingly. A time series (y, t) will be denoted also as y . A window W_i of a length $k > 1$ is a sequence of indexes $W_i = (i, i+1, \dots, i+k-1)$, $i \in \{1, \dots, n-k+1\}$. The sequence $y_{W_i} = (y_i, y_{i+1}, \dots, y_{i+k-1})$ of the corresponding values of time series y is called a partial time series induced by window W_i . A sequence $J = (W_1, W_2, \dots, W_{n-k+1})$ of all windows of size k , $(1 < k \leq n)$, is called a moving (or sliding) window. Such moving window is used, for example, in statistics in moving average procedure for smoothing time series when the value in the middle of the window replaced by the mean of values from this window.

Suppose J is a moving window of size k and $y_{W_i} = (y_i, y_{i+1}, \dots, y_{i+k-1})$, $i \in (1, 2, \dots, n-k+1)$, are corresponding partial time series in time points $(t_i, t_{i+1}, \dots, t_{i+k-1})$. A linear function $f_i = a_i t + b_i$ with parameters $\{a_i, b_i\}$ minimizing the criterion

$$Q(f_i, y_{W_i}) = \sum_{j=i}^{i+k-1} (f_i(t_j) - y_j)^2 = \sum_{j=i}^{i+k-1} (a_i t_j + b_i - y_j)^2. \quad (1)$$

is called a moving (least squares) approximation of y_{W_i} . The solution of (1) is well known and optimal values of parameters a_i, b_i can be calculated as follows:

$$a_i = \frac{\sum_{j=i}^{i+k-1} (t_j - \bar{t}_i)(y_j - \bar{y}_i)}{\sum_{j=i}^{i+k-1} (t_j - \bar{t}_i)^2}, \quad b_i = \bar{y}_i - a_i \bar{t}_i. \quad (2)$$

Where $\bar{t}_i$ and $\bar{y}_i$ are mean values of t and y .

Definition 1. A transformation $MAP_k(y, t) = a$, where $a = (a_1, \dots, a_{n-k+1})$ is a sequence of slope values obtained as a result of moving approximations of time series (y, t) in moving window of size k is called a moving approximation (MAP) transform of time series y . The slope values $a_1, \dots, a_{n-k+1}$ are called local trends.

Elements a_i , ($i = 1, \dots, n-k+1$) from $MAP_k(y, t)$ will be denoted as $MAP_{ki}(y, t)$.

In many applications time points $t_1, \dots, t_n$ are increasing with a constant step h such that $t_{i+1} - t_i = h$ for all $i = 1, \dots, n-1$. In such cases in MAP transform the set of time points $t = (t_1, \dots, t_n)$ can be replaced by the set of indexes $I = (1, \dots, n)$ as follows: $MAP_k(y, t) = (1/h)MAP_k(y, I)$ and the formula (2) for local trends can be simplified as follows [12].

As a measure of similarity between time series one can use measures of similarity between their MAP transforms. Some of these measures satisfy very nice properties of invariance to linear transformations of time and time series values.

Definition 2. Suppose $y = (y_1, \dots, y_n)$, $x = (x_1, \dots, x_n)$ are two time series and $MAP_k(y) = (a_{y1}, \dots, a_{ym})$, $MAP_k(x) = (a_{x1}, \dots, a_{xm})$, ($k \in \{2, \dots, n-1\}$, $m = n - k + 1$), are their MAP transforms. The following function is called a measure of local trend associations:

$$coss_k(y, x) = \sum_{i=1}^m (a_{yi} \cdot a_{xi}) / \sqrt{\sum_{i=1}^m a_{yi}^2 \cdot \sum_{j=1}^m a_{xj}^2}$$

Suppose p, q, r, s , ($p, r \neq 0$) are real values and (y, t) is a time series. Denote $py+q = (py_1+q, \dots, py_n+q)$ and $rt+s = (rt_1+s, \dots, rt_n+s)$. A transformation $L(y, t) = (py+q, rt+s)$ is called a linear transformation of time series (y, t) .

Theorem. Suppose L_1 and L_2 are two linear transformations of time series (y, t) and (x, t) given by the sets of parameters (p_1, q_1, r_1, s_1) and (p_2, q_2, r_2, s_2) , respectively, where $p_1, p_2, r_1, r_2 \neq 0$, then

$$coss_k(L_1(y, t), L_2(x, t)) = \text{sign}(p_1) \cdot \text{sign}(r_1) \cdot \text{sign}(p_2) \cdot \text{sign}(r_2) \cdot coss_k((y, t), (x, t)).$$

From this Theorem it follows a very nice invariance property of local trend association measure under various types of normalization of time series.

Analysis of associations between time series is based on the analysis of associations between them for different window size. The sequence of association values $AV(y, x) = (coss_2(y, x), \dots, coss_n(y, x))$ for all sizes of window is called an association function [12]. A specific measure of association between time series is defined by the subset of window sizes $J \subset \{2, \dots, n\}$ as a maximum or average of all associations $coss_k(y, x)$, $k \in J$. Examples of application of association measure to the classification of time series are considered in [12].

4 Results and Discussion

Trend association analysis was applied to pollutant and meteorological time series of three monitoring stations in MCMA, by means of the evaluation of local and global trends associations, in order to infer about possible dynamic relationships. Based on these associations the corresponding association networks were constructed. In these

networks only association values in the interval [0.5, 0.9] are discussed for each monitoring station and seasonal period.

In TLA for association values greater or equal to 0.8 most of the associations were for meteorological variables in all seasonal periods, mainly between the Relative Humidity and Temperature. For pollutants, in fall a positive association was the most relevant for the class {NO_x, PM_{2.5}}; and in winter only the class {NO₂, NO_x} with positive association exist. Since NO_x is defined as the sum of NO plus NO₂, it is expected that these chemical species correlate, but the reason to include it is because in the rest of the monitoring stations this class is present only in PED for the summer period. For association values greater or equal to 0.6 there is a cluster of positive association in the class {NO₂, NO_x, PM_{2.5}} for the summer period, a positive association in the class {O₃, PM_{2.5}} for summer and winter, and an inverse association for the class {WV, O₃} is present in fall and in winter. Example of association network is presented in Fig. 2, where solid lines denote positive associations and dashed lines denote negative associations. This latter class is also present in the same seasonal periods in MER station. So, it is possible that in TLA and MER, turbulent mixing promoted pollutant dispersion and transport which can reduce the concentration values of these species, since the maximum values in both stations were lower than in PED station. Moreover the complex wind patterns in MCMA [13], can influence the local wind patterns in PED possibly by slope flow. Moreover, in winter season the presence of cold fronts increases wind velocities.

The class {O₃, PM_{2.5}} is also relevant in MER and PED but only in winter season. This result suggests that the photochemical activity induces aerosol formation. However, as stated previously the higher concentration in PED is possibly related to stagnation conditions. The cluster {NO₂, NO_x, PM_{2.5}} is also present in PED station for winter season. This can suggest that road traffic is an important source for the aerosol particles in these regions [14].

Association values greater or equal to 0.6 in TLA show a cluster for the class {O₃, NO₂, NO_x} in spring season and a cluster for {T_{max}, NO_x, PM_{2.5}} for fall season. These clusters are not present neither in PED nor in MER. Both clusters suggest higher traffic activity for TLA in this season. Other clusters of interest are the class {PM_{2.5}, SO₂, RH_{max}} and the class {PM_{2.5}, SO₂, O₃}, which are relevant only for TLA in winter season. These clusters imply that in TLA, the sulfur content present in the atmospheric particles can be higher than in MER and PED. Composition analysis for particles in a study conducted in the MCMA showed that sulphate content was higher in TLA [15] because of large emissions of SO₂ proper of the industrial zone where TLA is located in. So, the MAP transform seems to be capturing important dynamics in the time series.

In this association level the class {RH_{max}, PM_{2.5}} is only present for MER in fall season. It is an inverse association that suggests the possibility that an increase in precipitation causes a decrease in particle concentration through scavenging [16]. Besides, the class of the inverse association {WV-PM_{2.5}} is important only in TLA and MER in winter season, suggesting that transport is an important source for decreasing the concentration of particles in the north part of MCMA.

Finally for the association value greater or equal to 0.5 more classes are obtained, but the most relevant ones are the class discussed above for TLA {PM_{2.5}, SO₂, O₃}

which now is present in MER for winter season and in PED for winter season too. The other class is the inverse association {RH, PM2.5} in MER for fall season.

It is worth to mention that in this stage the analysis can support other studies regarding the source contribution to the evolution of atmospheric pollutants. Moreover it is possible to apply this approach to information of aerosols measurements, sulfate ions concentration and precipitation among others in order to complement the present study.

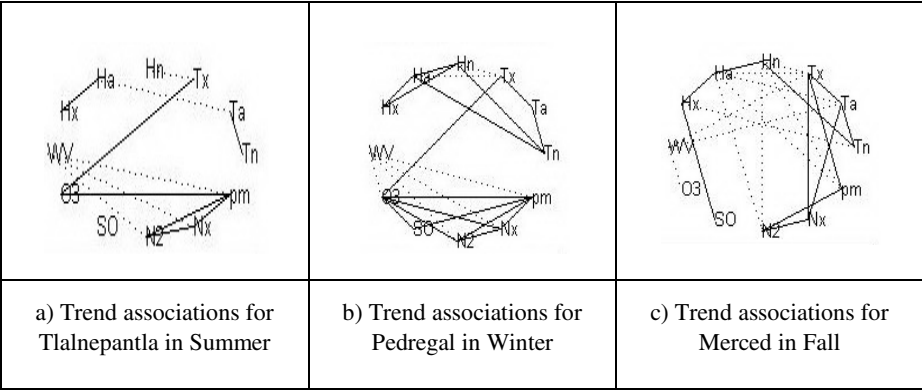


Fig. 2. Examples of association networks found for different seasons and stations

5 Conclusions

In this work new methods of data mining were applied to analyze relationships between dynamics of pollutants and meteorological time series of three representative monitoring stations in the MCMA for year 2004. This approach calculates associations between elements of the systems under consideration as associations between the respective time series. The association networks obtained by Moving Approximation Transform were discussed for association levels spanning from 0.5 to 0.9. At high levels the positive association class {NOx, PM2.5} was the most relevant for TLA station suggesting a high contribution from mobile sources. At moderate levels, the positive association class {O3, PM2.5}, was present in summer in winter in all the stations but not simultaneously. Besides the cluster {PM2.5, SO2, O3} gives insight about the photochemical activity in the north part of MCMA. Moreover, the method captures the inverse association {WV, O3} which suggest removal of particles by turbulent mixing, especially in winter seasons and in less degree in summer season only in TLA and MER. In wet season the inverse association {RHmax, PM2.5} suggest scavenging of particles in TLA and MER only. So, PED seems to be influenced by the local winds to promote stagnation conditions, which can be a reason for high concentration values in this year.

However it would be useful to include Volatile Organic Compounds, precipitation, and solar radiation time series in order to find new associations, and at another stage to consider time series of respiratory illness in order to find possible associations by

zone and by pollutant, which could be an aid in applying statistical time series analysis such as GAM or ARMA. Finally it is possible that MAP approach also could serve for finding patterns for fuzzy rules construction.

Acknowledgements

Victor Almanza thanks Dr. Gustavo Sosa for providing useful comments and suggestions for this work.

References

1. Roberts, S.: Biologically Plausible Particulate Air Pollution Mortality Concentration-Response Functions. *Environ. Health Perspect.* 112(3), 309–313 (2004)
2. Samoli, E., Analitis, A., Touloumi, G., Schwartz, J., Anderson, H.R., Sunyer, J., et al.: Estimating the Exposure-Response Relationships between Particulate Matter and Mortality within the APHEA Multicity Project. *Environ. Health Perspect.* 113, 88–95 (2005)
3. Galán, I., Tobías, A., Banegas, J.R., Aránguez, E.: Short-Term Effects of Air Pollution on daily Asthma Emergency Room Admissions. *Eur. Respir. J.* 22, 802–808 (2003)
4. Heo, J.K., Kim, D.S.: A New Method of Ozone Forecasting using Fuzzy Expert and Neural Network Systems. *Sci. Total Environ.* 325, 221–237 (2004)
5. Kukkonen, J., Partanen, L., Karppinen, A., Ruuskanen, J., Junninen, H., Kolehmainen, M., Niska, H., Dorling, S., Chatterton, T., Foxall, R., Cawley, G.: Extensive Evaluation of Neural Network Models for the Prediction of NO₂ and PM₁₀ Concentrations, compared with a Deterministic Modelling System and Measurements in Central Helsinki. *Atm. Env.* 37, 4539–4550 (2003)
6. Chaloulakou, A., Saisana, M., Spyrellis, N.: Comparative Assessment of Neural Networks and Regression Models for Forecasting Summertime Ozone in Athens. *Sci.Tot. Environ.* 313, 1–13 (2003)
7. Liu, Z., Lai, Y.C., Lopez, J.M.: Noise-induced Enhancement of Chemical Reactions in Nonlinear Flows. *Chaos.* 12(2), 417–425 (2002)
8. Cheng, W., Kuo, Y., Lin, P., Chang, K., Chen, Y., Lin, T., Huang, R.: Revised Air Quality Index Derived from an Entropy Function. *Atmos. Environ.* 38, 383–391 (2004)
9. Dillner, A.M.: A Quantitative Method for Clustering Size Distributions of Elements. *Atm. Env.* 39, 1525–1537 (2005)
10. Hyvönen, S., Junninen, H., Laakso, L., Dal Maso, M., Grönholm, T., Bonn, B., Keronen, P., Aalto, P., Hiltunen, V., Pohja, T., Launiainen, S., Hari, P., Mannila, H., Kulmala, M.: A Look at Aerosol Formation Using Data Mining Techniques. *Atmos Chem. Phys. Discuss.* 5, 7577–7611 (2005)
11. Molina, M., Molina, L.: Megacities and Atmospheric Pollution. *J. Air Waste Manage. Assoc.* 54, 644–680 (2004)
12. Batyrshin, I., Herrera-Avelar, R., Sheremetov, L., Panova, A.: Association Networks in Time Series Data Mining. In: *NAFIPS 2005 Soft Computing for Real World Applications*, Ann Arbor, Michigan, USA, pp. 754–759 (2005)

13. Edgerton, S.A., Bian, X., Doran, J.C., Fast, J.D., Hubbe, J.M., Malone, E.L., Shaw, W.J., Whiteman, C.D., Zhong, S., Arriaga, J.L., Ortiz, E., Ruiz, M., Sosa, G., Vega, E., Limon, T., Guzman, F., Archuleta, J., Bossert, J.E., Elliot, S.M., Lee, J.T., McNair, L.A., Chow, J.C., Watson, J.G., Coulter, R.L., Doskey, V.: Particulate Air Pollution in Mexico City: A Collaborative Research Project. *J. Air Waste M. A.* 49(10), 1221–1229 (1999)
14. Harrison, R., Deacon, A., Jones, M., Appleby, R.: Sources and Processes Affecting Concentrations of PM₁₀ and PM_{2.5} Particulate Matter in Birmingham (U.K). *Atm. Env.* 31(24), 4103–4117 (1997)
15. Chow, J.C., Watson, J.G., Edgerton, S.A., Vega, E.: Chemical Composition of PM_{2.5} and PM₁₀ in Mexico City During Winter 1997. *Sci. Tot. Environ.* 287, 177–201 (2002)
16. Tai, A., Mickley, L., Jacob, D.: Correlations Between Fine Particulate Matter (PM_{2.5}) and Meteorological Variables in the United States: Implications for the sensitivity of PM_{2.5} to Climate Change. *Atm. Env.* 44, 3976–3984 (2010)

Associative Memory Approach for the Diagnosis of Parkinson's Disease

Elena Acevedo^{*}, Antonio Acevedo, and Federico Felipe

Escuela Superior de Ingeniería Mecánica y Eléctrica, IPN, Mexico City, Mexico
{eacevedo,macevedo,ffelipe}@ipn.mx

Abstract. A method for diagnosing Parkinson's disease is presented. The proposal is based on associative approach, and we used this method for classifying patients with Parkinson's disease and those who are completely healthy. In particular, Alpha-Beta Bidirectional Associative Memory is used together with the modified Johnson-Möbius codification in order to deal with mixed noise. We use three methods for testing the performance of our method: Leave-One-Out, Hold-Out and K-fold Cross Validation and the average obtained was of 97.17%.

Keywords: Classification, Associative Models, Alpha-Beta BAM, Codification.

1 Introduction

Parkinson's disease (PD) was first described in a medical context in 1817 by James Parkinson, a general practitioner in London [1]. PD is the second most common neurodegenerative disorder after Alzheimer's disease. It has been suggested that the prevalence of the disease will double over the next 20 years [2].

Parkinson's disease is not an infection but a disease of the brain [3]. It is a chronic condition, an imbalance resulting from a loss of dopamine.

There are four cardinal features of PD [4] that can be grouped under the acronym TRAP: Tremor at rest, Rigidity, Akinesia (or bradykinesia) and Postural instability. The signs of Parkinsonism [5] are showed in Table 1.

There are treatments which help the patient to control PD in order to have a better way of life. But when there is not a previous diagnostic, the patient can suffer the consequences of the disease at an advanced stage, and in most of the time this can mean death. It is assessed that there is a 30% of patients without a diagnostic. However, PD is diagnosed on clinical criteria; there is no definitive test for diagnosis [4]. Certain tests may be done to help diagnose other conditions with similar symptoms. For instance, blood tests may be done to check for abnormal thyroid hormone levels or liver damage. An imaging test (such as a CT scan or an MRI) may be used to check for signs of a stroke or brain tumor.

^{*} María Elena Acevedo Mosqueda, Escuela Superior de Ingeniería Mecánica y Eléctrica del Instituto Politécnico Nacional, Av. IPN s/n Edif Z-4, 3er piso, Telecomunicaciones, Col. Lindavista, C.P. 07738, Mexico City, Mexico. E-mail: eacevedo@ipn.mx.

Table 1. Signs of Parkinsonism

Location or Activity	Manifestation
Face	• Loss of animation (masking) • Decreased blink rate
Speech	• Reduced volume • Dysarthric due to reduced amplitude and precision of the articulators of speech (lips, tongue, and palate)
Automatic movements	• Less gesturing when talking • Reduced arm swing while walking
Gait	• May have difficulty rising from seated position • Stooped • Shortened stride • Shuffling of feet (or feet more parallel to floor in contrast to normal landing on heel and pushing off with toes) • May exhibit freezing in place or hesitancy • Takes several steps to turn • Slowness (bradykinesia)
Balance	• Imbalance typically not an early sign in Parkinson's disease but may be in other parkinsonian disorders • Pull test may detect milder degrees of imbalance (retropulsion)
Tremor	• Hands when in position of repose (e.g., in lap or at sides during walking) • Legs when patient is seated with feet on floor • Chin • Markedly reduced with action (except with concurrent essential tremor)
Tone	• Rigidity of limbs and sometimes neck • If superimposed tremor, examiner appreciates cogwheel pattern
Rapid alternating movements	Slowed, reduced amplitude, and sometimes freezing of movement: ▪ Finger-thumb tapping ▪ Pronation-supination ▪ Opening-closing fist ▪ Foot or heel tapping
Other	• Eye movements may be slowed, and eye movement falls short of target (hypometric) • Meyerson's sign

Another type of imaging test, called PET (Position Emission Tomography), sometimes may detect low levels of dopamine in the brain. However, PET scanning is not commonly used to evaluate Parkinson's disease because it is very expensive, is not available in many hospitals, and is only used experimentally.

Speech analysis is an alternative for diagnosing PD. Speech is the most complex of innately acquired human motor skills [5], an activity characterized in normal adults by the production of about 14 distinguishable sounds per second through the coordinated actions of about 100 muscles innervated by multiple cranial and spinal nerves. The

ease with which we speak belies the complexity of the act, and that complexity may help explain why speech can be exquisitely sensitive to nervous system disease. In fact, changes in speech can be the only evidence of neurologic disease early in its evolution and sometimes the only significant impairment in a progressive or chronic neurologic condition. In such contexts, recognizing the meaning of specific speech signs and symptoms can provide important clues about the underlying pathophysiology and localization of neurologic disease.

In the other hand, a number of rating scales are used for the evaluation of motor impairment and disability in patients with PD [4], but most of these scales have not been fully evaluated for validity and reliability. The Hoehn and Yahr scale is commonly used to compare groups of patients and to provide gross assessment of disease progression, ranging from stage 0 (no signs of disease) to stage 5 (wheelchair bound or bedridden unless assisted). The Unified Parkinson's Disease Rating scale (UPDRS) [6] is the most well established scale for assessing disability and impairment. Studies making use of UPDRS to track the progression of PD suggest that the course of PD is not linear and that the rate of deterioration is variable and more rapid in the early phase of the disease and in patients with the postural instability gait difficulty (PIGD) of PD.

According to the World Health Organization [7] there are six millions of people affected by this disease in the world and fifty thousand in Mexico. The National Parkinson Foundation (NPF) [8] said that there are between 50 and 60 thousand new cases every year. Around de world, the incidence is from 20 to 25 new cases per year for every 100,000 citizens. A 2% from the affected people suffers the disease due to hereditary factors. Therefore, it is important to have the necessary means to classify Parkinson's patients.

Artificial Intelligence (AI) is an area which is extensively used for the classifying task. In medical diagnosis, a suitable classifier could be useful for an expert to increase the accuracy and reliability of the diagnostic and to minimize the possible errors.

Several works have been developed for diagnosing Parkinson's disease. Some of them use PET (Position Emission Tomography) or SPECT (Single Photon Emission Tomography) images as training data for Neural Network based systems [9] and Support Vector Machine based systems [10]. Another work analyzed the algorithms which model kinetics of dopamine by using the Laplace transformation of differential equations and by algebraic computation with the aid of Gröbner base constructions [11]. With the use of this method they obtained a rigorous solution with respect to the kinetic constants over the Laplace domain. Keijers et al. [12] use neural networks for the classification and rating dyskinesia as well as for extracting the important parameters to distinguish between dyskinesia and normal voluntary movements. An algorithm which combines a perceptron neural network with simple signal processing and rule-based classification [13] is used for automatic recognition and classification of walking patterns, in order to recognize disturbances during walking in PD patients.

Other related works used as training data the dataset introduced by Tsanas and Little [14] from the Oxford University. One approach for the classification of patients with Parkinson's disease is Neural Networks [15,16], another used Support Vector Machine algorithm [17], while Gil [18] applied a combination of both approaches.

In this work, we use this dataset but Associative Models are applied as an alternative approach for classifying patients with Parkinson's disease. In particular,

we used the Alpha-Beta Bidirectional Associative Memory [19]. The main feature of this model is its correct recall. It does not present the forgetting factor; every pattern trained is correctly recalled. The algorithm of the model is not an iterative process and has not stability problems. The correct recall is showed no matter the nature of the patterns.

In section 2, we present the basic concepts of Alpha-Beta associative memories, and we introduce the modified Johnson-Möbius code for avoiding mixed noise. Section 3 describes the main model used in this work. We present the results in section 4. Finally, conclusions are presented.

2 Alpha-Beta Associative Memories

An Associative Memory (AM) $\mathbf{M}$ is a system that relates input patterns, and outputs patterns. Two phases comprise the design of an AM: learning phase and recalling phase. In the learning phase, the memory is trained by associating input patterns $\mathbf{x}$ and output patterns $\mathbf{y}$ (see figure 1). Both input and output patterns can represent any association, for example: fingerprints with faces, names with telephone numbers, DNA sequences with names, etc.

Learning Phase

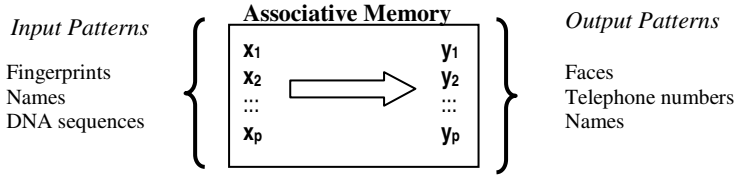


Fig. 1. The learning phase for an Associative Memory

After the associative memory was trained, output patterns can be recalled by presenting the input patterns to the memory. This task is performed by the recalling phase (see figure 2).

Recalling phase



Fig. 2. The recalling phase for an Associative Memory

In figure 2, one can observe that when an input pattern x^k is presented to the AM its corresponding pattern y^k must be recalled. Moreover, if a noisy version of an input pattern x^k represented by $\tilde{x}^k$ is presented to the associative memory, the corresponding pattern y^k should be recalled, if this happens then AM has a correct recall.

Formally, we can say that for a k integer and positive, the corresponding association will be denoted as (x^k, y^k) . The associative memory $\mathbf{M}$ is represented by a matrix whose ij -th component is m_{ij} . Memory $\mathbf{M}$ is generated from an a priori finite set of known associations, known as the fundamental or training set of associations.

If μ is an index, the fundamental set is represented as: $\{(x^\mu, y^\mu) | \mu = 1, 2, \dots, p\}$ with p the cardinality of the set. The patterns that form the fundamental set are called fundamental patterns. If it holds that $x^\mu = y^\mu, \forall \mu \in \{1, 2, \dots, p\}$, $\mathbf{M}$ is *autoassociative*, otherwise it is *heteroassociative*; in this case it is possible to establish that $\exists \mu \in \{1, 2, \dots, p\}$ for which $x^\mu \neq y^\mu$.

2.1 Alpha-Beta Associative Memories

Among the variety of associative memory models described in the scientific literature, there are two models that, because of their relevance, it is important to emphasize: morphological associative memories which were introduced by Ritter *et al* [20], and Alpha-Beta associative memories. Because of their excellent characteristics, which allow them to be superior in many aspects to other models for associative memories, morphological associative memories served as starter point for the creation and development of the Alpha-Beta associative memory.

The Alpha-Beta associative memories [21] are of two kinds (*max* and *min*-type memories) and are able to operate in two different modes. The operator α is useful at the learning phase, and the operator β is the basis for the pattern recall phase. The heart of the mathematical tools used in the Alpha-Beta model, are two binary operators designed specifically for these memories. These operators are defined as follows: first, we define the sets $A=\{0,1\}$ and $B=\{0,1,2\}$, then the operators α and β are defined in Tables 2 and 3, respectively:

Table 2. Alpha operator, $\alpha: A \times A \rightarrow B$

x	y	$\alpha(x,y)$
0	0	1
0	1	0
1	0	2
1	1	1

Table 3. Beta operator, $\beta: B \times A \rightarrow A$

x	y	$\beta(x,y)$
0	0	0
0	1	0
1	0	0
1	1	1
2	0	1
2	1	1

The sets A and B , the α and β operators, along with the usual $\wedge$ (minimum) and $\vee$ (maximum) operators, form the algebraic system $(A, B, \alpha, \beta, \wedge, \vee)$ which is the mathematical basis for the Alpha-Beta associative memories.

We present the learning and recalling phases for an autoassociative memory because the proposed model is designed with this type of memory.

Learning Phase

Step 1. For each $\mu = 1, 2, \dots, p$, from the pair $(\mathbf{x}^\mu, \mathbf{x}^\mu)$ a matrix is built

$$[\mathbf{x}^\mu \otimes (\mathbf{x}^\mu)^t]_{n \times n}$$

Step 2. If the memory is *max*-type, the maximum operator $\vee$ is applied to the matrices obtained in step 1, therefore a *max* matrix $\mathbf{V}$ is built. On the other hand, if the memory is *min*-type then the minimum $\wedge$ operator is applied for building a *min* matrix $\mathbf{\Lambda}$.

$$\mathbf{V} = \vee_{\mu=1}^p [\mathbf{x}^\mu \otimes (\mathbf{x}^\mu)^t] \quad \mathbf{\Lambda} = \wedge_{\mu=1}^p [\mathbf{x}^\mu \otimes (\mathbf{x}^\mu)^t]$$

Recalling Phase

The goal of this phase is recovering the output pattern $\mathbf{y}^\omega$ from input pattern $\mathbf{x}^\omega$ which is presented to the associative memory. β operator is used in this phase. If at learning phase a *max*-type memory $\mathbf{V}$ was built then pattern $\mathbf{x}^\omega$ will operate with matrix $\mathbf{V}$ and the *min* operator $\wedge$. Pattern $\mathbf{x}^\omega$ will operate with *min*-type memory $\mathbf{\Lambda}$ and the *max* operator $\vee$ if a *min*-type memory was built in learning phase.

$$(\vee \Delta_\beta \mathbf{x}^\omega)_i = \wedge_{j=1}^n \beta(v_{ij}, x_j^\omega) \quad \text{or} \quad (\wedge \nabla_\beta \mathbf{x}^\omega)_i = \vee_{j=1}^n \beta(\lambda_{ij}, x_j^\omega)$$

Types of Noise

An associative memory that works with binary values can encounter three types of noise: additive, subtractive and mixed. *Max* and *min*-type Alpha-Beta memories can handle with additive and subtractive noise, respectively. However, none of the two types can handle with mixed noise (see figure 3). Therefore, the patterns need a pre-processing in order to avoid the mixed noise. One way to avoid mixed noise is codifying the patterns using a code which allows the change of one bit. In the following section Johnson-Möbius code is presented.

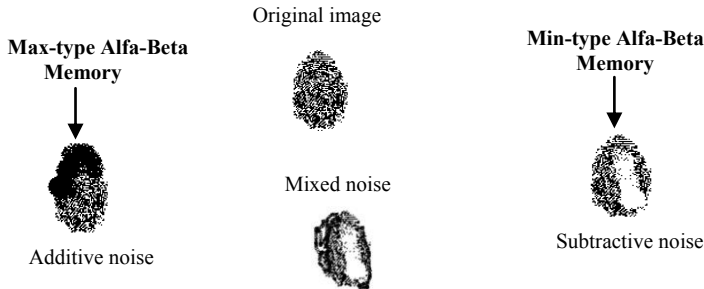


Fig. 3. Types of noise: additive, subtractive and mixed. They could appear when we work with binary values. *Max*-type Alpha-Beta associative memories can handle additive noise while *min*-type memories can handle subtractive noise but none of them handle mixed noise.

2.2 Johnson-Möbius Modified Code

Johnson-Möbius code is a binary code which allows the change of one bit between two consecutive numbers [22]. For representing a decimal number n with Johnson-Möbius code, $n/2$ bits are needed if n is even and $(n+1)/2$ bits are needed if n is odd. The following example shows the algorithm for the Johnson-Möbius code.

Johnson-Möbius Algorithm

Let be the set $r = \{2.5, 0.15, -0.1, 0.4, 1.4\} \subset \mathbf{R}$

Step 1. We can observe that in the set there is a negative number (-0.1), then the set has to be turn into a new set by adding 0.1 to each element. The new set is $t = \{2.6, 0.25, 0.0, 0.5, 1.5\}$

Step 2. Choose a fixed number d of decimals and truncate each of the numbers of the new set to d decimals. The number d depends on the required accurateness for the specific problem we are resolving. In this example we will use $d=1$ then we obtain $e = \{26, 2, 0, 5, 15\}$ where the maximum is $e_m = 26$.

Step 3. As 26 is an even number it is not necessary to add 1.

Step 4. $e_m / 2 = 13$.

Step 5. If $e_i < e_m/2$, then $e_m/2 - e_i$ zeros are generated and e_i ones are added. In the contrary, $e_m - e_i$ ones are generated and $e_i - e_m/2$ zeros are added. Table 4 shows the results.

Table 4. Johnson-Möbius codification for the set $\{2.5, 0.15, -0.1, 0.4, 1.4\}$

Decimal	Johnson-Möbius codification
26	11111111111111
2	11000000000000
0	00000000000000
5	11111000000000
15	00111111111111

Johnson-Möbius Modified Algorithm [23]

N bits are necessary for representing a decimal number n with the Johnson-Möbius code. We present an illustrative example.

Let be the set $r = \{2.5, 0.15, -0.1, 0.4, 1.4\} \subset \mathbf{R}$

Step 1 and Step 2. The same process is performed.

Step 3. For each e_i from the set e , $e_m - e_i$ zeros are concatenated with e_i ones. Table 5 shows the results.

Table 5. Modified Johnson-Möbius codification for the set $\{2.5, 0.15, -0.1, 0.4, 1.4\}$

Decimal	Johnson-Möbius codification
26	111111111111111111111111111111
2	000000000000000000000000000011
0	000000000000000000000000000000
5	000000000000000000000000011111
15	000000000001111111111111111111

Even though more bits are needed for representing a decimal number with Johnson-Möbius modified code, this pre-processing has shown to be better than the original code for avoiding mixed noise.

3 Alpha-Beta Bidirectional Associative Memories

Generally, any bidirectional associative memory model appearing in current scientific literature could be draw as figure 4 shows.



Fig. 4. General scheme of a Bidirectional Associative Memory

General BAM is a “black box” operating in the next way: given a pattern x , associated pattern y is obtained, and given the pattern y , associated pattern x is recalled. Besides, if we assume that $\tilde{x}$ and $\tilde{y}$ are noisy versions of x and y , respectively, it is expected that BAM could recover all corresponding free noise patterns x and y .

Before going into detail over the processing of an Alpha-Beta BAM, we will define the following.

In this work we will assume that Alpha-Beta associative memories have a fundamental set denoted by $\{(\mathbf{x}^\mu, \mathbf{y}^\mu) \mid \mu = 1, 2, \dots, p\}$ $\mathbf{x}^\mu \in A^n$ and $\mathbf{y}^\mu \in A^m$, with $A = \{0, 1\}$, $n \in \mathbb{Z}^+$, $p \in \mathbb{Z}^+$, $m \in \mathbb{Z}^+$ and $1 < p \leq \min(2^n, 2^m)$. Also, it holds that all input patterns are different; M that is $\mathbf{x}^\mu = \mathbf{x}^\xi$ if and only if $\mu = \xi$. If $\forall \mu \in \{1, 2, \dots, p\}$ it holds that $\mathbf{x}^\mu = \mathbf{y}^\mu$, the Alpha-Beta memory will be *autoassociative*; if on the contrary, the former affirmation is negative, that is $\exists \mu \in \{1, 2, \dots, p\}$ for which it holds that $\mathbf{x}^\mu \neq \mathbf{y}^\mu$, then the Alpha-Beta memory will be *heteroassociative*.

Definition 1 (One-Hot). Let the set A be $A = \{0, 1\}$ and $p \in \mathbb{Z}^+$, $p > 1$, $k \in \mathbb{Z}^+$, such that $1 \leq k \leq p$. The k -th one-hot vector of p bits is defined as vector $\mathbf{h}^k \in A^p$ for which it holds that the k -th component is $h_k^k = 1$ and the set of the components are $h_j^k = 0$, $\forall j \neq k$, $1 \leq j \leq p$.

Remark 1 In this definition, the value $p = 1$ is excluded since a one-hot vector of dimension 1, given its essence, has no reason to be.

Definition 2 (Zero-Hot). Let the set A be $A = \{0, 1\}$ and $p \in \mathbb{Z}^+$, $p > 1$, $k \in \mathbb{Z}^+$, such that $1 \leq k \leq p$. The k -th zero-hot vector of p bits is defined as vector $\bar{\mathbf{h}}^k \in A^p$ for which it holds that the k -th component is $h_k^k = 0$ and the set of the components are $h_j^k = 1$, $\forall j \neq k$, $1 \leq j \leq p$.

Remark 2. In this definition, the value $p = 1$ is excluded since a zero-hot vector of dimension 1, given its essence, has no reason to be.

Definition 3 (Expansion vectorial transform). Let the set A be $A = \{0, 1\}$ and $n \in \mathbf{Z}^+, y \in \mathbf{Z}^+$. Given two arbitrary vectors $\mathbf{x} \in A^n$ and $\mathbf{e} \in A^m$, the expansion vectorial transform of order m , $\tau^e : A^n \rightarrow A^{n+m}$, is defined as $\tau^e(\mathbf{x}, \mathbf{e}) = \mathbf{X} \in A^{n+m}$, a vector whose components are: $X_i = x_i$ for $1 \leq i \leq n$ and $X_i = e_i$ for $n+1 \leq i \leq n+m$.

Definition 4 (Contraction vectorial transform). Let the set A be $A = \{0, 1\}$ and $n \in \mathbf{Z}^+, y \in \mathbf{Z}^+$ such that $1 \leq m < n$. Given one arbitrary vector $\mathbf{X} \in A^{n+m}$, the contraction vectorial transform of order m , $\tau : A^{n+m} \rightarrow A^m$, is defined as $\tau(\mathbf{X}, m) = \mathbf{c} \in A^m$, a vector whose components are: $c_i = X_{i+n}$ for $1 \leq i < m$.

In both directions, the model is made up by two stages, as shown in figure 5. For simplicity, we will just described the process necessary in one direction, $x \rightarrow y$.

The function of Stage 2 is to offer a $\mathbf{y}^k$ as output ($k = 1, \dots, p$) given a $\mathbf{x}^k$ as input. Now we assume that as input to Stage 2 we have one element of a set of p orthonormal vectors. Recall that the *Linear Associator* has perfect recall when it works with orthonormal vectors. In this work we use a variation of the *Linear Associator* in order to obtain $\mathbf{y}^k$ from a one-hot vector $\mathbf{h}^k$ in its k -th coordinate.

For the construction of the modified Linear Associator, its learning phase is skipped and a matrix $\mathbf{M}$ representing the memory is built. Each column in this matrix corresponds to each output pattern $\mathbf{y}^\mu$. In this way, when matrix $\mathbf{M}$ is operated with a one-hot vector $\mathbf{h}^k$, the corresponding $\mathbf{y}^k$ will always be recalled.

The task of Stage 1 is: given a $\mathbf{x}^k$ or a noisy version of it ($\tilde{\mathbf{x}}^k$), the one-hot vector $\mathbf{h}^k$ must be obtained without ambiguity and with no condition. In its learning phase, stage 1 has the following algorithm:

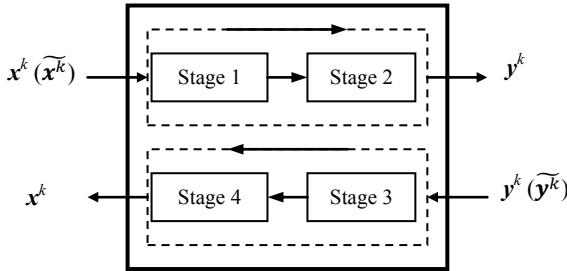


Fig. 5. Alpha-Beta BAM model scheme

1. For $1 \leq k \leq p$ do expansion: $\mathbf{X}^k = \tau^e(\mathbf{x}^k, \mathbf{h}^k)$
2. For $1 \leq i \leq n$ and $1 \leq j \leq n$: $v_{ij} = \bigvee_{\mu=1}^p \alpha(X_i^\mu, X_j^\mu)$
3. For $1 \leq k \leq p$ do expansion: $\bar{\mathbf{X}}^k = \tau^e(\mathbf{x}^k, \bar{\mathbf{h}}^k)$

4. For $1 \leq i \leq n$ and $1 \leq j \leq n$:

$$\lambda_{ij} = \bigwedge_{\mu=1}^p \alpha(\bar{X}_i^\mu, \bar{X}_j^\mu)$$

5. Create modified Linear Associator.

$$6. \mathbf{L}\mathbf{A}\mathbf{y} = \begin{bmatrix} y_1^1 & y_1^2 & \cdots & y_1^p \\ y_2^1 & y_2^2 & \cdots & y_2^p \\ \vdots & \vdots & \cdots & \vdots \\ y_n^1 & y_n^2 & \cdots & y_n^p \end{bmatrix}$$

Recall phase is described through the following algorithm:

1. Present, at the input to Stage 1, a vector from the fundamental set $\mathbf{x}^\mu \in A^n$, for some index $\mu \in \{1, \dots, p\}$.

2. Build vector: $\mathbf{u} = \sum_{i=1}^p \mathbf{h}^i$

3. Do expansion: $\mathbf{F} = \tau^e(\mathbf{x}^\mu, \mathbf{u}) \in A^{n+p}$

4. Obtain vector: $\mathbf{R} = \mathbf{V} \Delta_\beta \mathbf{F} \in A^{n+p}$

5. Do contraction: $\mathbf{r} = \tau^c(\mathbf{R}, n) \in A^p$

If $\mathbf{r}$ is one-hot vector, it is assured that $k = \mu$, then $\mathbf{y}^\mu = \mathbf{L}\mathbf{A}\mathbf{y} \cdot \mathbf{r}$. Stop. Else:

6. For $1 \leq i \leq p$: $w_i = u_i - 1$

7. Do expansion: $\mathbf{G} = \tau^e(\mathbf{x}^\mu, \mathbf{w}) \in A^{n+p}$

8. Obtain a vector: $\mathbf{S} = \mathbf{A} \nabla_\beta \mathbf{G} \in A^{n+p}$

9. Do contraction: $\mathbf{s} = \tau^c(\mathbf{S}^\mu, n) \in A^p$

10. If $\mathbf{s}$ is zero-hot vector then it is assured that $k = \mu$, $\mathbf{y}^\mu = \mathbf{L}\mathbf{A}\mathbf{y} \cdot \bar{\mathbf{s}}$, where $\bar{\mathbf{s}}$ is the negated vector of $\mathbf{s}$. Stop. Else:

11.12. Do operation $\mathbf{r} \wedge \bar{\mathbf{s}}$, where $\wedge$ is the symbol of the logical AND operator, so $\mathbf{y}^\mu = \mathbf{L}\mathbf{A}\mathbf{y} \cdot (\mathbf{r} \wedge \bar{\mathbf{s}})$. STOP.

The process in the contrary direction, which is presenting pattern $\mathbf{y}^k$ ($k = 1, \dots, p$) as input to the Alpha-Beta BAM and obtaining its corresponding $\mathbf{x}^k$, is very similar to the one described above. The task of Stage 3 is to obtain a one-hot vector $\mathbf{h}^k$ given a $\mathbf{y}^k$. Stage 4 is a modified Linear Associator built in similar fashion to the one in Stage 2.

4 Experiments and Results

The algorithm was implemented with the programming language Microsoft Visual C# 2008 Express Edition® and was tested on a PC with Intel Pentium 4® processor and 1 GB of RAM memory, the operating system was Microsoft Windows XP Professional®.

The information was taken from Oxford Parkinson's Disease Telemonitoring Dataset. This dataset is composed of a range of biomedical voice measurements from 42 people with early-stage Parkinson's disease recruited to a six-month trial of a telemonitoring device for remote symptom progression monitoring.

The description of the attributes is shown in Table 6. The total number of recordings is 195, from which 147 belong to the Parkinson class (class 1) and 48 to the non-Parkinson class (class 2). The database was created in June 2008.

The first step for the classification task is the designing of the Alpha-Beta BAM. Therefore, the memory has to be trained with the set of patterns which are contained in the database. As we can observe in Table 6, the attributes are represented by integer and real numbers, thus they have to be codified in order to obtain patterns with binary values. Each row in the database is binarized as follows (see figure 6): each feature is codified with the Johnson-Möbius code, and then they are concatenated to form a single vector which represents the input pattern x^k , in this case $k = 1, 2, \dots, n$ with $n=195$.

In this work, the algorithm of the learning phase (introduced in Section 3) was just applied for one direction: $x \rightarrow y$, because this application does not have a bidirectional behavior.

Due to Alpha-Beta BAM shows a correct recall for all the training patterns, we first trained the memory with the 195 records.

Table 6. Attribute information

Attribute	Description
subject#	Integer that uniquely identifies each subject
age	Subject age
sex	Subject gender '0' - male, '1' - female
test_time	Time since recruitment into the trial. The integer part is the number of days since recruitment
motor_UPDRS	Clinician's motor UPDRS score, linearly interpolated
total_UPDRS	Clinician's total UPDRS score, linearly interpolated
Jitter:%, Abs, RAP, PPQ5, DDP	Several measures of variation in fundamental frequency
Shimmer: dB, APQ3, APQ5, APQ11, DDA	Several measures of variation in amplitude
NHR,HNR	Two measures of ratio of noise to tonal components in the voice
RPDE	A nonlinear dynamical complexity measure
DFA	Signal fractal scaling exponent
PPE	A nonlinear measure of fundamental frequency variation

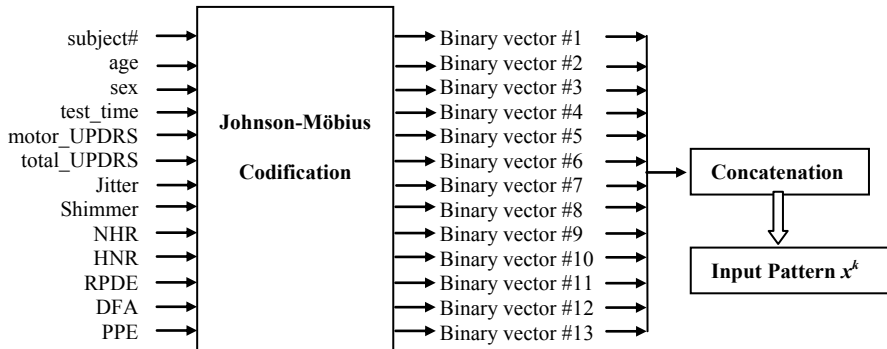


Fig. 6. Binarization of attributes using Johnson-Möbius codification

In the recalling phase (introduced in Section 3), each pattern is codified using the process showed in figure 6 and presented to the Alpha-Beta BAM. As such as we expected every pattern was correctly classified.

For testing the performance of Alpha-Beta BAM algorithm, three methods were used: Hold-Out, K-Fold Cross-Validation and Leave One Out.

Hold-Out Test

The complete database was divided into two sets, one of them was the training set and the other was the testing set. We selected randomly the elements contained in each set. The size of the sets was varying from 2% for training and 98% for testing to 98% for training and 2% for testing. For each size we performed 15 calculations. Table 7 shows the results obtained.

Most of the works performing classification take the effectiveness with the size of the training set of 80%. In our case, we obtained a 98.13 % for that size.

Table 7. Results from Hold-Out test

Size for training (%)	Effectiveness (%)
2	28.73
10	61.53
20	82.40
30	86.20
40	88.27
50	91.27
60	96.00
70	96.87
80	98.13
90	98.93
98	99.87

K-fold Cross-Validation Test

In this case, the whole database was divided into 10 sets (then K=10) with the same number of elements. We attempted to maintain the proportion of the number of the records in both classes because the database is not balanced. Therefore, we had 9 sets with 20 records: 15 from Parkinson class and 5 from non-Parkinson class, and 1 set with 15 records: 12 from class 1 and 3 from class 2.

Table 8. Results from K-fold Cross-Validation test with K=10

Set	Effectiveness (%)
K1	98
K2	98
K3	97
K4	100
K5	100
K6	96
K7	98
K8	98
K9	99
K10	98
Average	98.2

We took the first set K1 for testing and the remaining sets were for training, afterwards, the set K2 was used for testing and the others for training and so on. The same way, we performed 20 calculations for every test. Table 8 shows the results.

We can observe from the table that the worst effectiveness is equal to 96% while the best was of 100% giving an average of 98.2%.

Leave One Out Test

For this test, the training set contained all the records from the database but one, which was used for testing. Therefore, we performed 195 calculations. With this test, the effectiveness was of 95.19%. Table 9 shows the results.

Table 9. Results from Leave One-Out test

Class	Effectiveness
Parkinson	94.55 (8 mistakes)
Non-Parkinson	95.83 (2 mistakes)
Total	95.19%

Table 10 shows the results from the three testing methods.

Table 10. Results from K-fold Cross-Validation test with K=10

Method	Effectiveness (%)
Hold Out (80% training)	98.13
K-fold Cross-Validation (K=10)	98.2
Leave One Out	95.19
Average	97.17

From Table 10 we can observe that the average of effectiveness of the 3 testing methods was of 97.17%. But we cannot assure this result is good without compare it with other classification algorithms. Therefore, in Table 11 the results from four other methods using the same database are showed. They used different approaches to the one used in this work: Probabilistic Neural Networks (PNN), Mutual Information (MI), Support Vector Machine (SVM) and a combination of Neural Networks (NN) with SVM.

Table 11. The results of effectiveness of the five methods: PNN,MI,SVM, NN-SVM and Alpha-Beta BAM

Method	Effectiveness (%)
Probabilistic Neural Network	81.74
Mutual Information	81.53
Support Vector Machine	62.217
NN-SVM	93.33
Alpha-Beta BAM	97.17

The work that used MI approach tested its algorithm with Leave-One-Out test obtaining 81.53% of classification. With the same testing algorithm we obtained the 95.19% of effectiveness. The result from PNN algorithm was of 81.74% of effectiveness using Hold Out test with 70% for training and 30% for testing. For the same test we obtained 96.87% of effectiveness. In the work where SVM is used, authors applied K-fold Cross-Validation varying K from 2 to 10, and they find the best result (62.217% of effectiveness) with K=10. Alpha-Beta BAM approach shows, with the same test conditions, an effectiveness of 98.2. Finally, the effectiveness of NN-SVM method was of 93.33% but the testing method is not mentioned. However, any of the results from the three testing methods achieved by Alpha-Beta BAM is higher than 93.33%.

5 Conclusions

Parkinson's disease has become a frequent neurodegenerative disease which has affected many people in the world. Therefore, it is important to rely on a system capable of diagnosing patients with Parkinson's disease.

Alpha-Beta Associative models have shown to be an option as a tool for many applications, and recently they have been used specifically as classifiers improving the results by means of Johnson-Möbius modified code.

We implemented Alpha-Beta BAM algorithm together with Johnson-Möbius codification to classify patients with Parkinson's disease. We tested Alpha-Beta BAM using three methods: Hold Out, K-fold Cross-Validation (K=10) and Leave One Out. We compared the obtained results with other different approaches: Probabilistic Neural Networks, Mutual Information, Support Vector Machine and a combination of Neural Networks and SVM. Our approach showed the best results of performance surpassing the Mutual Information approach with 13.66% of effectiveness and Probabilistic Neural Networks with 15.13% of effectiveness. The difference between SVM and our BAM was of 35.983% of accuracy, and Alpha-Beta BAM showed better results than NN-SVM.

These results reassert the fact that Associative Models are a good alternative for the task of classification over other conventional approaches.

Acknowledgments. The authors would like to thank the Instituto Politécnico Nacional (COFAA and SIP), and SNI for their economical support to develop this work.

References

1. Goetz, C.G.: Early Iconography of Parkinson's Disease. In: Handbook of Parkinson's Disease, 4th edn., Informa Healthcare, New York (2007)
2. Factor, S.A., Weiner, W.J.: Parkinson's Disease: Diagnosis and Clinical Management, 2nd edn. Demos, New York (2008)
3. Lieberman, A.: 100 Questions and Answers about Parkinson Disease. Jones and Bartlett Publishers, Sudbury (2003)

4. Jancovic, J.: Parkinson's disease: clinical features and diagnosis. *J. Neurol. Neurosurg. Psychiatry* 79, 368–376 (2008)
5. Adler, C.H., Ahlskog, J.E.: *Parkinson's Disease and Movements Disorders: Diagnosis and Treatment Guidelines for the Practicing Physician*. Humana Press, New Jersey (2000)
6. Unified Parkinson Disease Rating Scale (UPDRS),
<http://www.uninet.edu/neurocon/neurologia/escalas/parkinson.html#UPDRS>
7. World Health Organization, <http://www.who.int>
8. National Parkinson Foundation,
<http://www.parkinson.org/parkinson-s-disease.aspx>
9. Acton, P.D., Newberg, A.: Artificial network classifier for the diagnosis of Parkinson's disease using [99mTc]TRODAT-1 and SPECT. *Phys. Med. Biol.* 51(12), 3057–3066 (2006)
10. Ericsson, A., Lonsdale, M.N., Astrom, K., Edenbrandt, L., Friberg, L.: Decision Support System for the Diagnosis of Parkinson's Disease. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) *SCIA 2005*. LNCS, vol. 3540, pp. 740–749. Springer, Heidelberg (2005)
11. Yoshida, H., Nakagawa, K., Anai, H., Horimoto, K.: Exact Parameter Determination for Parkinson's Disease Diagnosis with PET Using an Algebraic Approach. In: Anai, H., Horimoto, K., Kutsia, T. (eds.) *Ab 2007*. LNCS, vol. 4545, pp. 110–124. Springer, Heidelberg (2007)
12. Keijsers, N.L.W., Horstink, M.W.I.M., Gielen, C.C.A.M.: Automatic, unsupervised classification of dyskinesia in patients with Parkinson's Disease. In: Kaynak, O., Alpaydin, E., Oja, E., Xu, L. (eds.) *ICANN 2003 and ICONIP 2003*. LNCS, vol. 2714, Springer, Heidelberg (2003)
13. Djurić-Jovičić, M., Jovičić, N.S., Milovanović, I., Radovanović, S., Kresojević, N., Popović, M.B.: Classification of walking patterns in Parkinson's disease patients based on inertial sensor data. *Neural Network Applications in Electrical Engineering (NEUREL)*, 3–6 (2010)
14. Parkinsons Telemonitoring Data Set,
<http://archive.ics.uci.edu/ml/datasets/Parkinsons>
15. Ene, M.: Neural network-based approach to discriminate healthy people from those with Parkinson's disease. *Annals of the University of Craiova, Math. Comp. Sci. Ser.* 35, 112–116 (2008)
16. Sakar, O., Kursun, O.: Telediagnosis of Parkinson's Disease Using Measurements of Disphonia. *J. Med. Syst.* 34, 591–599 (2010)
17. Bhattacharya, I., Bhatia, M.P.S.: SVM classification to distinguish Parkinson disease patients. In: *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing A2CWic 2010, India* (2010)
18. Gil, D., Johnson, M.: Diagnosing Parkinson by using Artificial Neural Networks and Support Vector Machines. *Global Journal of Computer Science and Technology* 9(4), 63–71 (2009)
19. Acevedo, E., Yáñez, C., López, I.: Alpha-Beta Bidirectional Associative Memories: Theory and Applications. *Neural Processing Letters* 26, 1–40 (2007)
20. Ritter, G.X., Sussner, P., Diaz de León, J.L.: Morphological Associative Memories. *IEEE Transactions on Neural Networks* 9, 281–293 (1998)
21. Yáñez-Márquez, C.: *Associative Memories Based on Order Relations and Binary Operators* (in Spanish). PhD Thesis. Centro de Investigación en Computación, Mexico (2002)
22. Mano, M.: *Diseño digital*, 16–26, 292–294. Prentice Hall, Englewood Cliffs (2001)
23. Flores, R.: *Johnson-Möbius modified code-based Alpha-Beta Associative Memories* (in Spanish). MD thesis, Centro de Investigación en Computación, Mexico (2006)

Thermal Video Analysis for Fire Detection Using Shape Regularity and Intensity Saturation Features

Mario I. Chacon-Murguia and Francisco J. Perez-Vargas

Visual Perception Applications on Robotic Lab, Chihuahua Institute of Technology
mchacon@itchihuahua.edu.mx, fjperez@ieee.org

Abstract. This paper presents a method to detect fire regions in thermal videos that can be used for both outdoor and indoor environments. The proposed method works with static and moving cameras. The detection is achieved through a linear weighted classifier which is based on two features. The features are extracted from candidate regions by the following process; contrast enhance by the Local Intensities Operation and candidate region selection by thermal blob analysis. The features computed from these candidate regions are; region shape regularity, determined by Wavelet decomposition analysis and region intensity saturation. The method was tested with several thermal videos showing a performance of 4.99% of false positives in non-fire videos and 75.06% of correct detection with 7.27% of false positives in fire regions. Findings indicate an acceptable performance compared with other methods because this method unlike other works with moving camera videos.

Keywords: fire detection, thermal image processing, image segmentation.

1 Introduction

Fire detection is vital for early fire detection systems as well as in fire control. Fire detection systems may contribute to detect hazards situations that may reduce the danger for human lives as well as negative economic impact. Most of conventional fire detection systems are based on particle sampling techniques, temperature monitoring and air transparency. Unfortunately, these systems need to be located close to the fire and not always detect fire but smoke which not necessarily indicates fire. Conventional fire detectors used in buildings depend on the detection of smoke or fire particles [1], therefore they are not suitable for large areas. Besides, they cannot provide information of size, intensity or location of fire. These situations justify the research of fire detection based on vision systems which overcome the previous disadvantages of conventional methods. There are vision systems that work on the visible spectrum [1]-[4] analyzing color and movement but lack of robustness because of a high false positive rate due to colors similar to fire or illumination problems because of reflections. Also, conventional cameras cannot generate relevant images once dense smoke appears in the scene. Therefore, the proposed method described in this paper works with IR images acquired with a thermal camera. IR cameras have the advantage of generating relevant information even under smoke

conditions and fire detection with low radiation in the visible spectrum generated by alcohol and hydrogen [5].

The contributions of the work reported in this paper are the following. The proposed method can be used as an indoor as well as an outdoor fire detector system. The camera does not need to be close to the fire. Besides, the proposed method considers typical flame characteristics like irregular contours and the peculiarity of being the dominant heat source in the scene. Irregular contour features are converted to contour distance vectors. Other characteristic used in the method is related to the capacity of the flames to generate a large amount of heat which in turn may produce saturation levels in the camera scale. These features can be computed in static and moving IR cameras which represents an important advantage with respect other methods based only on static cameras.

2 Fire Detection Method

A general description of the proposed fire detection method is the following. Prospective fire blob detection is achieved based on maximum temperature, [6]. Blob contrast enhancement is done using the technique described in [7]. Then a binarization threshold is computed and an area filtering is performed to define fire candidate blobs. The decision over those candidate blobs is finally made based on region shape regularity, determined by Wavelet decomposition analysis and region intensity saturation.

2.1 Image Preprocessing

The thermal images are processed as gray level images. Besides, in order to eliminate some information added by thermal images (date, scale, etc.) a ROI is defined on the original image. The next step is to enhance the contrast of the ROI by the Local Intensities Operation, LIO, in its intensity brightening operation (IBO) mode, [7]. With this method high gray level values (high temperature values) are enhanced. The IBO operator at coordinates (x,y) is defined by

$$G(x, y) = \prod_{k=0}^8 z_k . \quad (1)$$

where z_k are the 8-neighbord pixels of the pixel z_0 located at (x,y) . Figure 1 shows examples before and after of the application of the IBO operator.

2.2 Prospective Fire Blobs Location

At this point, the image information is suitable to try to detect fire blob candidates. These blobs are found by determining the maximum gray level in the image, which in turn may correspond to high temperature areas

$$gmax = \{(x, y) | \max \{G(x, y)\}\} . \quad (2)$$

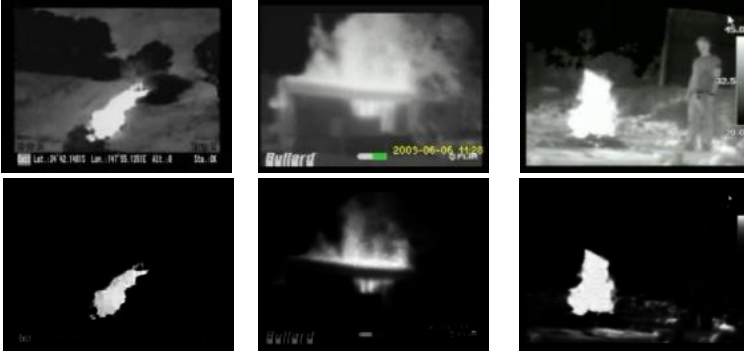


Fig. 1. Original images and fire blob enhanced after the IBO operator

Finding $gmax$ does not necessary warranties that it corresponds to a fire blob. Therefore a minimum level for $gmax$ needs to be determined. That is, a prospective fire blob must hold $gmax > \delta_l$. Considering non-fire as well as fire frames the value of δ_l was determined as 220. If a prospective blob is located in the frame, the next step is to define the region of the fire blob. This area is defined as

$$B(x, y) = \begin{cases} 1 & \text{if } G(x, y) > \delta_2 * gmax \\ 0 & \text{if } G(x, y) < \delta_2 * gmax \end{cases} \quad (3)$$

where δ_2 is a percentage threshold to define the pixels corresponding to the fire area. δ_2 was defined to be a value between 70% and 85% based on experimentation.

The binary image $B(x, y)$ may contain noise regions, some false fire areas. In order to get rid of them an area filter is applied to $B(x, y)$

$$F_k(x, y) = \{B_k(x, y) \mid Area(B_k(x, y)) > 40 \text{ and } Area(B_k(x, y)) > 0.2\alpha\} \quad (4)$$

where $F_k(x, y)$ is the k prospective fire blob and α is the area of the largest region in $B(x, y)$. Figure 2 illustrates the process to determine prospective fire regions.

The previous thresholds and parameters were determined by statistical analysis using information of different videos taken with different cameras and conditions, therefore it is expected that the statistical validity is hold for other videos and cameras.

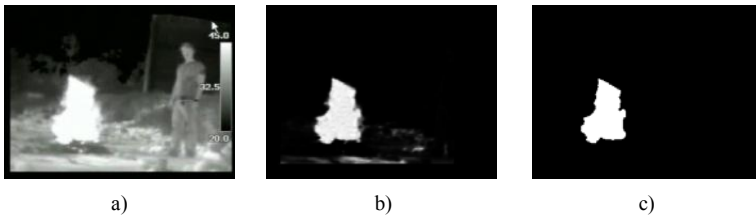


Fig. 2. a) Original image, b) Pre-processed image, c) Candidate blob

2.3 Feature Extraction

At this point the method has generated a set of fire candidate blobs. Therefore, it is necessary to design a classifier to determine if the prospective region $F_k(x,y)$ corresponds to a fire region. The features used in the classifier are related to the region shape regularity, determined by Wavelet decomposition analysis and region intensity saturation. The fire regions are mainly distinguishable from common objects or man-made objects as well as persons because fire regions present highly irregular contours, Figure 3 illustrates these cases. The irregularity analysis is performed in the Wavelet domain [8] as follows. A 1D signature $S[l]$ is obtained for $F_k(x,y)$ [9]. $S[l]$ contains the euclidean distance from the center of mass of $F_k(x,y)$ to its contour in function of the angle θ for $\theta = 0$ to 360° . The Wavelet analysis is done according to the high and low pass filters proposed in [5],

$$a[l] = s[l] * h[l] \quad \text{and} \quad d[l] = s[l] * g[l] . \quad (5)$$

where

$$h[l] = \left\{ -\frac{1}{4}, \frac{1}{2}, -\frac{1}{4} \right\} \quad \text{and} \quad g[l] = \left\{ \frac{1}{4}, \frac{1}{2}, \frac{1}{4} \right\} . \quad (6)$$

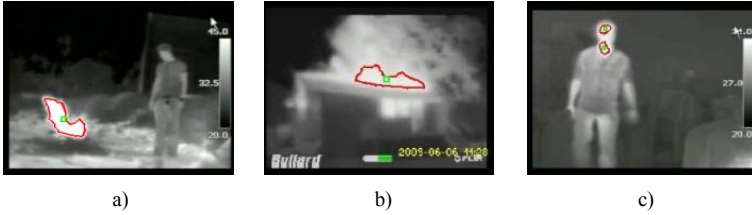


Fig. 3. Center of mass and contour of candidate regions a) Bonfire, b) House on fire, c) Person

The Figure 4a shows the signatures as well as the Wavelet decomposition of candidate regions of Figure 3a, fire region. On the other hand, Figure 4b illustrates the case of a no-fire region of Figure 3c. It can be observed the differences of the signatures on the scales on both figures. This difference can be computed through an irregularity contour parameter β expressed as,

$$\beta = \sum_l |d[l]| / \sum_l |a[l]| . \quad (7)$$

The irregularity parameter is normalized in order to be invariant to amplitude values. In this way, small values of β correspond to non-fire regions.

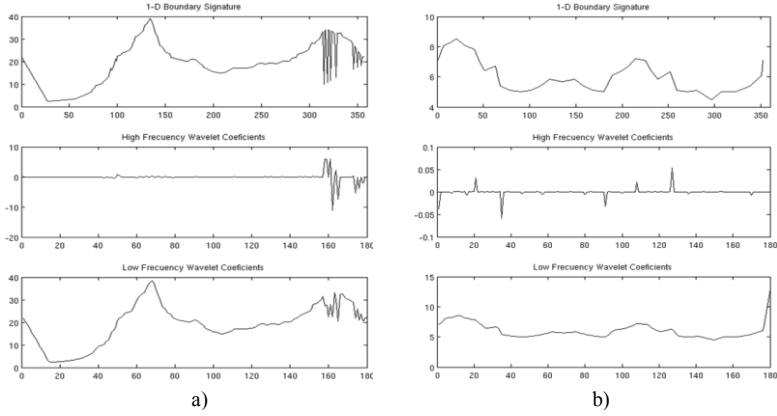


Fig. 4. Signatures of candidate regions and their Wavelet decomposition of Figures, a) 3a, b) 3c

The second feature, intensity saturation, is related to the high temperature values associated to the fire. Since the fire region is the most prominent source of heat in the scene the pixel blob associated to it tends to reach the saturation level of the thermal camera [4]. The intensity saturation feature is defined as

$$\sigma = \pi / \tau . \quad (8)$$

where

$$\pi = \left\| \{ g(x, y) \mid (x, y) \in F_K(x, y) > \delta_3 \} \right\| . \quad (9)$$

$$\tau = \left\| \{ g(x, y) \mid (x, y) \in F_K(x, y) \} \right\| . \quad (10)$$

$g(x, y) \in G(x, y)$, $\| \cdot \|$ stands for set cardinality. The threshold δ_3 is computed automatically for each frame under analysis and must be close to the maximum level allowed for the radiometric resolution of the camera and in consequence greater than zero, that is

$$\delta_3 = \max \left\{ G(x, y) \mid (x, y) \in F_k(x, y) \right\} - 5 . \quad (11)$$

Figure 5 illustrates the behavior of π for a fire and a non-fire blob. As it was expected the intensity saturation level is greater in the fire region than in the non-fire blob, $\sigma = 0.9125$ and $\sigma = 0.3072$ respectively.



Fig. 5. Illustration of saturation in a) Fire, b) Non-fire

2.4 Classification Scheme

As a first approach and in order to keep the computational cost low a linear classifier was chosen. Future work will include analysis with other classifiers. The classification of the candidate regions is determined by the following rule

$$F_k(x,y) \text{ is fire if } \gamma > 0.275 . \quad (12)$$

where

$$\gamma = w_1 \beta + w_2 \sigma . \quad (13)$$

w_1 and w_2 are weighting factors with values 0.75 and 0.25 respectively. These values were defined to represent the confident impact of the discrimination power of β and σ by analysis of their distribution values based on 6291 and 847 candidate regions of fire and non-fire regions. The threshold of 0.275 in Eq. (12) was also determined by statistical analysis of the mean values of the both distributions.

3 Results and Conclusions

3.1 Experimental Results

The method was tested in thermal videos with a resolution of 320x240 at 15 FPS acquired with a camera Fluke Ti45 working in the bands $8\mu m$ to $14\mu m$. The video set includes different types of situations in order to test the robustness of the method. Besides, a set of Internet videos acquired with a moving camera, low contrast and multiple fire regions were also included. Table 1 shows the information of the video data set. The complete data set and obtained results are available in <http://dspvisionlab.itch.edu.mx/fjperez>.

Table 1. Video data set

Video	Frames	Description	Camera
NoFire 1	230	Two walking people in a room	Static
NoFire 2	1692	Controlled fire, lighter	Static
NoFire 3	815	Pencil type soldering tin	Static
NoFire 4	182	Walking person in a room	Static
Fire 1	515	Fire with Blue-Red palette	Static
Fire 2	286	Fire	Moving
Fire 3	740	Fire close to a person	Static
Fire 4	1081	Fire with Blue-Red palette	Static
Fire 5	1551	Firefighter controlling an indoor fire	Moving
Fire 6	742	Fire video acquired from a helicopter	Moving
Fire 7	596	Interior fire and explosion	Static
Fire 8	1216	House on fire, part 1	Moving
Fire 9	1185	House on fire, part 2	Moving

Figure 6 shows an example of a non-fire high temperature and fire regions processing including their features values. These values are consistent with the information aforementioned as well as the justification of the weighting factors.

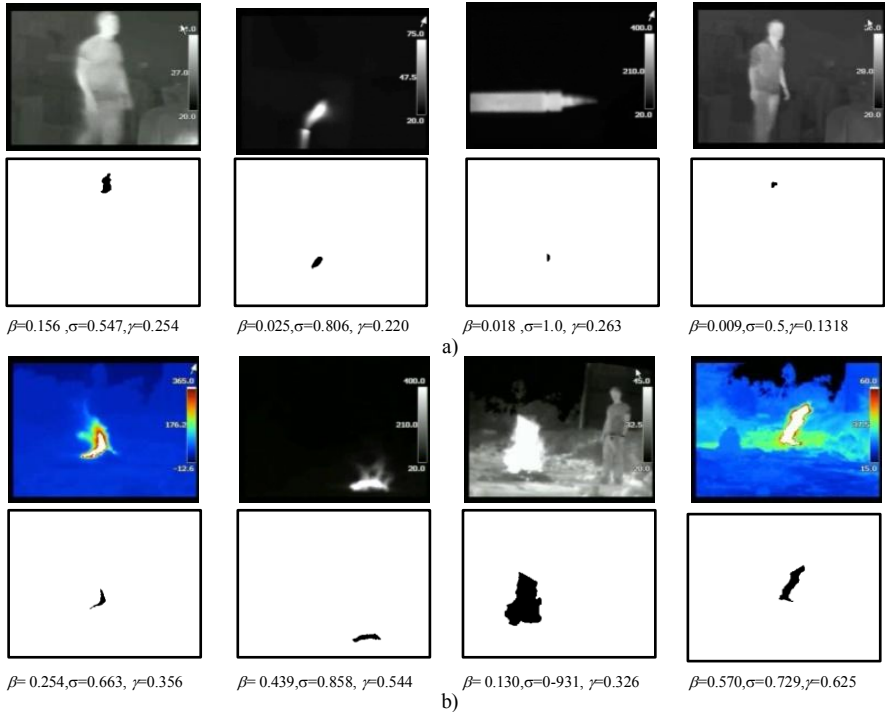


Fig. 6. a) Non-fire high temperature and, b) controlled fire regions with features values

On the other hand, Figure 7 shows cases of the Internet videos. These examples show the proposed method robustness under extreme conditions, low contrast, multiple fire regions and moving camera. These conditions make to fail other documented methods because they are based on fixed position pixel and temporal information.

3.2 Performance Metrics

The performance of the proposed method is presented in Table 2 and 3. A comparison with other methods was not directly achieved because data used by other methods was not available. The information provided is; number of processed frames, frames with fire, number of frames with fire detected, false positives and the percentages of hits, miss, and false positives.

Results in Table 2 indicate that for non-fire videos the method performs well with an average of 4.99% of false positives. In regards fire detection Table 3 shows that the average percentage of hits is 75.06%. Video 5 presents a high false positive rate because the fire region is very hard to define even by a person. The average performance is acceptable compared with other works [1][3] that report 66.4%, 86.1%, for true positives, 4.9% , 0.4% false positives, and 23.7% and 13.2% in missing rate. These methods do not consider the moving camera case or the multi-region fire situation. They work on the visible spectrum and do not use the same set of videos used in this work.

In conclusion, we can say that the proposed method has acceptable results in most of the tested situations and also compared with other methods based on color and temporal information which present a high false alarm rate. Also, the method shows robustness in moving camera videos which is not supported by methods based on temporal information. The current processing speed is 10.7 fps running in Matlab, therefore the method is guaranteed to run in real time.

For future work, we are currently developing a more sophisticated classification scheme based on Fuzzy Logic using the same features presented in this paper.

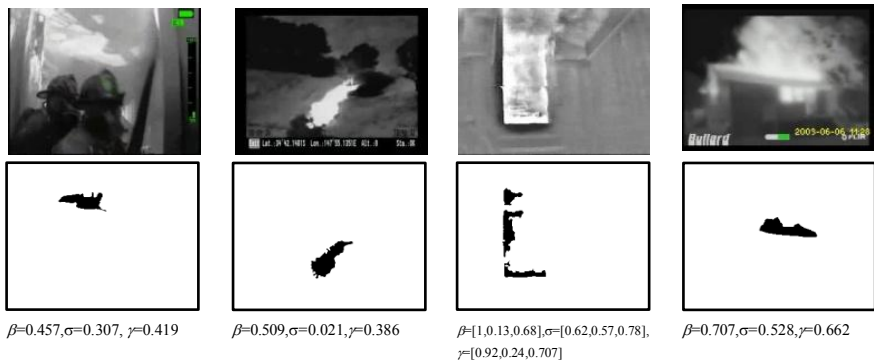


Fig. 7. Examples of extreme fire conditions processing and their features values

Table 2. Non-fire cases performance

Video	Frames		%False positives
	Processed	Fire False Positives	
NoFire 1	115	0 15	13.04%
NoFire 2	846	0 15	1.77%
NoFire 3	408	0 21	5.15%
NoFire 4	91	0 0	0.00%
Average			4.99%

Table 3. Fire cases performance

Video	Frames						
	Processed	Fire	Hits	False Positives	Hit%	Miss%	False Positives%
Fire 1	257	209	144	0	68.90%	31.10%	0.00%
Fire 2	143	138	107	0	77.54%	22.46%	0.00%
Fire 3	370	218	180	3	82.57%	17.43%	1.97%
Fire 4	540	442	366	2	82.81%	17.19%	2.04%
Fire 5	775	630	390	89	61.90%	38.10%	61.38%
Fire 6	371	154	92	0	59.74%	40.26%	0.00%
Fire 7	298	296	293	0	98.99%	1.01%	0.00%
Fire 8	608	588	370	0	62.93%	37.07%	0.00%
Fire 9	592	590	473	0	80.17%	19.83%	0.00%
Average					75.06%	24.94%	7.27%

Acknowledgements. The authors thanks to Fondo Mixto de Fomento a la Investigación Científica y Tecnológica CONACYT-Gobierno del Estado de Chihuahua, by the support of this research under grant CHIH-2009-C02-125358. Special thanks to the SOFI de Chihuahua by providing the thermal equipment used in this research.

References

1. Toreyin, B.U., Dedeoglu, Y., Gudukbay, U., Cetin, A.E.: Computer Vision Based Method for Real-time Fire and Flame Detection. *Pattern Recognition Letters* 27, 49–58 (2006)
2. Phillips III, W., Shah, M., Lobo, N.V.: Flame Recognition in Video. *Pattern Recogn. Letters* 23(3), 319–327 (2002)
3. Ko, B.C., Cheong, K.H., Nam, J.Y.: Fire Detection Based on Vision Sensor and Support Vector Machines. *Fire Safety Journal* 44, 322–329 (2009)
4. Marbach, G., Loepfe, M., Brupbacher, T.: An Image Processing Technique for Fire Detection in Video Images. *Fire Safety Journal* 41, 285–289 (2006)
5. Uğur, B., Gökberk, R., Dedeoğlu, Y., Enis, A.: Fire Detection in Infrared Video Using Wavelet Analysis. *Optical Engineering* 46, 067204 (2007)
6. Kamgar-Parsi, B.: Improved image thresholding for object extraction in IR images. *IEEE International Conference on Image Processing* 1, 758–761 (2001)
7. Heriansyah, R., Abu-Bakar, S.A.R.: Defect detection in thermal image for nondestructive evaluation of petrochemical equipments. In: *NDT & E International*, vol. 42(8), pp. 729–774. Elsevier, Amsterdam (2009)
8. Chacon, M.I.: *Digital Image Processing* (in spanish). Editorial Trillas (2007)
9. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*, 2nd edn., pp. 648–649. Prentice-Hall, Englewood Cliffs (2002)

People Detection Using Color and Depth Images*

Joaquín Salas¹ and Carlo Tomasi²

¹ Instituto Politécnico Nacional

jsalasr@ipn.mx

² Duke University

tomasi@cs.duke.edu

Abstract. We present a strategy that combines color and depth images to detect people in indoor environments. Similarity of image appearance and closeness in 3D position over time yield weights on the edges of a directed graph that we partition greedily into *tracklets*, sequences of chronologically ordered observations with high edge weights. Each tracklet is assigned the highest score that a Histograms-of-Oriented Gradients (HOG) person detector yields for observations in the tracklet. High-score tracklets are deemed to correspond to people. Our experiments show a significant improvement in both precision and recall when compared to the HOG detector alone.

1 Introduction

The detection of human beings from visual observations is a very active research area. The recent introduction of inexpensive depth sensors that work at frame rate offers new opportunities to address this difficult problem. In this paper, we combine depth and color data from a single sensor to track and classify people. More specifically, we introduce a directed graph whose edges connect chronologically ordered observations. Weights on the graph capture similarity of appearance and closeness in space, and a greedy traversal of the graph produces *tracklets*, that is, chronological sequences of observations that are likely to correspond to the same person. Each tracklet then receives a score from a color-based person detector from the literature [1]. Tracklets with scores exceeding a predefined threshold are deemed to correspond to people. Our experiments show that our strategy reduces the number of detected false positives by a factor of fifty, while increasing the detection of true positives threefold. The rest of the

* Thanks to Julian (Mac) Mason for his gentle introduction to the Kinect, including his help for calibrating the sensor and obtaining the first set of images. This work was supported by the Consejo Nacional de Ciencia y Tecnología under Grant No. 25288, the Fulbright Scholarship Board, and the Instituto Politécnico Nacional under Grant No. 20110705 for Joaquín Salas, and the National Science Foundation under Grant No. IIS-1017017 and by the Army Research Office under Grant No. W911NF-10-1-0387 for Carlo Tomasi.

paper is structured as follow. After a brief review of related work, Section 3 describes a method to extract foreground objects using depth information. Then, Section 4 discusses the creation of tracklets, and Section 5 presents results on three color/depth video sequences. Comparison with ground truth data illustrates the benefits of our approach when compared to HOG detection alone. A concluding Section suggests directions for future research.

2 Previous Work

An account of early efforts on people tracking can be found in [7]. These include the analysis of parts of the body, both internal or external, as well as dynamical characteristics, such as the gait. Some of the first results can be traced back to the Seventies [12], when psychophysical studies [11] showed that humans could perceive people based on pure motion data. Prompted in part by security considerations [20], new techniques, protocols and standards have emerged in the past decade. Some approaches have used silhouettes [5] or body-part matching [14, 21, 23, 22]. The combination of cascades of increasingly complex classifiers has produced fast and robust recognition algorithms [28] for relatively stylized person poses. Features for tracking people include the Scale Invariant Feature Transform (SIFT) [13], [15], Haar-like wavelets [29], shape [33], and Histograms of Oriented Gradients (HOG) [1]. The latter have proven to be particularly successful. To build a HOG descriptor, the window of interest in an image is subdivided into a grid of cells, and a histogram of the orientations of luminance gradients is computed in each cell. The histograms are normalized and concatenated into a single vector for the whole window. A linear Support Vector Machine (SVM) [27] classifies the resulting vectors into person or non-person. This work was later extended [2] to include the use of motion. Motion information had been used in other work as well [29], [6]. SVMs have been used with other descriptors for whole bodies [16] or body parts [19]. Schwartz *et al.* [25] further incorporated texture information.

Some researchers have combined spatial and light intensity information to detect people. For instance, Zhao and Thorpe [34] use a stereo system to segment the silhouettes that are fed to a neural network that detects pedestrians. Xu and Fujimora [32] also extract body silhouettes but with a time-of-flight device. The use of body, the whole or just parts of it, has proven to increase the robustness of the detection and tracking methods. Consider for example the strategy proposed by Muñoz *et al.* [17] where there is the combined use of a face detector and depth information to track people. Javed *et al.* [10] instead combine color with position information inferred from the locations of multiple cameras. In our work, we use similar principles for combining color and position information. However, we work in the field of view of a single color/depth sensor, and derive position information from a depth map through background subtraction. In addition, we also run a HOG classifier on every color frame, and propagate the best scores it generates to all observations in the same tracklet. Thus, we classify one tracklet at a time, rather than one window at a time. While this approach propagates

both true positives and false positives, our reliance on the *best* detection result in each tracklet ensures that the HOG classifier is given the opportunity to operate on body poses that fit the HOG model particularly well. The good results of our experiments in Section 5 show the validity of our approach.

3 Detection of Foreground Objects

We first classify the measurements $X = \{\mathbf{x}_1, \dots, \mathbf{x}_m\}$ from a depth sensor, where $\mathbf{x}_k = [x_k, y_k, z_k]^T$, into background $\mathcal{B}$ and foreground $\mathcal{F}$. To this end, a Gaussian background model is used to detect the foreground by Maximum A Posteriori (MAP) estimation. The resulting foreground points are then grouped into separate objects by connected component analysis. For our purposes, we divide the tridimensional space into equally spaced bins centered at $\mathcal{X} = \{\bar{x}_1, \dots, \bar{x}_a\}$, $\mathcal{Y} = \{\bar{y}_1, \dots, \bar{y}_b\}$, and $\mathcal{Z} = \{\bar{z}_1, \dots, \bar{z}_c\}$ with grid spacing Δx , Δy , and Δz . At the workspace boundaries, the bins extend to either ∞ or $-\infty$. In the following, $\mathcal{N}$ is a function that counts the number of observations that fall into each of the bins of a histogram.

3.1 Planar Background Elimination

Similarly to Vruble *et al.* [30], we assume that the workspace has either a flat floor or a flat ceiling. Furthermore, we assume that the number of points describing either one of these structures is a significant fraction of the points in the depth map. We then compute the sensor roll and pitch angles that produce a maximum bin value over the marginals on the vertical axis. Specifically, let $h(j, \alpha, \beta) = \mathcal{N}(|\bar{y}_j - y| \leq \Delta y/2)$ be the marginal histogram along the vertical direction after a rotation of the reference system by roll and pitch angles α and β . The rotation that maximizes the number of points in the most populated bin, that is, $(\alpha, \beta) = \arg \max_{\alpha, \beta} \max_j h(j, \alpha, \beta)$, can be estimated using the Nelder-Mead or Simplex method [18]. For efficiency, the points below the floor and above the ceiling are deleted after this rotation.

3.2 Background Model

The Occupancy Grid framework [4] provides a suitable platform for background subtraction. Let $s(\mathbf{x})$ be a foreground/background map for the spatial coordinates $\mathbf{x} \in X$, with $p(s(\mathbf{x}) = \mathcal{F}) + p(s(\mathbf{x}) = \mathcal{B}) = 1$. The probability that a particular space position $\mathbf{x} = [x, y, z]^T$ is part of the background is

$$p(s(\mathbf{x}) = \mathcal{B}|z) \propto p(z|s(\mathbf{x}) = \mathcal{B})p(s(\mathbf{x}) = \mathcal{B}). \quad (1)$$

Similarly to Gordon *et al.* [9], who presented a method to combine dense stereo measurements with color images, we model the background with a mixture of Gaussians and detect the foreground as those points that are more than 3σ away from the nearest background mode.

3.3 Foreground Objects

We extract foreground objects by connected components with 26-connectivity in 3D space, while reasoning about the positions of the detected objects relative to the sensor. Let $\mathcal{H}$ be a histogram constructed out of the points in X , such that $\mathcal{H}(i, j, k) = \mathcal{N}(|\bar{x}_i - x| \leq \Delta x/2, |\bar{y}_j - y| \leq \Delta y/2, |\bar{z}_k - z| \leq \Delta z/2)$. Let $v(i, j, k)$ be an indicator variable that is 1 whenever $\mathcal{H}(i, j, k) > 0$ and 0 otherwise. Objects correspond to connected components in $v(i, j, k)$. Finally, we eliminate clusters that are smaller than a depth-dependent threshold of the form $\tau(d) = \rho e^{-\nu d}$ that models the fact that the size of an object decreases with its distance d from the sensor. The values of ρ and ν are found by data fitting on training samples. Each output *blob* is given in the form of the tightest axis-aligned box around each component.

4 Combining Detections

To combine measurements of depth and appearance, we use depth for tracking blobs across frames and connecting them into *tracklets*, and we use the HOG detector [1] in one of its available implementations [3,31] to assign scores to individual blobs. The highest score on each tracklet is then propagated to all the blobs on that tracklet. Blobs that are in tracklets with a score that exceeds a given threshold are classified as people. In this Section we describe our construction of tracklets.

Adapting the framework proposed by Javed *et al.* [10], in our case for a single camera, let k_i^j be a binary indicator for the hypothesis that two observations $O_i = \{\mathbf{f}_i, \mathbf{x}_i, t_i\}$ and $O_j = \{\mathbf{f}_j, \mathbf{x}_j, t_j\}$ belong to the same object. In each observation, $\mathbf{f}$ is the blob color signature [24], $\mathbf{x}$ is the position of the centroid of the points in a blob, and t is the timestamp of the observation. The conditional probability distribution of k_i^j given two observations O_i, O_j is

$$p(k_i^j | O_i, O_j) \propto p(\mathbf{f}_i, \mathbf{f}_j | k_i^j) p(\{\mathbf{x}_i, t_i\}, \{\mathbf{x}_j, t_j\} | k_i^j) p(k_i^j), \quad (2)$$

assuming independence of $\mathbf{f}$ from $(\mathbf{x}, t)$. Lacking further information, we may assume that $p(k_i^j)$ is uniformly distributed. We define

$$p(\mathbf{f}_i, \mathbf{f}_j | k_i^j) \propto e^{-\alpha d(\mathbf{f}_i, \mathbf{f}_j)}, \quad (3)$$

where $d(\mathbf{f}_i, \mathbf{f}_j)$ is the Earth Movers Distance (EMD) [24]. We also define

$$p(\{\mathbf{x}_i, t_i\}, \{\mathbf{x}_j, t_j\} | k_i^j) \propto e^{-\beta \|\mathbf{x}_i - \mathbf{x}_j\| - \gamma \|t_j - t_i - \Delta t\|} \quad (4)$$

where Δt is the inter-frame time. We estimate the constants α , β and γ in these expressions through data fitting to training samples.

To compute tracklets, we build a directed graph $G = (V, E, P)$ whose node set V is the set of observations O_i , edge (i, j) in E connects observations O_i

and O_j such that $t_i < t_j$, and the weights in P are the probabilities $\pi_{ij} = p(k_i^j = 1 | O_i, O_j)$, evaluated as explained above. Edges with zero weight are omitted from E . In G , we define *tracklets* as strongly connected paths, constructed greedily as follows: Let i_0 be the oldest observation in V . For each i , let

$$j(i) = \arg \max_{j \in V, (i,j) \in E} \pi_{ij}. \quad (5)$$

A tracklet is the resulting path $i_0, i_1 = j(i_0), i_2 = j(i_1), \dots$. The path ends when $j(i_n)$ is undefined. We then remove the elements of the tracklet from the graph, and repeat.

5 Experimental Results

Our experiments evaluate the impact of depth information, used as described earlier, on the performance of the HOG person detector. To this end, we captured and processed three sequences s_1, s_2 and s_3 with a Microsoft Kinect sensor [8]. Each sequence contains roughly 2,000 color/depth image pairs at VGA resolution (640×480). Sample frames are shown in Fig. 1. We divided the workspace in cells with grid step $\Delta x = \Delta y = \Delta z = 0.1\text{m}$. Using the MATLAB implementation of the Nelder-Mead optimization algorithm [18], we estimated the pitch (β) and roll (α) angles. We used floor points in s_1 and s_3 , and ceiling points in s_2 . The estimated angles for pitch and roll are -3.4 and -1.1 degrees for s_1 , 0.9 and 4.1 for s_2 , and -0.9 and 3.4 for s_3 . Only points between 0.1m and 2.5m above floor level are considered for processing.

To construct a model of the background, we chose 20 frames from s_1 , 80 from s_2 , and 160 from s_3 , consecutive and without people. To detect people, we used the OpenCV [31] implementation of the HOG [1] algorithm. From the HOG, we retained all the detections with a strictly positive SVM score. Fig. 1 shows some intermediate results for three scenarios. Part (a) illustrates the detection of blobs in the depth images. Part (b) illustrates the performance of the HOG detector. Scenes without people, like in part (c) and (e), were used to build the background model for the depth maps. The combined use of space-time and color constraints to detect people is illustrated in (d) and (f). Tracklets are in red, and the HOG windows with top scores are shown in (d) and the foreground blobs are shown in (f).

The multiscale search of the OpenCV implementation of the HOG detector examines 34,981 candidates per image. Out of these, the HOG algorithm eliminates many false positives, depending on the threshold used on the SVM score. Adding depth information by our method improves detection performance significantly. In Fig. 2, we plot two curves for false positives (**fp**) and true positives (**tp**) for different HOG score thresholds. These curves relate results without and with the use of depth information. When the HOG score threshold is zero, our method reduces the number of **fp** from 2,594 to 76, while the number of **tp** increases from 245 to 615. When the threshold is set to 3.15, the highest value that results in some HOG detections, the number of **fp** goes from 2 to 7 and

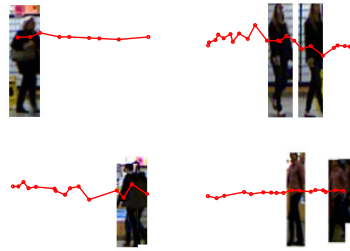
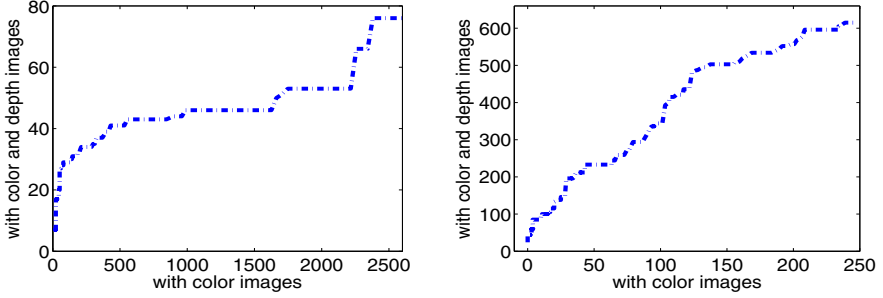
(a) Depth blobs from s_1 (b) Color HOG detection from s_1 (c) Empty scene from s_2 (d) Tracklets from s_2 with top-score HOG detections(e) Empty scene from s_3 (f) Tracklets from s_3 with foreground blobs

Fig. 1. People Detection using Color and Depth Images. (a) Bounding boxes of foreground connected components found in a depth frame from s_1 . (b) HOG detections in a color frame from s_1 . In this particular frame, the HOG finds two people, misses two, and produces a false positive to the right of the plant. (c) A frame from s_2 , with no people. (d) Tracklets (red polygonal lines) from s_2 with superimposed top-scoring HOG detection results. (e) A frame from s_3 , with no people. (f) Tracklets (red polygonal lines) from s_3 with superimposed foreground blobs.

that of $\mathbf{tp}$ goes from 0 to 16. Overall, with our approach, the number of false positives is greatly reduced, and the number of true positives is simultaneously increased.



(a) False positives with and without depth information (b) True positives with and without depth information

Fig. 2. Performance Evaluation. In these graphs, we vary the acceptance threshold τ for the HOG detector from 0 (all pass) to 3.15 (most strict). In each diagram, the horizontal axis is the number of HOG detections on the color images, and the vertical axis is the number of detections with our method. (a) Number of false positive detections (**fp**). For $\tau = 0$, HOG alone obtains 2,594 **fp** and our method yields 76. For $\tau = 3.15$, **fp** is 2 for HOG alone and 7 with our approach. (b) Number of true positive detections (**tp**). For $\tau = 0$, HOG alone finds 245 **tp** and our method finds 615. When $\tau = 3.15$, **tp** is 0 for HOG alone and 16 for our approach. A standard ROC curve [26] would be essentially meaningless, because the multiscale image scan examines 34,981 windows per image, vastly more than the expected number of targets.

6 Conclusion

In this paper, we presented a strategy to combine depth-based tracking and appearance-based HOG people detection. This strategy greatly improves *both* precision and recall. Our object detector is computationally efficient and accurate. Overall, our strategy seems to give excellent results in indoor environments. In the future, we plan to explore less greedy methods for the construction of tracklets, more nuanced models of image similarity and space-time closeness, and more detailed models of sensor uncertainty. We also plan to extend our method to part-based person detection methods.

Our study suggests that the emergence of new, inexpensive depth sensors presents new opportunities for surveillance, activity analysis and people tracking. Nonetheless, these sensors are unlikely to supplant regular cameras altogether. This is because current depth sensors typically project infrared light, either temporally modulated or spatially structured, on the scene. Black or dark surfaces do not reflect well, and sometimes not at all, making background subtraction harder, and creating difficulties with people with dark hair or clothing. In addition, depth sensors are inherently limited to shorter distances because eye safety demands low illumination power levels. However, when the right conditions are met, range sensors provide an invaluable resource of information that can enhance the performance of demanding perceptual tasks.

References

1. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: IEEE Computer Vision and Pattern Recognition, vol. 1, pp. 886–893 (2005)
2. Dalal, N., Triggs, B., Schmid, C.: Human Detection using Oriented Histograms of Flow and Appearance. In: European Conference on Computer Vision, pp. 428–441 (2006)
3. Dalal, N.: INRIA Person Database (September 2010), <http://pascal.inrialpes.fr/soft/olt/>
4. Elfes, A.: Using Occupancy Grids for Mobile Robot Perception and Navigation. *Computer* 22(6), 46–57 (2002)
5. Gavrila, D.: Pedestrian Detection from a Moving Vehicle. In: European Conference on Computer Vision, pp. 37–49 (2000)
6. Gavrila, D., Giebel, J., Munder, S.: Vision-based Pedestrian Detection: The Protector System. In: Intelligent Vehicles Symposium, pp. 13–18 (2004)
7. Gavrila, D.: The Visual Analysis of Human Movement: A Survey. *Computer Vision and Image Understanding* 73(1), 82–98 (1999)
8. Giles, J.: Inside the Race to Hack the Kinect. *The New Scientist* 208 (2789) (2010)
9. Gordon, G., Darrell, T., Harville, M., Woodfill, J.: Background Estimation and Removal based on Range and Color. In: IEEE Computer Vision and Pattern Recognition, p. 2 (1999)
10. Javed, O., Shafique, K., Rasheed, Z., Shah, M.: Modeling Inter-Camera Space-Time and Appearance Relationships for Tracking Across Non-Overlapping Views. *Computer Vision and Image Understanding* 109(2), 146–162 (2008)
11. Johansson, G.: Visual Perception of Biological Motion and a Model for its Analysis. *Perceiving Events and Objects* 3 (1973)
12. Kelly, M.: Visual Identification of People by Computer. Ph.D. thesis, Stanford University (1971)
13. Lowe, D.: Object Recognition from Local Scale-invariant Features. In: IEEE International Conference on Computer Vision, p. 1150 (1999)
14. Micilotta, A., Ong, E., Bowden, R.: Detection and Tracking of Humans by Probabilistic Body Part Assembly. In: British Machine Vision Conference, vol. 1, pp. 429–438 (2005)
15. Mikolajczyk, K., Schmid, C., Zisserman, A.: Human Detection based on a Probabilistic Assembly of Robust Part Detectors. In: European Conference on Computer Vision, pp. 69–82 (2004)
16. Mohan, A., Papageorgiou, C., Poggio, T.: Example-based Object Detection in Images by Components. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 23(4), 349 (2001)
17. Muñoz, R., Aguirre, E., García, M.: People Detection and Tracking using Stereo Vision and Color. *Image and Vision Computing* 25(6), 995–1007 (2007)
18. Nelder, J., Mead, R.: A Simplex Method for Function Minimization. *The Computer Journal* 7(4), 308 (1965)
19. Papageorgiou, C., Poggio, T.: A Trainable System for Object Detection. *International Journal of Computer Vision* 38(1), 15–33 (2000)
20. Phillips, P.: Human Identification Technical Challenges. In: IEEE International Conference on Image Processing (2002)
21. Ramanan, D., Forsyth, D., Zisserman, A.: Strike a Pose: Tracking People by Finding Stylized Poses. In: IEEE Computer Vision and Pattern Recognition, pp. 271–278 (2005)

22. Roberts, T., McKenna, S., Ricketts, I.: Human Pose Estimation using Learnt Probabilistic Region Similarities and Partial Configurations. In: European Conference on Computer Vision, pp. 291–303 (2004)
23. Ronfard, R., Schmid, C., Triggs, B.: Learning to Parse Pictures of People. In: European Conference on Computer Vision, pp. 700–714 (2006)
24. Rubner, Y., Tomasi, C., Guibas, L.: The earth mover’s distance as a metric for image retrieval. *International Journal of Computer Vision* 40(2), 99–121 (2000)
25. Schwartz, W., Kembhavi, A., Harwood, D., Davis, L.: Human Detection using Partial Least Squares Analysis. In: IEEE International Conference on Computer Vision, pp. 24–31 (2010)
26. Swets, J., Dawes, R., Monahan, J.: Better Decisions through Science. *Scientific American*, 83 (2000)
27. Theodoridis, S., Koutroumbas, K.: Pattern Recognition. Elsevier, Amsterdam (2009)
28. Viola, P., Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: IEEE Computer Vision and Pattern Recognition, vol. 1 (2001)
29. Viola, P., Jones, M., Snow, D.: Detecting Pedestrians using Patterns of Motion and Appearance. *International Journal of Computer Vision* 63(2), 153–161 (2005)
30. Vrabel, A., Bellon, O., Silva, L.: Planar Background Elimination in Range Images: A Practical Approach. In: IEEE International Conference on Image Processing, pp. 3197–3200 (2009)
31. Willow Garage: OpenCV (September 2010), <http://opencv.willowgarage.com>
32. Xu, F., Fujimura, K.: Human Detection using Depth and Gray Images. In: IEEE Advanced Video and Signal Based Surveillance. pp. 115–121. IEEE, New York (2003)
33. Zhao, L., Davis, L.: Closely coupled object detection and segmentation. In: IEEE International Conference on Computer Vision, pp. 454–461 (2005)
34. Zhao, L., Thorpe, C.: Stereo and Neural Network-based Pedestrian Detection. *IEEE Transactions on Intelligent Transportation Systems* 1(3), 148–154 (2000)

Measuring Rectangularity Using GR-Signature

Jihen Hentati¹, Mohamed Naouai^{1,2}, Atef Hamouda¹, and Christiane Weber²

¹ Faculty of Science of Tunis, University campus el Manar DSI

2092 Tunis Belvédère-Tunisia

Research unit URPAH

naouai@polytech.unice.fr, atef_hammouda@yahoo.fr,
hentati.jihen@gmail.com

² Laboratory Image and Ville UMR7011-CNRS-University Strasbourg

3rue de l'Argonne F-67000 Strasbourg

naouai@polytech.unice.fr,
christiane.weber@live-cnrs.unistra.fr

Abstract. Object recognition often operates by making decisions based on the values of several shape properties measured from an image of the object. In this paper, we propose a new exploitation of the Radon Transform using the gradient measurement to generate a new signature (GR-signature) which provides global information of a binary shape regardless its form. We also develop a new method for measuring the rectangularity based on GR-signature. This original approach looks very promising and has several useful properties that keep fundamental geometrical transformations like scale, rotation and translation.

Keywords: Rectangularity, Shape descriptor, Radon Transform, gradient measurement.

1 Introduction

Object recognition is one of the main issues described in most of computer vision applications. This object identification is made in the forms analysis phase which generally occurs after a step of image segmentation [14]. The shape analysis is used in several application areas such as medicine to detect anomalies, security to identify individuals, computer-aided design and computer-aided manufacturing process to compare design parts or mechanical objects design etc. Discrimination of objects is based on their appearance: texture, color and shape. The shape is obviously a powerful tool to describe and differentiate objects since it is a discriminating characteristic of the object. The form according to the mathematician and statistician David George Kendall is defined as [7]: “The shape is the set of geometric information that remains when location, scale and rotational effects are filtered from an object”.

Once the shapes are extracted from the image, they must be simplified before a comparison can be made. The simplified representation of forms is often called the shape descriptor or signature. This is an abstraction of a structured model that captures most of the important information of the form. These simplified

representations are easier to handle, store and compare than the forms directly. The shape may not be entirely reconstructable from the descriptors, but the descriptors for different shapes should be different enough that the shapes can be discriminated [2]. So instead of directly comparing two models, both models are compared by comparing their shape descriptors.

Although some researches has been done in terms of circularity, ellipticity and rectangularity. However many textbooks and surveys do not consider this last as a measure of shape [1, 2]. Moreover, rectangularity can be an advantageous characteristic to extract useful tasks such as filtering of images to find parts potential road in a satellite image.

Beside there are many attempts to measure the rectangularity. The standard method, the Minimum Bounding Rectangle method (MBR), responds unequally to protrusions and indentations, and is sensitive to noise (especially protrusions). Also the research of P.Rosin in [8] develops three methods:

- The agreement method (R_A): breaks down for compact regions and is prone to errors due to inaccuracies in the perimeter estimation. The errors depend on both the region's orientation and resolution.
- The moment-based method (R_M): can respond to other shapes such as rectangles if they have a similar ratio of moments. For compact shapes (e.g. the near square on the bottom row), the orientation estimation is sensitive to noise, which can lead to incorrect rectangularity estimation.
- The discrepancy method (R_D): uses moments to estimate the rectangle fit and is similarly prone to poor orientation estimation for compact shapes.

In his research, Rosin proves that the bounding rectangle (MBR) and discrepancy method (R_D) are the best. Moreover, in [12] the Radon Transform (RT) is used to calculate the R-signature (i.e. the square of the RT) which just characterizes very well the shape of the filled and not emptied object (i.e. object contour only). In this approach, the R-signature of an object is compared to a theoretic R-signature which represents a perfect rectangle and calculates the similarity between them.

In this study a simple but effective method is proposed and it utilizes the RT and the gradient to make a signature (we called the GR-signature). With the help of this signature and a metric of measuring the rectangularity which we propose, we calculate the percentage of the rectangularity of a given object.

This paper is outlined as follows. After we recall the definition and the properties of the RT and the gradient in sections 2 and 3, we describe our GR-signature in section 4. Our new metric is described in section 5 and evaluated on synthetic data in section 6 to determine how well it surmounts the imperfections of previous approaches by making comparison with them result. Finally, we summarize our research and conclude the paper in section 7.

2 Radon Transform

To be useful, a shape recognition framework should allow explicit invariance under the operations of translation, rotation, and scaling. For this reasons, we have decided

to employ the Radon transform. By definition, the RT [9] of an image is determined by a set of projections of the image along lines taken at different angles. For discrete binary image data, each non-zero image point is projected into a Radon matrix. Let $f(x, y)$ be an image. Its Radon transform is defined by [4]:

$$R(\rho, \theta) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} f(x, y) \delta(\rho - x \cos \theta - y \sin \theta) dx dy. \quad (1)$$

Where $\delta(\cdot)$ is the Dirac function, $\theta \in [0, \pi]$ and $\rho \in [-\infty, \infty]$.

To represent an image, the Radon Transform takes multiple, parallel-beam projections of the image from different angles by rotating the source around the center of the image. Fig.1 shows a single projection at a specified rotation angle. For example, the line integral of $f(x, y)$ in the vertical direction is the projection of $f(x, y)$ onto the x -axis; the line integral in the horizontal direction is the projection of $f(x, y)$ onto the y -axis [13]. The RT is robust to noise, provided with fast algorithms, and it projects a two-dimensional function into one-dimensional function.

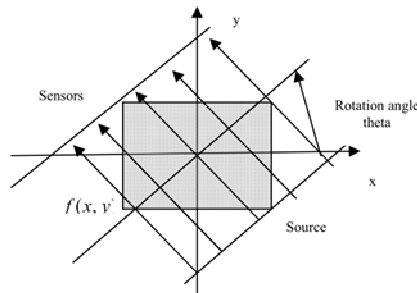


Fig. 1. Parallel-beam projection at rotation angle θ

The Radon transform has several useful properties. Some of them are relevant for shape representation [13]:

- Periodicity $T_{Rf}(\rho, \theta) = T_{Rf}(\rho, \theta + 2k\pi)$, for any integer k . The period is 2π .
- Symmetry $T_{Rf}(\rho, \theta) = T_{Rf}(-\rho, \theta \pm \pi)$.
- Translation of a vector $\vec{u} = (x_0, y_0)$: $T_{Rf}(\rho - x_0 \cos \theta - y_0 \sin \theta, \theta)$ A translation of f results in the shift of its transform in the variable ρ by a distance equal to the projection of the translation vector on the line $\rho = x_0 \cos \theta + y_0 \sin \theta$.
- Rotation by θ_0 : $T_{Rf}(\rho, \theta + \theta_0)$. A rotation of the image by an angle θ_0 implies a shift of the Radon transform in the variable θ .
- Scaling of α : $\frac{1}{\alpha} T_{Rf}(\alpha \rho, \theta)$. A scaling of f results in a scaling of both the ρ coordinates and the amplitude of the transform.

3 Gradient

To exploit the RT and like it contains several peaks (loci of concentration), we choose to use the gradient to locate those peaks. In physics, the gradient is a vector quantity that indicates how a physical quantity varies in space. In image processing, the gradient profile prior is a parametric distribution describing the shape and the sharpness of the gradient profiles in natural image [6]. The definition of the gradient vector $\vec{\nabla}f$ of the fonction $f(x, y)$ is:

$$\vec{\nabla}f = \begin{pmatrix} \frac{\partial f(x,y)}{\partial x} \\ \frac{\partial f(x,y)}{\partial y} \end{pmatrix} \quad (2)$$

The direction of $\vec{\nabla}f$ is the orientation in which the directional derivative has the largest value and $|\vec{\nabla}f|$ is the value of that directional derivative [10]. The gradient profile is a 1-D profile along the gradient direction of the zero-crossing pixel in the image [6]. We use it to find the modes of density in a feature space. The modes are located among the zeros of gradient ($\vec{\nabla}f(x, y) = 0$).

4 GR-Signature

In our research a new exploitation of the RT is proposed. Our method differs from previous 2D RT applications [10, 11]. In these approaches, the encoded information is contour-based allowing only the detection of specific primitives like straight line.

The context of our application is different from previous works. We provide global information of binary shape, whatever its form is, by generating a new signature (GR-signature). In fact, the operating principle of the RT is the summation of the intensity of pixels along the same line for each projection. To obtain an outcome that reflects only the shape, the object must have a unique color. Otherwise the result of RT reflects the brightness of the object in addition to its shape. For that, we will use binary images. Moreover, we do not need any pretreatment like computing the centroid of shapes under consideration when using Fourier descriptors [11].

In the discrete case, fast and accurate algorithms [5] haven been proposed to transform the continuous plane of Radon into an accumulator matrix $R: N_\rho * N_\theta$ cells described by the sinogram in Fig. 2.b. From this 2D accumulator we generate a discrete 1D GR-signature by calculating the gradient.

$$\frac{\partial R}{\partial \theta} \quad (3)$$

The modulus of the gradient vector represents the surface slope Radon point calculation. The local presence of a high modulus indicates a high variation of the coefficients around this point. Where we fix one line of a matrix, the gradient will locate the high coefficients variation in this line. We want to catch the variation in

each θ projection, for that the θ must be the first dimension of the matrix. But in fact θ is the second dimension and the first one is ρ . So we reflect R over its main diagonal (which runs top-left to bottom-right) to obtain R' (the transpose of a matrix R). After the transposing we apply the formula (3) on R' . We obtain the result shown in Fig. 2.c. The graph is very dense and contains an enormous amount of information. Hence, we choose to take only the external shell (i.e. contour) of the gradient result and this is the GR-signature (Fig. 2.d).

In addition, the GR-signature proves to be an excellent measure of shape and it gives very good results with full or empty symbols. This is caused by the fact that the GR-signature is based on the corners of the shape.

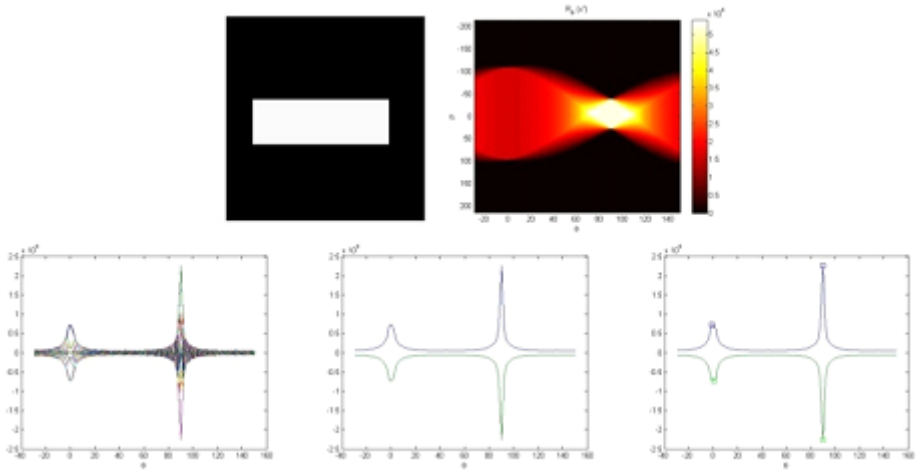


Fig. 2. Definition of the GR-signature: a.shape, b.Radon space, c.gradient result calculation, d.GR-signature and e.peaks selection

5 Rectangularity Measure (R_{GR})

In this phase, we will use our GR-signature to define the percentage of rectangularity of every given shape. For this objective and before we come to our metric of rectangularity verification, we study the GR-signature of an ideal rectangle. We find that the two sides (positive and negative) of the GR-signature are symmetric. Also the sum of the absolute values of each opposite peaks is equal to one of the rectangle dimension (i.e. the sum of the two high peaks is equal to the length of rectangle and the sum of the two low peaks is equal to the width). Furthermore, the difference in θ scale between the high and the low peaks of the GR-signature is 90° in each side, which represent the angle between the two perpendicular bisectors of the rectangle. We recall that the two rectangle bisectors are the lines perpendicular to the length and width segments in there middle as shown in Fig. 3.

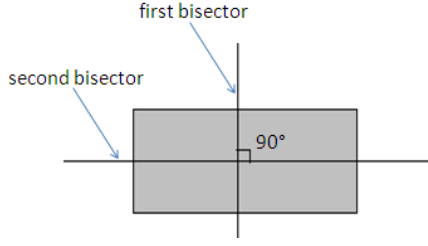


Fig. 3. The two bisectors of a rectangle

After this pretreatment phase, we create our metric of rectangularity which is a combination of two different measures: Angle measurement and Amplitude measurement. To proceed to the two measurements, and since they depend of the number of shape's corners, a phase of detecting peaks in the GR-signature is needed. We treat the signature side by side identically. First, we extract its extrema (maxima in the positive side and minima in the negative) and sort them according to its amplitude in an ascend order. Depending on the number of the rectangular shape's corners, we choose four extrema. We locate the highest peak in the GR-signature. The second extremum is located in the same side taking into account that it is far from the first by 90° with margin of tolerance of $\pm 5^\circ$. The third and fourth peaks are located in the same way symmetrically.

The Angle measurement is described by formula (4):

$$\text{Angle\_measurement} = \frac{|90 - (\theta_{\text{low}} + \theta_{\text{high}})|}{90} \quad (4)$$

Where θ_{low} is the difference between the two low peaks and θ_{high} is the difference between the two high peaks. The sum of these two differences represents the angle rate error. The expression $(90 - |\theta_{\text{max}} + \theta_{\text{min}}|)$ represents the angle between the two bisectors of the rectangle. To unify it, we devise it by 90. A value of one is produced for an exact rectangle, while decreasing values correspond to less rectangular figures.

The Amplitude measurement is calculated with the help of the amplitudes of each selected peaks. At first we normalize these amplitudes to be in the range of [0 1] using formula (5):

$$A_i = \frac{A_i - A_{\min}}{A_{\max} - A_{\min}} \quad (5)$$

Where $A_{\max}$ is the greatest amplitude and $A_{\min}$ is the smallest one. A_i ($i \in [1..4]$) is the amplitude of each one of the four peaks. After that we sort these amplitudes in an ascend order and then we calculate the difference of the first two devised by the difference of the last two. This measurement is described by formula (6) which peaks at one for perfect rectangles:

$$\text{Amplitude\_measurement} = 1 - \left| \frac{A_2 - A_1}{A_4 - A_3} \right| \quad (6)$$

Where A_i ($i \in [1..4]$) is the amplitudes of the four peaks.

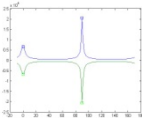
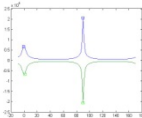
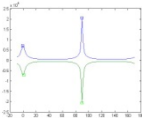
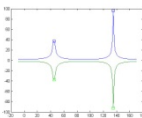
After the calculation of the two measurements (Angle measurement and Amplitude measurement) and since percentage of rectangularity depends equally on both of them, we define the rectangularity measure (R_{GR}) as the average of the two measurements which peaks at one for perfect rectangles:

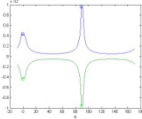
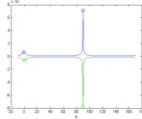
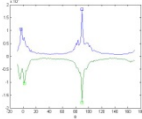
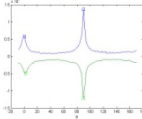
$$R_{GR} = \frac{\text{Angle\_measurement} + \text{Amplitude\_measurement}}{2} \tag{7}$$

6 Evaluation

We evaluate the R_{GR} measure by applying it to some synthetic shapes. This enables us to track the rectangularity values, as we continuously change the shapes, to prove that the GR-signature conserves the several useful properties of Radon and well behaves with noised figures. These evaluations are illustrated in Table 1.

Table 1. Properties of the GR-signature

Property	Full shape	Empty shape	Translation	Rotation
Shape				
GR-signature				
R_{GR}	1.0000	1.0000	1.0000	0.9868

Property	Scaling	Gaussian noise	Protrusions and Indentations	Boundary noise
Shape				
GR-signature				
R_{GR}	0.9994	0.9773	0.9926	0.9775

We conclude that full or empty shape don't affect the rectangularity measurement. This is of crucial importance in object recognition, because each object must have a unique representation either is full or not. Our descriptor is invariant under geometrical transformations (translation, rotation and scaling). When we applied our rectangularity measurement on geometric transformed shapes we obtain very good results (the means of R_{GR} over 0.98) which provide the stability of our metric. This R_{GR} measurement is robust to the noise also. We applied Gaussian noise, Boundary noise and protrusions and

indentations on a shape but the measurement still have good values of measurement (the means of R_{GR} over 0.97). This refers to the ability of the representation to express the basic features of a shape and to abstract from detail. So the R_{GR} appears a good rectangularity measure and the GR-signature looks a crucial descriptor.

We evaluate our descriptor by applying it to an images' database and compare its figures classification result to classification [5] proposed by Paul Rosin for one hand and that based on the R-signature as looks in figures Fig. 4, Fig. 5 and Fig. 6.



Fig. 4. The classification of the images database using the rectangularity measurement proposed by Paul Rosin



Fig. 5. The classification of the images database using the rectangularity measurement based on the R-signature

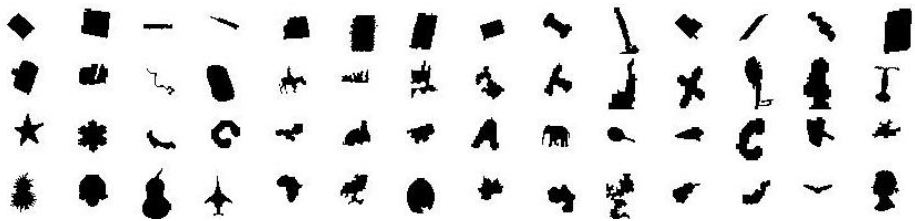


Fig. 6. The classification of the images database using the rectangularity measurement based on our GR-signature

The analysis of the GR-signature arrangement of the images' database reveals that on viewpoint discrimination of the rectangular shape; our descriptor looks well since all the 18th first figures have a rectangular form. A small comparison between the results of classification according to Rosin, R-signature and GR-signature is illustrated in Table 2.

Table 2. Comparison between the results of classification presented in fig. 4-6

Images' database	Rosin rank	R-signature rank	GR-signature rank
Face 1	9	10	44
Oval shape	12	9	49
Face 2	13	21	56
Tree	14	18	27
Guitar	16	26	45
Snow crystal	18	19	30
Maple leaf	21	33	50
Africa map	26	20	47
Sword	30	23	4
Noised rectangle 1	25	16	9
Noised rectangle 2	55	53	10
Noised rectangle 3	23	17	12
Noised rectangle 4	24	27	13
Noised rectangle 5	40	48	21
Noised rectangle 6	35	36	22
Noised rectangle 7	41	37	24

Table 2 reflects that our descriptor is able to discriminate the rectangular shapes from others forms since it improves the rank of rectangular shapes and disapproves that of other forms compared to other classifications.

7 Conclusions

Our paper shows that the GR-signature can be of great interest to differentiate between graphical symbols and also in the measure of rectangularity. The computation of such a feature is fast (low complexity). Moreover, it overcomes the problems of other approaches. A weakness of using the MBR is that it is very sensitive to protrusions from the region [8]. But using our metric, protrusions and indentations have no considered effect on the rectangularity measurement. The rectangularity value of the rectangular shape with protrusions and indentations illustrate in Table 1 using our metric is 0.9926. It is a very good value despite the protrusions and indentations. Its signature clearly shows the rectangular form and picks are well chosen (Table 1).

What makes this metric better in comparison with Rosin methods and the standard method (MBR) is the fact that the proprieties of GR-signature inherited from the Radon transform overcame the problems of geometrical transformations. As regards the mismatch problem that appears in R_M , it is solved by the similarity pretreatment preceding our rectangularity measurements. And what differentiate our method from

the R-signature is that we found a better exploitation of the Radon space which allowed revealing useful properties (Angle, Amplitude measurements) and not only matching two signatures. This gives more accurate result in the rectangularity measurements.

Of course, the results presented in this paper must still be considered as preliminary. We need to process much larger databases of graphical symbols to assess the discriminating power and the robustness of the method.

References

1. Ballard, D.H., Brown, C.M.: Computer Vision. Prentice Hall, Englewood Cliffs (1982)
2. Morse, B.S.: Lecture 9: Shape Description (Regions). Brigham Young University (1998–2000)
3. Campbell, L., MacKinlay: The Econometrics of Financial Markets. Princeton University Press, NJ (1996)
4. Deans, S.R.: Applications of the Radon Transform. Wiley Inter-science Publications, New York (1983)
5. Girardeau-Montaut Daniel, application de la transformé de Radon à l'identification de symboles graphiques, DEA de Institut National Polytechnique de Lorraine (2002).
6. Sun, J., Sun, J., Xu, Z., Shum, H.-Y.: Image Super-Resolution using Gradient Profile Prior. In: IEEE Conference on Computer Vision and Pattern Recognition, Xi'an Jiaotong University, Microsoft Research Asia Beijing, P. R. China (2008)
7. Kendall, D.G.: Shape Manifolds, Procrustean Metrics, and Complex Projective Spaces. Bulletin of the London Mathematical Society (1984)
8. Rosin, P.: Measuring rectangularity. Machine Vision and Applications 11, 191–196 (1999)
9. Radon, J.: Ufiber die Bestimmung von Funktionen durch ihre Integral-werte langs gewisser Mannigfaltigkeiten, Berichte Sa-chsische Akademie der Wissenschaften, Leipzig. Math.-Phys. Kl. 69, 262–267 (1917)
10. Schey, H.M.: Div, Grad, Curl, and All That: An Informal Text on Vector Calculus, 3rd edn. W. W. Norton, New York (1997)
11. Tabbone, Ramos Terrades, O., Barrat, S.: Histogram of Radon Transform. In: A useful descriptor for shape retrieval of 19th International Conference on Pattern Recognition - ICPR, S University of Nancy, LORIA (2008)
12. Tabbone, S., Wendling, L., Girardeau-Montaut, D.: Mesures de rectangularité et d'ellipticité à partir de la transformée de Radon. C.I.F.E.D. Hammamet, Tunisie (2002)
13. Jia-wen, W., Yang-jun, L.: MATLAB 7.0 Image Processing, pp. 190–191. National Defence Industry Publishing, Beijing (2006)
14. Naouai, M., Hamouda, A., Weber, C., Melki, N.: linear structure recognition based on image vectorization. In: International Conference on Imaging Theory and Applications, Algarve, Portugal, March 5-7 (2011)

Multi-modal 3D Image Registration Based on Estimation of Non-rigid Deformation

Roberto Rosas-Romero, Oleg Starostenko, Jorge Rodríguez-Asomoza,
and Vicente Alarcon-Aquino

Department of Computing, Electronics, and Mechatronics,
Universidad de las Américas Puebla, Cholula, 72820, México
{roberto.rosas, oleg.starostenko, jorge.rodriguez,
vicente.alarcon}@udlap.mx

Abstract. This paper presents a novel approach for registration of 3D images based on optimal free-form rigid transformation. A proposal consists in semiautomatic image segmentation reconstructing 3D object surfaces in medical images. The proposed extraction technique employs gradients in sequences of 3D medical images to attract a deformable surface model by using imaging planes that correspond to multiple locations of feature points in space, instead of detecting contours on each imaging plane in isolation. Feature points are used as a reference before and after a deformation. An issue concerning this relation is difficult and deserves attention to develop a methodology to find the optimal number of points that gives the best estimates and does not sacrifice computational speed. After generating a representation for each of two 3D objects, we find the best similarity transformation that represents the object deformation between them. The proposed approach has been tested using different imaging modalities by morphing data from Histology sections to match MRI of carotid artery.

Keywords: 3D image matching, non-rigid deformation estimation, wavelet.

1 Introduction

Estimation of non-rigid deformation associated with objects in medical images can be used as assisting clinical tool to identify abnormal organ behavior. Medical image registration can be performed first by finding the deformation between a pair of images and then correcting the changes associated with such deformations to perform multi-modal image registration between medical data sets to integrate information from different modalities (ultrasound, X-ray, Magnetic Resonance Imaging (MRI), Histology, etc.), as well as register images taken at different times (temporal registration). So that, changes associated with disease evolution can be inferred. Since the resolution and distortion are different in every imaging modality and the tissue often changes in size and shape with time, estimation of deformation is a growing field.

Besides its medical application, image registration also has a variety of other applications, such as aerial image analysis, stereo vision, automated cartography, motion analysis, recovery of 3-D characteristics of a scene, and performing morphing of 3-D data sets for computer animation of visual effects [1-4]. One problem in the most efforts to describe organ deformations is that they require extensive human interaction. For instance, visual inspection of shape of the heart within a cardiac cycle is widely used to detect abnormalities [5]. In other cases, deformation is estimated by manually tracking the movement of predefined landmark points of organ borders. Techniques that create markers on images have the drawbacks of being invasive and the fact that the more tags there are, the poorer the background image signal to noise is. Thus, we propose a non-invasive technique that reduces the amount of human intervention.

Usually, image registration is accomplished first by systematically reconstructing the surfaces and feature point sets of two 3D objects extracted from two sets of images. Therefore, for each set there is a representation of the object that consists in its surface and a set of feature points. Feature points to be extracted are those that can be used as a reference before and after a deformation. After generating a representation for each of the two 3D objects, we find the best similarity transformation that represents the object deformation between them. Well-known methods based on the registration of 3D curves are efficient enough but these methods are not useful when registration of 3D surfaces is required [7], [8]. Other efforts try to model non-rigid a deformation using successive transformation such as twisting, banding, tapering and the inconvenience of these approaches is that a non-rigid deformation might require a description, which is not provided by a combination of simple deformation models [9]. Registration of 3-D medical images under non-rigid deformation using physical properties of the objects has been widely studied, however, one problem of these techniques is that physical properties must be obtained for each specific application and they might be difficult to obtain or not available [10], [11].

2 3D Object Surface Extraction Based on Active Contour Models

Extraction of the surface of an object from an image set consists in reconstructing its shape from points collected from its physical surface. There is a set of 3D images that describes an object that is used as a reference. A set of imaging planes is obtained by scanning the object in parallel slices and the intersection of each imaging plane with the object gives a contour. Tracing of contour points on parallel imaging planes and joining them generates a 3D surface. We use active contour models to extract contour points from images [12].

Consider the problem of detecting the borders of a 3D reference object. If there are m planes per 3D object, and n points per plane; then there are $N = mn$ contour points to be searched. By using the Original Snake Active Contour Model, a single imaging plane is used to detect n contour points on that specific plane, and this process is repeated for each of m imaging planes in the object. Instead of attempting to perform contour detection for each imaging plane in isolation, we directly approach it as a 3D problem; so that the mn contour points corresponding to the object surface are

detected at once from multiple imaging planes. A snake is a deformable curve, which approaches contours on images by optimizing the placement of the snake points that form the curve. In the 3D model, each snake point

$$\mathbf{v}(r,s) = [x(r,s), y(r,s), z(r,s)]^T \quad (1)$$

is a function of two parameters r (spatial index), s (imaging plane index); so that the 3D snake function to be optimized is defined as

$$f(\mathbf{v}) = \alpha_1 \|\mathbf{v}_r\| + \alpha_2 \|\mathbf{v}_s\| + \beta_1 \|\mathbf{v}_{rr}\| + \beta_2 \|\mathbf{v}_{ss}\| + \beta_4 \|\mathbf{v}_{rs}\| + E \quad (2)$$

where $\{\alpha_i\}$ are constants imposing a tension constraint, and $\{\beta_i\}$ are constants imposing a bending constraint and E is some sort of image gradient function. Since the snake points stick to their plane, they are functions of the x and y coordinates only. The minimization of the snake energy function yields two Euler equations which can be iteratively solved to find a local minimum of the snake energy function,

$$\begin{aligned} (\mathbf{D} + \gamma \mathbf{I}) \mathbf{x}_i &= \gamma \mathbf{x}_{i-1} - \mathbf{f}_x(\mathbf{x}_{i-1}, \mathbf{y}_{i-1}) \\ (\mathbf{D} + \gamma \mathbf{I}) \mathbf{y}_i &= \gamma \mathbf{y}_{i-1} - \mathbf{f}_y(\mathbf{x}_{i-1}, \mathbf{y}_{i-1}) \end{aligned} \quad (3)$$

where $\mathbf{D}$ is an $N \times N$ penta-diagonal matrix; the vectors $\mathbf{x}_i$ and $\mathbf{y}_i$ are the coordinates of the snake points at the i th iteration; the vectors $\mathbf{f}_x(\mathbf{x}_i, \mathbf{y}_i)$ and $\mathbf{f}_y(\mathbf{x}_i, \mathbf{y}_i)$ are the image forces of the partial derivatives of the external energy for the snake points; and γ is the step size control parameter. Since the matrix $\mathbf{D}$ is quite big, it is impractical to invert it directly, it is assumed that the change between $\mathbf{x}_i$ and $\mathbf{x}_{i-1}$ is small enough and LU decomposition is used for each plane.

3 Extraction of 3D Object Feature Points from Sets of Images

For feature extraction, similar regions of the object from two different sets of images are manually identified with the help of a radiologist. Then two feature points, from the sets of images that contain each of the identified regions, are extracted; so that feature extraction and feature correspondence establishment are accomplished simultaneously. Each selected feature point is an edge point whose edge response is maximum within the identified region. This edge-based approach for extraction of pairs of corresponding feature points from 3D regions applies the Wavelet Transform [13].

Let's consider two objects O and O' , which are related with a non-rigid deformation. To find a pair of correctly-matched feature-points from these two objects, we must first manually identify a 3D region from the set of imaging planes $\{I_1, I_2, \dots, I_m\}$ that describes O , and also a similar region from the set $\{I'_1, I'_2, \dots, I'_n\}$ corresponding to O' . A region of interest in O is defined as a 3D discrete function $f(x, y, z)$ that gives a gray level value. At $z = z_r$, $f(x, y, z_r)$ corresponds to a rectangular window within one particular imaging plane from the set of images $\{I_i\}$. Similarly, a region of interest in O' is a discrete function $g(x, y, z)$, where $g(x, y, z_s)$ corresponds to a rectangular window on one imaging plane from $\{I'_i\}$. Basically, these discrete functions are generated by extracting sections from images and stacking them. The condition used to identify regions of interest is that they must contain structures that are common to both objects O and O' . These structures correspond to sharp variations, which are generally located at boundaries, edges or corners. Once $f(x, y, z)$

and $g(x, y, z)$ are established; one feature point $P(x, y, z)$ is automatically extracted from $f(x, y, z)$ and a corresponding point Q is obtained from $g(x, y, z)$. The pair (P, Q) is called a correctly-matched feature-point pair. Wavelet transform for multiresolution local analysis is applied to extract these points.

Let $S(x, y, z)$ be a 3D smoothing function. We call a smoothing function any function $S(x, y, z)$ equal to a Gaussian. Three wavelets, $\Psi_1(x, y, z)$, $\Psi_2(x, y, z)$ and $\Psi_3(x, y, z)$ are the partial derivatives of the smoothing function $S(x, y, z)$ in the x , y and z directions, respectively, where:

$$\begin{aligned}\psi^1(x, y, z) &= \frac{\partial S(x, y, z)}{\partial x}, \quad \psi^2(x, y, z) = \frac{\partial S(x, y, z)}{\partial y}, \\ \psi^3(x, y, z) &= \frac{\partial S(x, y, z)}{\partial z}\end{aligned}\quad (4)$$

Dilating these functions by a scaling factor 2^j ,

$$\begin{aligned}\psi_j^1(x, y, z) &= \frac{1}{8^j} \psi^1\left(\frac{x}{2^j}, \frac{y}{2^j}, \frac{z}{2^j}\right), \quad \psi_j^2(x, y, z) = \frac{1}{8^j} \psi^2\left(\frac{x}{2^j}, \frac{y}{2^j}, \frac{z}{2^j}\right), \\ \psi_j^3(x, y, z) &= \frac{1}{8^j} \psi^3\left(\frac{x}{2^j}, \frac{y}{2^j}, \frac{z}{2^j}\right)\end{aligned}\quad (5)$$

At each scale, 2^j , the 3D wavelet transform of a function $f(x, y, z)$ can be decomposed into three directions as,

$$\begin{aligned}W_j^1 f(x, y, z) &= f(x, y, z) * \Psi_j^1(x, y, z); \\ W_j^2 f(x, y, z) &= f(x, y, z) * \Psi_j^2(x, y, z); \\ W_j^3 f(x, y, z) &= f(x, y, z) * \Psi_j^3(x, y, z).\end{aligned}\quad (6)$$

These three components are equivalent to the gradients of $f(x, y, z)$ smoothed by $S(x, y, z)$ at scale 2^j in the x , y and z directions. The local extreme of $W_i^1 f(x, y, z)$, $W_i^2 f(x, y, z)$ and $W_i^3 f(x, y, z)$ corresponds to the inflection points of surface $f(x, y, z) * S^j(x, y, z)$ along the x , y and z directions respectively. The direction of a gradient vector at a point (x_0, y_0, z_0) indicates the direction in the space (x, y, z) along which the directional derivative of $f(x, y, z)$ has the largest absolute value. Three-dimensional edges are defined as points (x_0, y_0, z_0) , where the modulus of gradient vector is maximum. Hence, 3D edge points can be located from the three components of $W_i^1 f(x, y, z)$, $W_i^2 f(x, y, z)$ and $W_i^3 f(x, y, z)$ of the wavelet transform. At a specific scale 2^j , the modulus of the gradient vector of $f(x, y, z)$ can be calculated as,

$$M_j f(x, y, z) = \sqrt{\left|W_i^1 f(x, y, z)\right|^2 + \left|W_i^2 f(x, y, z)\right|^2 + \left|W_i^3 f(x, y, z)\right|^2} \quad (7)$$

If the local maxima of $M_j f(x, y, z)$ are located, then all the 3D edge points of $f(x, y, z)$ at scale 2^j can be detected.

In general, noise is the main cause of false detection of edge points. In order to suppress the effect of noise, a criterion called edge correlation is introduced,

$$R_n(j, x, y, z) = \prod_{i=0}^{n-1} M_j + i f(x, y, z) \quad (8)$$

where n is a positive number indicating the number of scales involved in the multiplication, and j represent the initial scale for edge correlation. This process detects edge points, whose edge responses are the strongest within a local area. Two conditions are adopted to judge whether a point (x_o, y_o, z_o) is a feature point or not:

Condition 1. (x_o, y_o, z_o) must be a 3D edge point of the function $f(x, y, z)$. This means that (x_o, y_o, z_o) is a local maxima of $M_j f(x, y, z)$.

Condition 2. $M_j f(x_o, y_o, z_o) = \max \{M_j f(x, y, z) \mid (x, y, z) \in N_p\}$, where N_p is the region represented by $f(x, y, z)$.

4 Estimation of Non-rigid Deformation

Once a set of surface points S_1 and a set of feature points FP_1 in the object are established for each set of images, we need to find the transformation function T that matches the sets S_1 and FP_1 to the sets S_2 and FP_2 , $T(\{S_1, FP_1\}, \mathbf{p}) \approx \{S_2, FP_2\}$, where $\mathbf{p}$ are the transformation parameters that have to be found. The search of the deformation parameters is an optimization process that minimizes the *differences* between two sets of points (Levenverg-Marquardt Least Squares Minimization) [14]. During this optimization process, deformations are systematically applied to S_1 and FP_1 , by adjusting the set of parameters $\mathbf{p}$, until the corresponding sets of transformed points $T(\{S_1, FP_1\}, \mathbf{p})$ get as close as possible to the sets $\{S_2, FP_2\}$, until the distance $d()$ between both sets is minimized. Thus, estimating deformation can be referred to as the minimization of the cost function, $C(\mathbf{p}) = d(\{S_2, FP_2\}, T(\{S_1, FP_1\}))$.

For a similarity metric, the distance function establishes a parametric representation of the 3D object surface, using imaging planes at the first time frame, and enables measurement of 3D deformation during object movement within a time sequence. After a distance function is constructed for the initial 3D shape, the tracked surface based on the snake is fed to the distance function to perform deformation estimation. The model used for a 3D distance function is based on the 2D *chamfer distance model* [15]. Tri-linear interpolation is used to transform the resulting discrete distance map into a continuous distance function. In this interpolation process, the distance from any point $\mathbf{r}$ to S_2 is computed by finding the eight grid points which form the voxel that contains $\mathbf{r}$ and then interpolate the distance value $d(\mathbf{r}, S_2)$ contributed by the distance values d_{ijk} at the eight vertices of the voxel.

Assume there is a total of N points on the sample surface $\{\mathbf{q}_i \mid i = 1, 2, \dots, N\}$, and the corresponding points after transformation are $\{\mathbf{r}_i = T(\mathbf{q}_i, \mathbf{p}) \mid i = 1, 2, \dots, N\}$. It is defined that d_i is the distance between $\mathbf{r}_i$ and the nearest point on the reference surface. There is a total of N distance terms from each transformed point to the reference surface, so that the cost function can be formulated as

$$C(\mathbf{p}) = \sum_{i=1}^N d_i(\mathbf{p})^2 \quad (9)$$

Free Form Deformation FFD Models are used to describe non-rigid deformations. These models describe the deformation of an object in terms of the deformation of the space that contains the object. The set of parameters $\mathbf{p}$ for the transformation function $\mathbf{T}$ consists of a set of deformation vectors $\{\mathbf{v}_{ijk}\}$ located in a 3D discrete grid.

The mathematical model for the function that represents the free-form deformation of an arbitrary point corresponds to a mapping from the source coordinates (x, y, z) to the transformed coordinates (x', y', z') .

$$[x', y', z']^T = [x, y, z]^T + \mathbf{v}(x, y, z) \quad (10)$$

where the displacement function $\mathbf{v}(x, y, z)$ is a linear combination of interpolating functions.

$$\mathbf{v}(x, y, z) = \sum_i \sum_j \sum_k \psi_{ijk}(x, y, z) \mathbf{v}_{ijk} \quad (11)$$

with the set of interpolating functions $\{\psi_{ijk}(x, y, z) = \psi_i(x)\psi_j(y)\psi_k(z); i = 0, \dots; j = 0, \dots; k = 0, \dots\}$ generated from a first-order spline function ψ by simple translation. As the indexes (i, j, k) change, the location of the function ψ_{ijk} moves along a 3D grid. For each point grid, there is only one function with non-zero value.

A general block diagram for estimation of deformations is shown in Fig. 1.

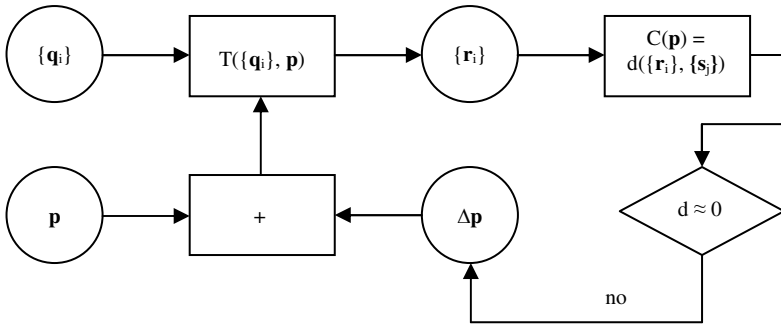


Fig. 1. Block diagram for the optimization process to estimate non-rigid deformations

5 Experiments and Results

Experiments were conducted using real medical data. The reference objects for registration were the lumen of a carotid artery, and were conducted to register MRI and Histology data sets. The most common distortion between data from MRI and data from Histology is the shrinkage of the tissue during the histological process. Thus, to perform registration of an object extracted from these modalities, non-rigid deformation between them is estimated and then the object from Histology is morphed back to match the one from MRI.

During these experiments, sets of 16 MRI imaging planes were obtained from a section of the carotid artery, with each imaging plane represented by a 512×512 pixel matrix over a field of view of $90 \text{ mm} \times 90 \text{ mm}$, with a distance of 2 mm between adjacent planes. The histological section of the same lumen was digitized to generate a set of 36 imaging planes, using a matrix of 480×512 pixels over a field of view of $170 \text{ mm} \times 180 \text{ mm}$, with variable distances between adjacent slices. Fig. 2 shows different views of the reconstruction of the lumen surface from the set of MRI images.

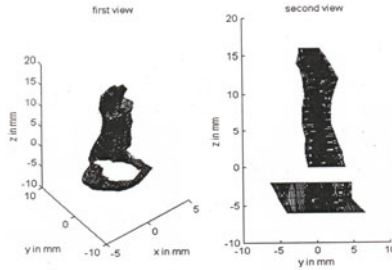


Fig. 2. Surface reconstruction from MRI data of carotid artery

Fig. 3 shows the corresponding reconstruction from Histology. Images from both modalities were used to extract feature points from regions of interest, and the criterion used to identify these features was that they had to contain structures common to both modalities, with such structures corresponding to sharp variations generally located at boundaries, edges or corners. Therefore, for each region of interest in the Histology set, there is a similar region selected from the MRI set.

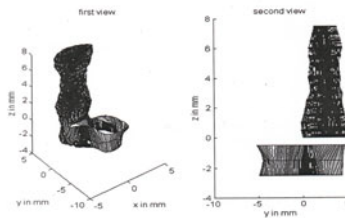


Fig. 3. Carotid artery surface reconstruction from Histology data

Surface and feature points from Histology data are matched to those from MRI by estimating non-rigid deformation between both modalities. These estimates took less than 5 minutes on a PC machine and required initialization of the Levenberg-Marquardt algorithm by setting the deformation parameters to zero. After performing 10 iterations of rigid matching followed by 40 iterations of non-rigid matching, the registered data sets appear in Fig. 4.

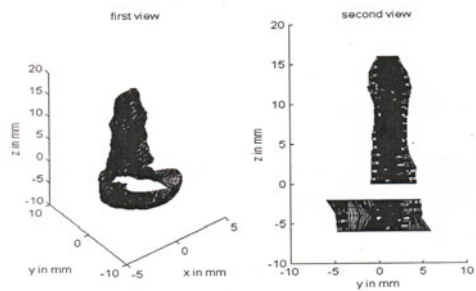


Fig. 4. Matching of histology data to MRI data

To measure the error for this matching, the distance between the set of feature points extracted from the MRI images and the set of feature points from Histology images after matching (in different combinations: Object1 from MRI - Object 2 from Histology, Object1 from MRI - Object 2 from Histology MRI, and Object1 from Histology- Object 2 from Histology) is estimated by computing the root mean square error between both sets. The average number of feature points from object 1 and objects 2 is 15 for each one. The estimated absolute errors for these experiments were 3.23, 2.1, and 1.23 mm respectively. Table 1 shows the error corresponding to the matching of two objects after performing 68 experiments for multi-modal image registration.

Table 1. Average estimated error for 68 different experiments on registration

Modalities of matching images	Number of experiments	Average error of two object matching	Average related error of object matching
MRI - Histogram	22	3.32 mm	3.68 %
MRI – MRI	34	1.23 mm	1.36 %
Histogram - Histogram	12	2.65 mm	2.94%

6 Conclusions

This paper presents a new technique for multi-modal image registration based on the estimation of non-rigid deformation in the three dimensional space. The specific case under study consists in data registering sets of data from different imaging modalities by the morphing data from histology sections to match MRI. The effectiveness and accuracy of the deformation estimates depend on the number of surface points and the number of feature points extracted from sets of medical images. An issue concerning this relation is difficult and deserves attention to develop a methodology to find the optimal number of points that gives the best estimates and does not sacrifice speed of computation. In order to obtain a set of correctly-matched feature-point pairs, our approach requires selections of similar regions of interest between two imaging modalities. Consequently, it also requires manual establishment of correspondence between two sets of features. In order to avoid manual selection of regions of interest,

we have suggested the automatic extraction of feature points from the whole region described in a sequence of images. The obtained results show satisfactory functionality of the proposal, particularly, the related error of image matching is on about 3% for different modalities of image sets with dimensions about 90x90 mm. The disadvantage consists in used simple conditions to judge the selection of feature points. One way to automatically establishing correspondence between two sets of feature points is the use of combinatorial search. This will require the development of a measurement of the similarity for two features points that must overcome the differences between two target images.

Acknowledgments. This research is sponsored by Mexican National Council of Science and Technology, CONACyT #109115 and #109417.

References

1. Mironenko, A., Song, X.B.: Image registration by minimization of residual complexity. In: IEEE Computer Soc. Conf. on Computer Vision and Pat. Recog. USA, pp. 49–56 (2009)
2. Xai, M., Liu, B.: Image Registration by Super-Curves. IEEE Transactions on Image Processing 13(5) (2004)
3. Zhu, Z., Hanson, A.R., Riseman, E.M.: Generalized Parallel-Perspective Stereo Mosaics from Airborne Video. IEEE Trans. on Pattern Analysis and Machine Intel. 26(2) (2004)
4. Adiga, U., Malladi, R., Gonzalez, R., Ortiz, C.: High-Throughput Analysis of Multispectral Images of Breast Cancer Tissue. IEEE Transactions on Image Processing 15(8) (2006)
5. Moore, C.C., et al.: Three-dimensional Systolic Strain Patterns in the Normal Human Left Ventricle: Characterization with Tagged MR Imaging. Radiology 214, 453–466 (2000)
6. Yau, H.T., Tsou, L.S., Tseng, H.M.: Automatic Registration Using Virtual Polar Ball. Computer-Aided Design & Applications 4(1-4), 427–436 (2007)
7. Pouderoux, J.: Global Contour Lines Reconstruction in Topographic Maps (2007)
8. Sumengen, B., Manjunath, B.S.: Graph Partitioning Active Contours (GPAC) for Image Segmentation. IEEE Trans. on Pattern Analysis and Machine Intelligence 28(4) (2006)
9. Lazaridis, G., Petrou, M.: Image Registration Using the Walsh Transform. IEEE Transactions on Image Processing 15(8) (2006)
10. Zayer, R., Rossli, C., Karmi, Z., Seidel, H.: Harmonic Guidance for Surface Deformation Journal: Computer Graphics Forum, vol. 24(3), pp. 601–609 (2005)
11. Kempeneers, P., et al.: Generic Wavelet-Based Hyperspectral Classification Applied to Vegetation Stress Detection. IEEE Trans. on Geoscience and Remote Sensing 43(3) (2005)
12. Kuman, R.: Snakes, Active Contour Models: Implements snakes or active contour models for image segmentation, Matmal (2010)
13. Alarcón-Aquino, V., Starostenko, O., et al.: Initialisation and Training Procedures for Wavelet Networks Applied to Chaotic Time Series. J. of Eng. Intelligent Systems 18(1), 1–9 (2010)
14. Gill, P.E., Murray, W.: Practical Optimization. Academic Press, New York (1981)
15. Borgefors, G.: Digital Transformations in Digital Images. Computer Vision, Graphics and Image Processing 34 (1986)

Performance of Correlation Filters in Facial Recognition

Everardo Santiago-Ramirez, J.A. Gonzalez-Fraga, and J.I. Ascencio-Lopez

Facultad de Ciencias, Universidad Autónoma de Baja California, Km. 103, Carretera Tijuana-Ensenada, Ensenada, Baja California C. P. 22860
{everardo.santiagoramirez, angel_fraga, ascencio}@uabc.edu.mx

Abstract. In this paper, we compare the performance of three composite correlation filters in facial recognition problem. We used the ORL (Olivetti Research Laboratory) facial image database to evaluate K-Law, MACE and ASEF filters performance. Simulations results demonstrate that K-Law nonlinear composite filters evidence the best performance in terms of recognition rate (RR) and, false acceptance rate (FAR). As a result, we observe that correlation filters are able to work well even when the facial image contains distortions such as rotation, partial occlusion and different illumination conditions.

Keywords: Facial Recognition, Correlation Filters, PSR performance.

1 Introduction

The facial biometric recognition is an important tool for the nonintrusive identification of a person. Although, this is a challengeable task because of the facial variability caused over the time, such as age sign, facial marks, beard, mustache, occlusion, and others, as to changing dress-looking wearing glasses, sunglasses, hats, scarves, and the variation of physical positions. All these problems must be taken into account in personal appearance that a face may suffer due to the different sources and facial expressions.

Biometric recognition algorithms try to match a biometric feature with a template stored in a database [1]. Thus, correlation filters are excellent candidates for the precision of matching in presence of geometric variability and tolerance to noise presented in facial images. Currently, there are few proposals that make use of correlation filters in face recognition problem, which opens a great field of research for the development of robust and efficient algorithms for face recognition.

Some advantages of the correlation filters are: a) they can use all the information of the object, i.e., both form and content (color and intensity) [2][3], b) has a good mathematical foundation [4], c) its design may include geometric distortion invariance and tolerance to certain types of noise (additive, background, light, etc.) and, d) are good candidates to be implemented with fast algorithms as the Fast Fourier Transform (FFT).

Based on these characteristics, this work presents the performances for three correlation filters: a) K-Law nonlinear composite filter, b) minimum average of correlation energy (MACE) filter and, c) average of synthetic exact filters (ASEF). The rest of paper is organized as follow. Section 2 presents the mathematical foundation of the correlation filters evaluated. Section 3 presents and discusses the performances of the Correlation Filters in Face Recognition. Finally, section 4 presents the conclusions of this work.

2 Correlation Filters

Correlation is a robust technique in pattern recognition and is used in many computer applications, such as target automatic recognition, biometrics recognition, phoneme recognition, optical character recognition [2], etc.

The correlation in the frequency domain, as show in Fig. 1, is performed by applying the fast Fourier transform (FFT) over a composite filter (synthesized with the training images set) and a test image, an element-wise multiplication of the FFT of the test image and the filter and, the inverse of fast Fourier transform (IFFT) of this product yield the correlation output.

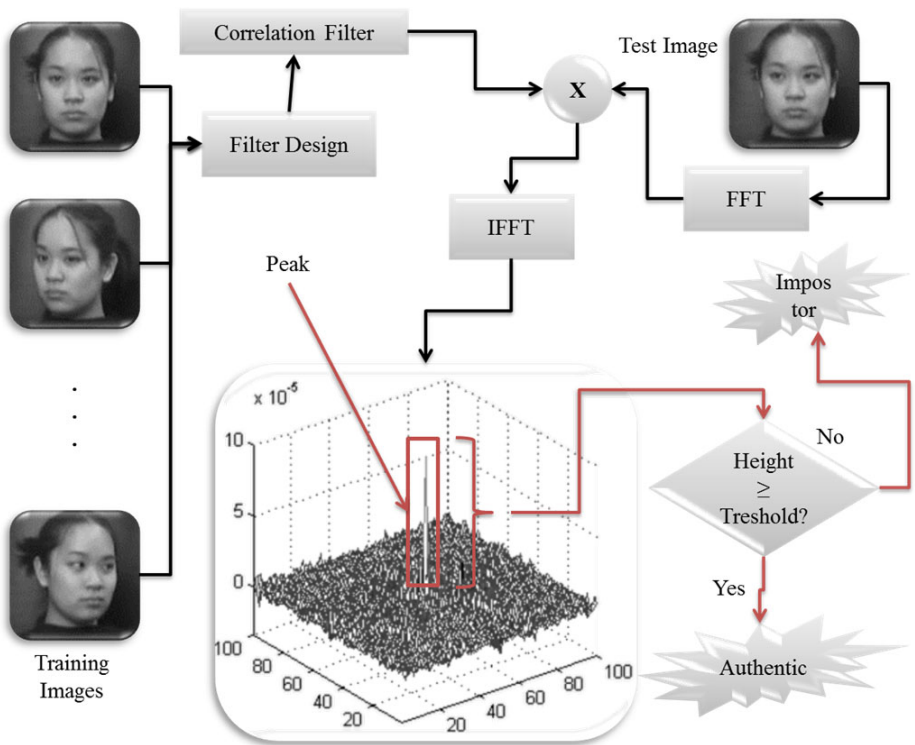


Fig. 1. Facial recognition process by composite correlation filters

A well-designed filter produces a sharp correlation peak for the true class objects (targets known). To determine a matching between a biometric template and a test facial image, it is necessary to measure the sharpness correlation peak. A good measure for the peak sharpness is the peak-to-sidelobe ratio (PSR), which is presented in equation 1. In this work, PSR is obtained from a 11×11 window centered at the correlation peak.

$$PSR = \frac{(\text{peak} - \text{mean of side lobe region})}{\text{Standard deviation of side lobe region}} \quad (1)$$

2.1 K-Law Nonlinear Composite Filter

Let $x_1(x, y), x_2(x, y), \dots, x_N(x, y)$ as the N training images with d pixels each one. We rearrange each image as a column vector by lexicographic ordering, i.e., from left to right and top to bottom. This operation produces a vector with d elements. Let X be a matrix of dimension d rows $\times$ N columns, where each column is a training image. Thus the expression for a basic composite SDF filter is [5]:

$$h = X(X^+X)^{-1}u, \quad (2)$$

where $+$ is the transpose conjugate, -1 is the inverse of the matrix and u is a vector that contains correlation values desired for each image in the training set. Generally, the vector u is assigned values of 1 for true class objects, while 0 is assigned to the false class objects.

To improve the performance of composite filters in terms of discrimination against objects similar to the target, correlation-peak sharpness, and correlation noise robustness we apply nonlinear filtering techniques to composite filters such as in reference [6].

In order to apply the nonlinearity on Fourier domain to the filter in equation 2, let $\tilde{X}$ a matrix which contains in column vector form all Fourier transforms of the training images. When a non-linearity is applied to the matrix, the nonlinear operation is applied on each element of the matrix. Hence, the K-Law nonlinearity applied to the matrix $\tilde{X}$ can be described as:

$$\tilde{X}^k = |\tilde{X}|^k \exp [i\varphi_{\tilde{X}}], \quad 0 \leq k \leq 1, \quad (3)$$

where $|\tilde{X}|$ is the modulus of $\tilde{X}$ and φ is its phase. The k value controls the strength of the non-linearity. Now, modifying the equation 2 for build a K-Law SDF composite filter is obtained [6]

$$\tilde{H}^k = \tilde{X}^k (\tilde{X}^{k+} \tilde{X}^k)^{-1} u \quad (4)$$

When the filter is synthesized with only one image and by setting $k = 1$ we get a classical matched filter and when $k = 0$, establishing the magnitude of all frequencies to 1, thus we get a phase-only filter,

The nonlinear operator raises to the k -th power the magnitude of the Fourier spectrum of both the analyzed image and the filter, while keeping intact the phase information. This characteristic cause that the filter have a good discrimination capacity. Based on many experiments we determined that the nonlinearity factor $k=0.3$ offers the best performance for k-law filter in facial recognition.

The K-Law filters, using appropriate images, may be scale and rotation invariant, distortions presented for test images. Also, they have greater tolerance to additive noise. Also, this filter yields a correlation plane with a sharp and strength peak while that a conventional SDF filter doesn't. For this reasons, we chose the filter K-Law for to evaluate his performance in facial recognition problem.

2.2 Minimum Average Correlation Energy

The MACE filter was developed for minimize large side-lobes produced by the SDF filters and is represented in equation (5) [2]:

$$h = D^{-1}X(X^+D^{-1}X)^{-1}u \quad (5)$$

Let a training set $\{x_1, x_2, x_3, \dots, x_N\}$ containing N images with d pixels each one, then the average energy over all training images is the diagonal matrix $D = \frac{1}{d.N} \sum_{i=1}^N \tilde{x}_i \tilde{x}_i^*$ (size $d \times d$). $\tilde{x}_i$ is the FT of each training image and $*$ represents the complex conjugate.

2.3 Average of Synthetic Exact Filters

For the ASEF filter, each training image x_i is associated to a desired output y_i , the desired output y_i is an output plane of correlation with the peak center over the object of interest. For each pair x_i, y_i , an exact filter is design as follow:

$$H_i^*(w, v) = \frac{\tilde{y}_i(w, v)}{\tilde{x}_i(w, v)} \quad (6)$$

where the division is element by element between the FT of the output objective $\tilde{y}_i$ and the FT of the training image $\tilde{x}_i$. The ASEF filter is presented in equation (7) [7]:

$$H_\mu^*(w, v) = \frac{1}{N} \sum_{i=1}^N H_i^*(w, v) \quad (7)$$

3 Performances of the Correlation Filters in Face Recognition

In computer simulations we considered a target set of 40 known subjects and a query set of 400 facial images of unknown subjects for identification. Each target subject set has 10 facial images for build a filter. The facial identification is performed by cross-correlating a filter with each of the query set and processing the 400 correlation outputs on which only 10 facial images belongs to an authentic subject. Each one of the correlation output is searched for peaks and height of these peaks are used to determine whether the facial image is of an authentic or impostor subject.

The threshold for an authentic subject is setup with the smallest PSR value of the 10 correlation outputs of the authentic subject. Thus, performed a cross-correlating 40 filters with the query set, we obtain 16000 correlation outputs, of which only 400 are of authentic subjects.

Some facial images that contain the target set and query set has distortions such as: facial expressions, small degrees of rotation, profile (left end right) and, partial occlusion caused by sunglasses, beard and mustache. Sample ORL training face images are shown in Fig. 2 [9]. Each facial image was cropped and scaled manually to a resolution of 64×64 pixels.

One of the main problems facing the facial processing systems is the variation in lighting [10]. To address this problem, it has been proposed methods that combine samples of facial images affected by illumination [11], while in [12] performed normalization in facial images. In this paper we used the logarithmic transformation to improve the intensity of the pixels in the shaded region and even with the light regions as described in [10]. Fig. 3 shows how the log transformation improves illumination in facial images.



Fig. 2. Sample ORL data set. Training set for subject 27 (top) and subject 30 (bottom).

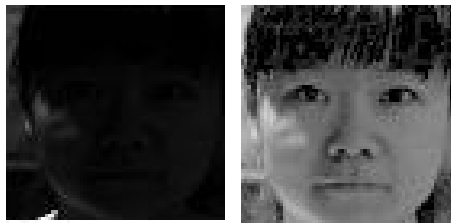


Fig. 3. Logarithmic transformation for improve the intensity of the pixels in shaded areas. Left: original image and right: image improved.

An advantage to establishing a PSR threshold for each authentic subject is that false rejection rate is low and the accuracy of algorithms based in correlation is significantly improved. Other advantage on this procedure is that in case that an algorithm receives as input a test facial image that improve the PSR threshold established for these subject, then is possible run an update to the filter over the time, getting a precise and exact recognition. Figs. 4 and 5 shows box plots for the best K-Law filter PSR performance (subject 27, showed in fig. 2) and the worst PSR performance (subject 1), respectively.

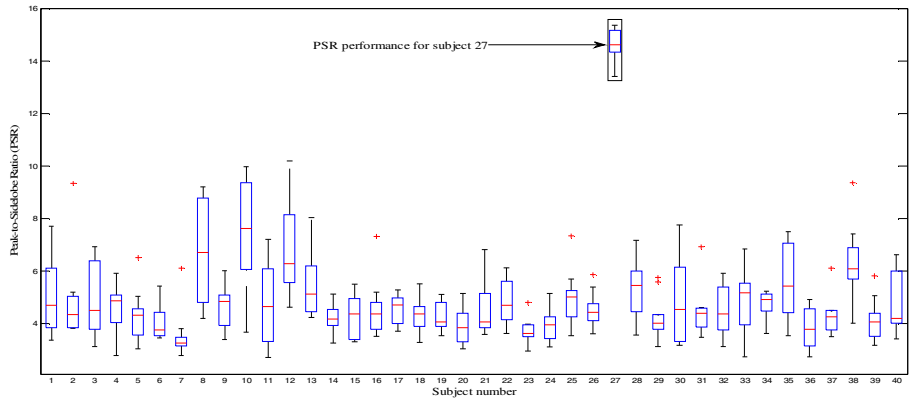


Fig. 4. Best K-Law filter PSR performance (subject 27)..

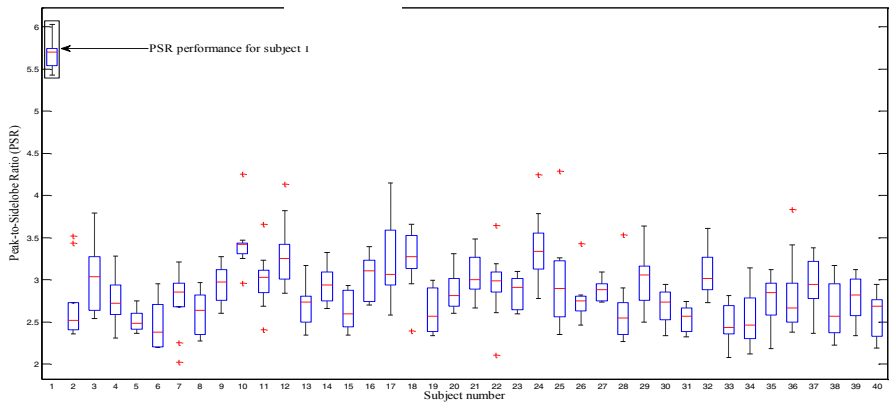


Fig. 5. Worst K-Law filter PSR performance (subject 1)

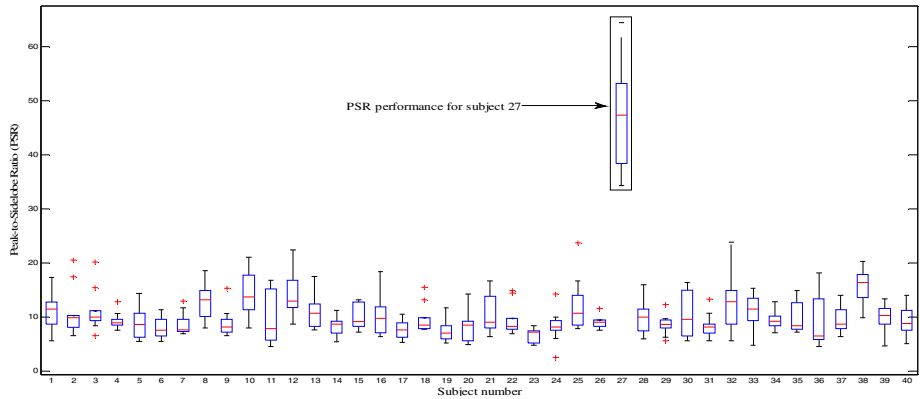


Fig. 6. Best MACE filter PSR performance (subject 27)

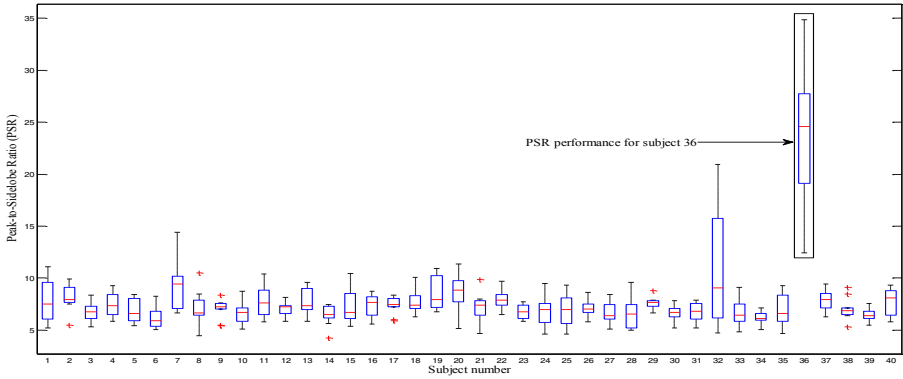


Fig. 7. Worst MACE filter PSR performance (subject 36)

Figs. 6 and 7 shows the best and worst MACE filter PSR performance. Although [13] obtained a recognition rate of 100% with the MACE filter, in these study only was considered facial images with facial expressions, no other distortions used as in this work.

ASEF was originally applied in eye localization, showing a good performance. This paper presents the results of this filter applied in facial recognition. The experiment showed that this filter produces PSR values greater than or equal to 14 for authentic subjects. Figs. 8 and 9 show the box plots for the best and the worst ASEF filter PSR performance, respectively. This filter shows a poor discriminating capacity, it produce PSR values greater that threshold for impostors subjects and, do not recognized correctly many facial images of authentic subjects.

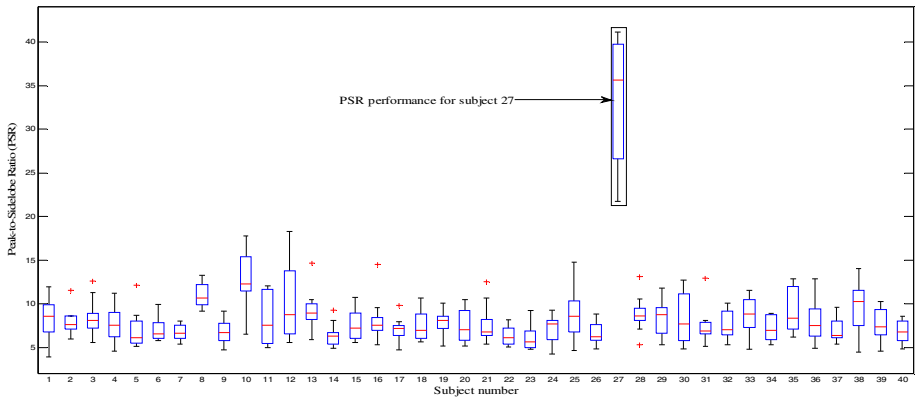


Fig. 8. Best ASEF filter PSR performance (subject 27)

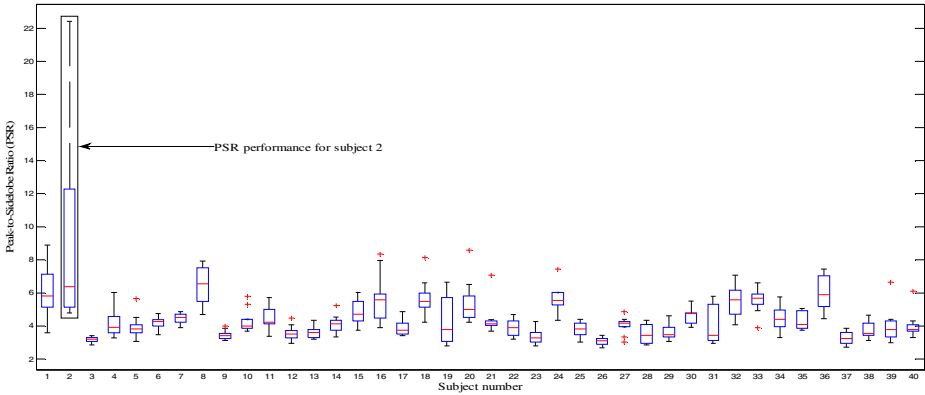


Fig. 9. Worst ASEF filter PSR performance (subject 2)

The performance of correlation filters in face recognition is summarized in Table 1. As can be see, the K-Law nonlinear composite filter with a nonlinear factor $k = 0.3$ offers the best performance with a recognition rate of 100%. The filter K-Law states two different facial images of different subjects as equals the 0.2%.

Where FAR (False Accept Rate) is the percentage of time that is reported as equal to two individuals who are not, FRR (False Rejection Rate) which is the percentage of times that two equal individuals are wrongly considered to be different and, RR (Recognition Rate) which is the percentage of times it correctly recognizes a subject and AC (Accuracy) is the proportion of the total number of predictions that were correct by the classifier.

Table 1. Performance of correlation filters in face recognition

Correlation filter	FAR %	FRR %	RR%	AC%
K-Law	0.2	0	100	99.50
MACE	11.5	0	100	89.69
ASEF	316	0	100	24.04

4 Conclusions

This document provides a brief assessment of the performance of correlation filters in facial recognition problem. The experiments show that K-Law nonlinear composite filter with logarithmic transformation to the facial images, both test and training achieved a recognition rate of 97.5%. The algorithm that implemented this filter had a 99.50% of accuracy. It also shows that the quality of images affected from variations in lighting was improved by applying the logarithmic transformation. Advantages of the correlation method include shift-invariance and ability to suppress impostor faces using a PSR threshold. We are currently improving the filters design methods and testing the correlation filters on the much larger database, including different pose, illumination, expressions, scale and rotation variations.

Acknowledgments. Credit is hereby given to the Massachusetts Institute of Technology and to the Center for Biological and Computational Learning for providing the ORL database. This work was financed by CONACYT through the scholarship provided by the first author (CONACYT 45360/344833). This work has been developed within the program Maestría y Doctorado en Ciencias e Ingeniería (MyDCI) at UABC.

References

1. National Science and Technology Council, <http://biometrics.gov>
2. Vijaya Kumar, B., Mahalanobis, H., Juday, R.: Correlation Pattern Recognition. Cambridge University Press, New York (2005)
3. Gonzalez-Fraga, J.A., Kober, V., Alvarez Borrego, J.: Adaptive Synthetic Discriminant Function Filters for Pattern Recognition. *Optical Engineering* 45, 057005 (2006)
4. VanderLugt, A.B.: Signal detection by complex spatial filtering. *IEEE Transactions Information Theory* 10, 139–145 (1964)
5. Casasent, D., Chang, W.: Correlation synthetic discriminant functions. *Applied Optics* 25, 2343–2350 (1986)
6. Javidi, B., Wang, W., Zhang, G.: Composite Fourier-plane nonlinear filter for distortion-invariant pattern recognition. *Optical Engineering* 36, 2690 (1997)
7. Bolme, D.S., Draper, B.A., Ross Beveridge, J.: Average of Synthetic Exact Filters. Computer Science Department Colorado State University, Fort Collins (2010)
8. Samaria, F., Harter, A.: Parameterization of a stochastic model for human face identification. In: 2nd IEEE Workshop on Applications of Computer Vision, Sarasota (1994)
9. Savvides, M., Vijaya Kumar, B.V.: Illumination normalization using logarithm transforms for face authentication. In: Kittler, J., Nixon, M.S. (eds.) AVBPA 2003. LNCS, vol. 2688, pp. 549–556. Springer, Heidelberg (2003)
10. Sim, T., Kanade, T.: Combining Models and Exemplars for Face Recognition: An Illuminating Example. In: Proceedings of the CVPR (2001)
11. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs Fisherfaces: Recognition Using Class Specific Linear Projection. In: PAMI-19 (1997)
12. Savvides, M., Vijaya Kumar, B.V., Khosla, P.: Face Verification using Correlation Filters. In: Proc. of the third IEEE Automatic Identification Advanced Technologies, Tarrytown, NY, pp. 56–62 (2002)

Evaluation of Binarization Algorithms for Camera-Based Devices

M. Nava-Ortiz, W. Gómez-Flores, A. Díaz-Pérez, and G. Toscano-Pulido

Information Technology Laboratory, CINVESTAV-IPN, Ciudad Victoria, Mexico
`mnavatamps.cinvestav.mx`

Abstract. Segmentation is an important step within optical character recognition systems, since the recognition rates depends strongly on the accuracy of binarization techniques. Hence, it is necessary to evaluate different segmentation methods for selecting the most adequate for a specific application. However, when gold patterns are not available for comparing the binarized outputs, the recognition rates of the entire system could be used for assessing the performance. In this article we present the evaluation of five local adaptive binarization methods for digit recognition in water meters by measuring misclassification rates. These methods were studied due to of their simplicity to be implemented in based-camera devices, such as cell phones, with limited hardware capabilities. The obtained results pointed out that Bernsens method achieved the best recognition rates when the normalized central moments are employed as features.

Keywords: local adaptive binarization, optical character recognition, camera-based devices, feature selection.

1 Introduction

Optical Character Recognition (OCR) has been widely studied for decades, specially in document image analysis. In many situations only digit characters are required to be recognized. For instance, a common use of OCR is recognizing numbers in car plates [12]. Another useful application developed for helping persons who are blind or have low vision was presented by Shen and Coughlan [3]. The authors implemented an OCR algorithm for camera cell phones that read aloud the numbers presented in LCD/LED displays.

As one can note, the availability of cheap portable devices with suitable computation power makes possible to process images in real time. Also, the built-in cameras in mobile devices (e.g. smart phones) give additional capabilities to implement OCR algorithms.

However, the task of recognizing characters acquired with cameras is not trivial, since uncontrolled environment variables, such as uneven illumination and shadows, leads low quality images. Moreover, compared to general desktop environment, most camera-based mobile devices have limitations of low computing

power and insufficient storage space. Also its CPUs are only integer processor and therefore the floating point operations must be simulated in integer arithmetic operations, degrading processing rate seriously.

The paradigm of image processing can be depicted in terms of five basic steps: (i) image acquisition, (ii) image preprocessing (noise filtering and enhancement), (iii) image segmentation, (iv) feature extraction and selection, and (v) object recognition [7]. For assessing the performance of each step, several authors have been proposed techniques for measuring the quality of their algorithms. For example, Tin and Kamata [4] presented an iterative image enhancement algorithm and developed a new evaluation framework based on objective and subjective quality image measures. Besides, Thulke et al. [5] proposed a general evaluation approach for comparing document segmentation algorithms based on the segments directly. Obviously, these evaluation approaches need to define the range of values that represent the levels of efficiency.

The image segmentation is a critical step within the recognition process. It is convenient to use references to compare the output of the segmentation algorithm for evaluating its performance. However, when the references are not available, the recognition rates of the complete system could be used for evaluating the segmentation performance. Trier and Jain [6] used this approach to evaluate recognition accuracy of an OCR system, by comparing different binarization methods.

Numerical meters, such as wattmeters or water meters, are devices used to measure the amount of some service commonly supplied by a public utility. Generally, the records from these devices are captured manually by employees. However, this procedure may cause wrong readings due to human errors. To overcome this inconvenient, an OCR system could be implemented in camera-based devices handled by public utility employees. In order to deal with the hardware limitations and uncontrolled environment variables, it is necessary to develop adaptive methods that do not demand a lot of computational resources.

In this article we present the evaluation of five local adaptive binarization methods for digit recognition in water meters by measuring misclassification rates. The methodology involves the five steps in image processing paradigm for object recognition. First, the photograms coming from two brands of water meters were cropped to separate each digit to create the image data set. Next, five local adaptive binarization methods were applied to the entire dataset. Thereafter, seven scale invariant moments were calculated from the segmented images. Besides, a feature selection method based on mutual information and intrinsic dimensionality was applied to the space of characteristics. This procedure took the first three moments ranked with maximum-relevance-minimum-redundance (MRmR) criteria [16]. Moreover, the digit recognition was performed by a minimum distance classifier. Finally, we measured the recognition rates for determining which is the most adequate binarization method for this particular application. Additionally, we made a comparison using the selected features and all of them.

2 Materials and Methods

2.1 Binarization Methods

An image with size $M \times N$ can be represented by a 2D gray-level intensity function $f(x, y)$, within the range 0 to $L - 1$, where L is the maximum number of gray-levels. A binarization method creates bilevel images from $f(x, y)$ by turning all pixels below some threshold to zero (black) and all pixels equal to or greater than that threshold to one (white). This procedure separates the image into two classes: background and foreground. The main problem is how to select the optimal or most adequate threshold. In general, the binarization methods could be classified into two main approaches: global and local adaptive [7]. The former attempts to find a single threshold value for the overall image, whereas the latter computes a threshold value for each pixel regarding its neighborhood information. Due to local methods are able to cope with uneven illumination, we tested five local binarization method depicted bellow. The objective is to evaluate the impact of the each segmentation method in the recognition process. All the algorithms depicted in this article were developed in Matlab 7.10 (The MathWorks Inc., Natick, Mass, USA).

Bernsen's Method. [8] determines the threshold for the pixel (x, y) as the average between the minimum and maximum gray level pixel values in a square $b \times b$ neighborhood centered at (x, y) . However, if the contrast $C(x, y) = I_{max}(x, y) - I_{min}(x, y)$ is lower than a predetermined threshold t , then the pixel (x, y) is labeled as background. The threshold is caculated as follows:

$$T(x, y) = (I_{min} + I_{max}) / 2 . \quad (1)$$

Niblack's Method. [9] obtains the threshold values by calculating the local mean and local standard deviation. The threshold at pixel (x, y) is calculated as:

$$T(x, y) = m(x, y) + k \cdot s(x, y), \quad (2)$$

where $m(x, y)$ and $s(x, y)$, are the mean and standard deviation, respectively, in a local neighborhood of pixel (x, y) . It is necessary define the size of the neighborhood small enough in order to preserve local details, and large enough to supress noise. The value of k is used to adjust how much of the total print object boundary is taken as a part of the given object.

Sauvola's Method. is an improvement to Niblack's method [10]. It attempts to efficiently reduce the non-homogeneous illumination in image. This algorithm is not sensitive to the value of k parameter. Thus, the threshold at pixel (x, y) is computed as:

$$T(x, y) = m(x, y) + \left[1 + k \cdot \left(\frac{\sigma(x, y)}{R} - 1 \right) \right], \quad (3)$$

where $m(x, y)$ and $s(x, y)$ are the mean and standard deviation, respectively. The R value is the dynamic range of the standard deviation and for this study was set as 8. The k parameter is positive.

Wellner's Method. [11] applies image smoothing to the input image by using an average filter with a window size $b \times b$. Next, the threshold pixel (x, y) is calculated as:

$$T(x, y) = J(x, y) \cdot \left[1 - \left(\frac{t}{100} \right) \right], \quad (4)$$

where $J(x, y)$ is the filtered image and t is the predetermined threshold that scales each gray value of the filtered image to a lower value.

White's Method. [12], each gray value at pixel (x, y) is compared with its neighborhood average value, in a square $b \times b$. If pixel (x, y) is significantly darker than its neighborhood mean, then is classified as foreground, otherwise it is classified as background as follows:

$$B(x, y) = \begin{cases} 1 & \text{if } m_{b \times b}(x, y) < I(x, y) \cdot w \\ 0 & \text{otherwise} \end{cases}, \quad (5)$$

where $I(x, y)$ is the original image, $m_{b \times b}(x, y)$ is the local mean, and $w > 1$ is the *bias* value.

2.2 Image Acquisition

The dataset includes images captured from two brands of water meters. The NOKIA N80 cell phone was used to acquired the 8-bit images. Moreover, all the images were taken directly from the electronic viewfinder to speed up the image time-processing. As the images presented complete lectures (Fig. 1), it was necessary to crop the digits for separating each number in categories, that means, 0 to 9. Then, the size of each single image containing a number was 19×30 pixels. The total number of images within the dataset was 1,418.



Fig. 1. Examples of images captured from two brands of water meters

2.3 Image Binarization

The entire dataset was binarized by the five thresholding methods depicted previously: Bernsen [8], Niblack [9], Sauvola [10], Wellner [11], and White [12]. There was not necessary to preprocess the images before segmentation procedure, since the local adaptive methods are able to suppress some amount of noise. Despite

each segmentation method is performed as local adaptive, they depend on the tuning of a global parameter that will be applied indistinctly to overall pixels. For instance, Niblack's and Sauvola's methods depends on k parameter, Bernsen's and Wellner's methods depends on t threshold parameter, and White's method depends on a w bias value. If we see all these variables as a single parameter to be tuned, we can denote all of them as K . We propose to adapt the K value as follows [13]:

$$K = c \cdot \frac{m_g(i, j)\sigma_g(i, j) - m_l(i, j)\sigma_l(i, j)}{\max[m_g(i, j)\sigma_g(i, j), m_l(i, j)\sigma_l(i, j)]}, \quad (6)$$

where $m_g(i, j)$ and $m_l(i, j)$ are the global and local mean values, respectively, and $\sigma_g(i, j)$ and $\sigma_l(i, j)$ are the global and local standard deviation values, respectively. c is a parameter to keep the value of K within the range of threshold values for each pixel in the image regarding the binarization method. For instance, in Bernsen's method a contrast image is computed by subtracting local maximum and minimum gray-levels. Then, if the contrast is below a certain global threshold, then a pixel is said to consist only one class, object or background. Generally, for Bernsen's method the global threshold is set at gray value 15 for 8-bit images. However, this threshold will affect all pixels in the contrast image evenly. Therefore, we proposed to adapt that threshold by using local information where the c parameter keeps the threshold around the gray value 15 and, consequently, the threshold will be adapted for each pixel in the contrast image. This strategy was applied to overall binarization methods, considering the typical global threshold values used for each one. The values for the c parameter were set as follows: Bernsen, 15; Niblack, 0.1; Sauvola, 0.01; Wellner, 0.1; and White, 0.1. Besides, empirical data obtained from experiments varying the window size, reveal that using a window of 5×5 pixels for all methods, produces better results in terms of recognition accuracy.

After performing the digit binarization some undesired regions remain around the binarized number. We made some assumptions about the image to design the strategy for eliminating the noisy regions. First, we assumed that the area of the digit object is greater than any other object in the image. And second, the digit object is centered on the image centroid. Thus, the strategy for eliminating undesired regions involves 3 steps:

1. Label the objects within the image with 4-connectivity.
2. Calculate the centroid of the image.
3. Measure both the area (A) of all labeled regions and the mean Euclidian distance (D) from all pixels within the region to the center of the image. Next, calculate the ratio $(D - A)/(D + A)$ for each region. We assumed that the object with the minimum ratio value correspond to the binarized digit and remains within the image, whereas the other regions were automatically eliminated.

Thereafter, we decided to compare the performance of applying additional post-processing operations to the binary images, just before we extract the features.

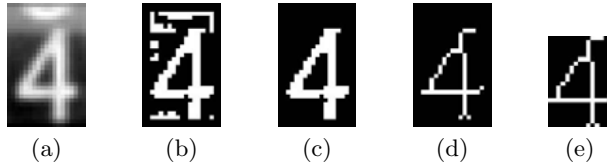


Fig. 2. Segmentation process. (a) Input image, (b) Binarized image by Niblack's method, (c) Binarized image after cleaning strategy, (d) Skeleton of the digit, and (e) Minimum bounding box containing the digit used for classification purposes.

Thus, it was computed the skeleton of the binarized digit and, at last, the image size was adjusted to the minimum bounding box containing the thinned digit. In Fig. 2 is illustrated the segmentation process for Niblack's method using skeletonized digits. Since skeletonization can be made by using mathematical morphology, in specific through successive erosions, the computacional requirements are lower than those required by other algorithms.

2.4 Feature Extraction

The feature extraction process consists in computing attributes from the objects of interest to get some quantitative information for differentiating one class of objects from another. The attributes used in this work were the normalized central moments, denoted η_{pq} , which are defined as [14]:

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^\gamma}, \quad (7)$$

where

$$\gamma = \frac{p+q}{2} + 1,$$

and μ_{pq} correspond to the central moments of order $(p + q)$, and for binary images they are expressed as:

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x - \bar{x})^p (y - \bar{y})^q, \quad (8)$$

for $p = 0, 1, 2, \dots$ and $q = 0, 1, 2, \dots$, M and N are the width and height of the image, respectively, and $(\bar{x}, \bar{y})$ is the center of mass of the object. Herein, we used the first 3 orders, resulting in 7 normalized central moments: η_{11} , η_{20} , η_{02} , η_{30} , η_{03} , η_{21} , η_{12} .

2.5 Feature Selection

In many pattern recognition problems, a higher number of used features (or attributes) do not necessarily translate into higher classification accuracy. Therefore, feature selection is the process commonly used for removing irrelevant and

redundant features while maintaining acceptable classification accuracy. An irrelevant feature does not contribute to distinguish data of different classes and can be removed without affecting the classification performance. On the other hand, a redundant feature implies the co-presence of another feature, being both attributes relevant, but the removal of one of them will not affect learning performance [15].

In this article, a feature selection technique based on mutual information and intrinsic dimensionality was tested to reduce the space of attributes, which was developed by our group and whose technical details can be found in [16]. We employed a mutual information scheme based on minimal-redundancy-maximal-relevance (mRMR) criterion to rank the input data. Besides, the intrinsic dimensionality of the feature space was calculated by using principal component analysis (PCA). Thus, when using ranking features algorithms, the intrinsic dimensionality could estimate automatically the number of m features to be introduced into the classifier.

2.6 Digit Recognition

Although it is one of the earliest methods suggested, the minimum distance classifier is still an effective tool in solving the pattern recognition problem. With the minimum distance classifier data belonging to a class, are assumed to be represented by the mean value of this class. Suppose that we define the prototype of each pattern class to be the mean vector of the patterns of that class:

$$\mathbf{m}_j = \frac{1}{N_j} \sum_{\mathbf{x} \in \omega_j} \mathbf{x}_j, \quad j = 1, 2, \dots, W, \quad (9)$$

where W is the number of pattern classes, N_j is the number of pattern vectors from class ω_j and the summation is taken over these vectors. One way to determine the class membership of an unknown pattern vector $\mathbf{x}$ is to assign it to the class of its closest prototype. Using the Euclidian distance to determine closeness reduce the problem to computing the distance measures:

$$D_j(\mathbf{x}) = \|\mathbf{x} - \mathbf{m}_j\|, \quad j = 1, 2, \dots, W. \quad (10)$$

We then assign $\mathbf{x}$ to class ω_j if $D_j(\mathbf{x})$ is the smallest distance [7].

3 Results

Each digit class (0 to 9) of the entire dataset was segmented by using the five binarization methods depicted in Section 2.1. Thereafter, normalized central moments were calculated from each single segmented image to create the feature space. Next, feature selection procedure described in Section 2.5 determines automatically that the first 3 ranked normalized central moments are enough for classifying. The results of this stage are presented in Table 1.

Table 1. Selected features based on mutual information and intrinsic dimensionality

Method Name	Bernsen	Niblack	Sauvola	Wellner	White
Selected features	$\eta_{12}, \eta_{02}, \eta_{11}$	$\eta_{02}, \eta_{12}, \eta_{11}$	$\eta_{02}, \eta_{12}, \eta_{30}$	$\eta_{12}, \eta_{02}, \eta_{30}$	$\eta_{02}, \eta_{21}, \eta_{30}$

For the classification stage, we compare the approach of using just the selected characteristics and all of them, separately. Thus, the technique of cross-validation divided randomly both, the selected and the whole dataset, in 70% for training and 30% for testing. This procedure was performed 500 times for each segmentation method. Table 2 and Table 3 show the percentage of digit classes correctly classified for each binarization method, when using feature selection and when employed the complete set of features.

Table 2. Percentage of recognition rates for each digit class for the five binarization methods, using the selected moments. The results are the mean values of 500 cross-validation runs.

Method	% Recognition rate in each class										Mean	σ	CV
	0	1	2	3	4	5	6	7	8	9			
Bernsen	85.4	91.6	83.6	94.3	91.8	85.2	78.1	79.8	81.1	92.1	86.3	5.8	0.07
Niblack	83.4	79.8	75.0	83.5	87.5	79.1	80.0	74.1	84.4	85.8	81.3	4.4	0.05
Sauvola	81.3	86.7	80.7	86.1	81.6	74.7	72.4	73.6	86.7	88.2	81.2	5.9	0.07
Wellner	86.2	72.8	68.8	87.6	76.7	69.0	79.7	72.3	85.0	88.8	78.7	7.8	0.10
White	77.7	97.0	82.4	89.1	85.3	75.2	70.3	88.1	76.7	79.2	82.1	7.9	0.10

These results pointed out that the Bernsen’s method achieved the best recognition rates in both cases, by using feature selection and considering the whole feature space, with a total mean values of $86.3\pm5.8\%$ and $90.0\pm9.6\%$, respectively. This suggests that Bernsen’s method is capable to preserve better the digits attributes than the other four methods, when the normalized central moments are employed. Besides, we used the coefficient of variation (CV) for measuring the dispersion of probability distribution for each binarization technique. One can note that the results obtained with feature selection presented lower variance for all segmentation methods than when using the entire feature space.

An additional computational issue considered to compare the binarization methods was the consuming time processing. Thus, we obtained the elapsed time during the binarization operations without considering the cleaning strategy. For these experiments we used a PC with dual core AMD CPU operated in 2.10 Ghz and Linux OS. The Table 4 shows the sum of running times for each binarization method considering all the images in our dataset. Now, through this comparison it is noticeable that Bernsen’s method, that obtained the best

Table 3. Percentage of recognition rates for each digit class for the five binarization methods, using all the moments. The results are the mean values of 500 cross-validation runs.

Method	% Recognition rate in each class										Mean	σ	CV
	0	1	2	3	4	5	6	7	8	9			
Bernsen	70.1	96.2	92.7	99.0	98.4	84.0	83.1	82.5	96.9	96.9	90.0	9.6	0.11
Niblack	69.8	83.8	80.3	89.0	97.0	69.5	77.9	84.9	97.2	93.7	84.3	10.1	0.12
Sauvola	68.9	86.4	87.8	86.8	95.1	75.2	78.9	80.6	97.9	92.5	85.0	9.1	0.11
Wellner	68.9	77.9	77.0	87.4	95.9	68.8	71.4	82.3	97.1	91.9	81.9	10.8	0.13
White	76.0	99.8	81.9	93.6	96.4	85.0	84.2	90.8	95.1	96.4	89.9	7.7	0.09

accuracy, used the largest time to make the recognition. Since our future goal is to implement an OCR system in a mobile platform with limited hardware capabilities, White’s method could be used instead of Bernsen’s method as it has an adequate execution time and produces the second better accuracy results.

Table 4. Execution times of the binarization methods under study

Method Name	Bernsen	Niblack	Sauvola	Wellner	White
Execution time (seg)	16.46	7.04	6.55	6.40	6.04

4 Conclusion

In this article we evaluated five binarization methods in terms of recognition rates. These methods were studied because of their simplicity to be implemented in based-camera devices with limited hardware capabilities. Thus, the objective was to investigate which method is able to preserve important information about the digit nature. We used seven normalized central moments to differentiate quantitatively among digits. Also, there was employed a feature selection technique (for reducing the data dimensionality) as well as the entire feature set. Despite classification rates apparently are better when using all features, the feature selection reduces the problem dimensionality and kept a more stable recognition response with an acceptable misclassification rate by using only 3 moments from the original 7 moments. The representation of an object is not trivial, and for future work we are planning to use other kind of attributes for reaching better recognition rates (>90%) by using the same minimum distance classifier. The final objective of our investigation is to implement an OCR system in a camera-based cell phone, with limited hardware capabilities, for recognizing digits of water meters.

References

1. Anagnostopoulos, C.N.E., Anagnostopoulos, I.E., Loumos, V., Kayafas, E.: A License Plate-Recognition Algorithm for Intelligent Transportation System Applications. *IEEE Transactions on Intelligent Transportation Systems*, 377–392 (2006)
2. Ji-yin, Z., Rui-rui, Z., Min, L., Yin, L.: License Plate Recognition Based on Genetic Algorithm. In: *International Conference on Computer Science and Software Engineering*, pp. 965–968 (2008)
3. Coughlan, J.H.S.: Reading lcd led displays with a camera cell phone. In: *Conference on Computer Vision and Pattern Recognition Workshop (CVPRW 2006)*, pp. 119–119. *IEEE Computer Society, Washington, DC, USA* (2006)
4. Tian, L., Kamata, S.: An iterative image enhancement algorithm and a new evaluation framework. In: *IEEE International Symposium on Industrial Electronics (ISIE 2008)*, pp. 992–997 (2008)
5. Thulke, M., Margner, V., Dengel, A.: Quality evaluation of document segmentation results. In: *Proceedings of the Fifth International Conference on Document Analysis and Recognition, ICDAR 1999* (1999)
6. Trier, O.D., Jain, A.K.: Goal-directed evaluation of binarization methods. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 1191–1201 (1995)
7. Gonzalez, R.C., Woods, R.E.: *Digital Image Processing*. Prentice-Hall, New Jersey (2002)
8. J. Bernsen.: Dynamic thresholding of gray-level images. In: *Proc. Eighth Int. Conf. Pattern Recognition*, pp. 1251–1255 (1986)
9. Niblack, W.: *An Introduction to Digital Image Processing*, pp. 115–116. Prentice Hall, Englewood Cliffs (1986)
10. Sauvola, J., Pietikinen, M.: Adaptive document image binarization. *Pattern Recognition* 33(2), 225–236 (2000)
11. Wellner, P. D.: Adaptive Thresholding for the DigitalDesk. Technical Report EPC-1993-110, Rank Xerox Ltd. (1993)
12. White, J.M., Rohrer, G.D.: Image thresholding for optical character recognition and other applications requiring character image extraction. *IBM J. Research and Development* 27(4), 400–411 (1983)
13. Rais, N.B., Hanif, M.S., Taj, I.A.: Adaptive thresholding technique for document image analysis. In: *Proc. 8th International Multitopic Conference*, pp. 61–66 (2004)
14. Huang, Z., Leng, J.: Analysis of hu's moment invariants on image scaling and rotation. In: *2nd International Conference on Computer Engineering and Technology (ICCET 2010)*, vol. 7, pp. 476–480. *IEEE Computer Society, Los Alamitos* (2010)
15. Liu, H., Motoda, H.: *Computational Methods of Feature Selection*. Taylor & Francis, Boca Raton (2008)
16. Gómez, W., Leija, L., Díaz-Pérez, A.: Mutual Information and Intrinsic Dimensionality for Feature Selection. In: *7th International Conference on Electrical Engineering, Computing Sciences and Automatic Control (CCE 2010)*, Tuxtla Gutiérrez, Chiapas, September 8–10, pp. 339–344 (2010)

A Hybrid Approach for Pap-Smear Cell Nucleus Extraction

M. Orozco-Monteagudo¹, Hichem Sahli², Cosmin Mihai²,
and A. Taboada-Crispi¹

¹ Universidad Central de Las Villas, Cuba
morozco@uclv.edu.cu, ataboada@uclv.edu.cu

² Vrije Universiteit Brussel, Electronics and Informatics Dept. - ETRO,
Pleinlaan 2, 1050 Brussels, Belgium
hsahli@etro.vub.ac.be, cmihai@etro.vub.ac.be

Abstract. This paper, proposes a two-phases approach for a computer-assisted screening system that aims at early diagnosis of cervical cancer in Pap smear images and accurate segmentation of nuclei. The first phase uses spectral, shape as well as the class membership to produce a nested hierarchical partition (hierarchy of segmentations). The second phase, selects the best hierarchical level based on an unsupervised criterion, and refines the obtained segmentation by classifying the individual regions using a Support Vector Machine (SVM) classifier followed by merging adjacent regions belonging to the same class. The effectiveness of the proposed approach for producing a better separation of nucleus regions and cytoplasm areas is demonstrated using both ground truth data, being manually segmented images by pathologist experts, and comparison with state-of-art methods.

Keywords: microscopic images, cell segmentation, watershed, SVM classification.

1 Introduction

Cervical cancer, currently associated with the Human Papilloma Virus as one of the major risk factors, affects thousands of women each year. The Papanicolaou test (known as Pap test) is used to detect pre-malignant and malignant changes in the cervix [1]. Cervical cancer can be mostly prevented by early detection of abnormal cells in smear tests.

Due to the cervix is wiped out with a swap, Pap test is classified as an invasive method. It is only used for screening purposes and not for diagnosis. These cells are examined under a microscope for abnormalities. Trained biologist are required to evaluate these tests. In underdeveloped countries, the death rate due to cervical cancer is significantly higher due to the lack of personnel trained in this field and repeated follow-up tests. As a result, women in developed countries have less than 0.1% chance of developing cervical cancer while their counterparts in underdeveloped countries have a 3-5% chance of developing cervical cancer.

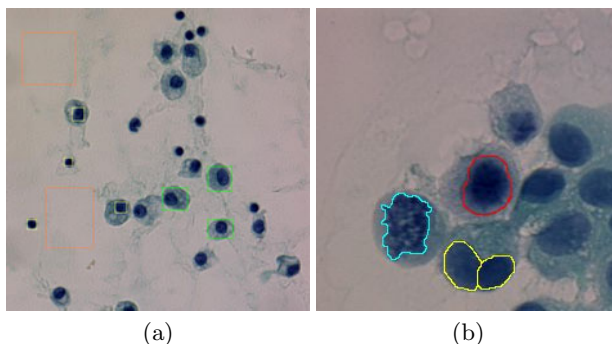


Fig. 1. Pap-smear cell images. (a) Dark blue parts (yellow rectangle) represents the nuclei. Pale blue part (green rectangle) are the cytoplasm. Magenta parts (orange rectangle) are the background. (b) Nucleus Variability.

As illustrated in Fig. 1, two classes of regions are considered: nucleus regions and other regions that include cytoplasm and background. The overall proportion of the nucleus pixels is approximately between 7% and 10%. Cells nucleus are blue (dark to pale) and cytoplasm are blue-green (Fig. 1a). Red blood corpuscles are coloured reddish. The spatial configuration and the colour of the cells is extremely variable (Fig. 1b). Isolated or touching cells as well as clustered or overlapping cells can be found.

The automated segmentation of cell nuclei in Pap smear images is one of the most interesting fields in cytological image analysis [2]. In the last years, cell nucleus segmentation has been extensively studied by several researchers, in [3], frequency domain features are used to detect abnormal cervical cell images. In [4], statistical geometric features, which are computed from several binary thresholded versions of texture images are used to classify among normal and abnormal cervical cells. Lezoray et al. [5] extract the nuclei of the cervical cells using a combination of a colour pixel classification scheme (k -means and Bayesian classification) with a colour watershed segmentation algorithm. In [6] a segmentation scheme and its performance is evaluated using Pap-smear samples in the presence of heavy additive noise.

Developing automated algorithms for segmenting nuclei continue to pose interesting challenges. Much of the difficulty arises from the inherent colour and shape variability. The goal of the present work is to develop automated and computationally efficient algorithms that improve upon previous methods using watershed approach. In this work, we propose a hybrid two-steps approach to cell segmentation in Pap smear images. The first phase consists of creating a nested hierarchy of partitions, which produces a hierarchical segmentation that uses the spectral, shape information as well as the class information. The most meaningful hierarchical level is then detected using a segmentation quality criterion. The second phase aims at identifying the nucleus and cytoplasm

areas by classifying the segments (regions) resulting from the first phase using multiple spectral and shape features, and further merging the neighboring regions belonging to the same class. The selection of individual regions is obtained using a SVM classifier, based on spectral and shape features.

The reminder of the paper is organized as follows. Section 2 describes the segmentation algorithm used to segment the images and produces a hierarchy of nested partitions. Section 3.1 proposes an unsupervised segmentation quality criterion to select a hierarchical level on which an SVM classification is applied in Section 3.2 to classify the segmented region in nucleus/non-nucleus and pruning the segmentation errors by merging adjacent segmented regions which may have been over-segmented. Section 4 presents and discusses the obtained results. Finally, conclusions are presented in Section 5.

2 Hierarchy of Partitions of Pap-Smear Cell Images

The *waterfall* algorithm [7] is used here for producing a nested hierarchy of partitions, $\mathcal{P}^h = \{r_1^h, r_2^h, \dots, r_{m_h}^h\}; h = 1, \dots, n$, which preserves the inclusion relationship $\mathcal{P}^h \supseteq \mathcal{P}^{h-1}$, implying that each atom of the set $\mathcal{P}^h$ is a disjoint union of atoms from the set $\mathcal{P}^{h-1}$. For successively creating hierarchical partitions, the waterfall algorithm removes from the current partition (hierarchical level) all the boundaries completely surrounded by higher boundaries. The staring partition, is obtained using watershed transform [8], being a morphological segmentation applied on the gradient magnitude of an image in order to guide the watershed lines to follow the crest lines and the real boundaries of the regions. In our implementation, we use the DiZenno gradient [9], which calculates the maximum rate of change in one pixel based in partial derivatives in *RGB* colour space.

For producing the nested hierarchy, in this work, we use the approach proposed in [10], where the saliency measure, $E(\tilde{r} = r_i \cup r_j | r_i, r_j)$, of a boundary between two neighboring segments r_i and r_j (being the cost of merging the regions r_i and r_j), is based on two energy functions used to characterize desired single-segment properties, and pair-wise segment properties [10]:

$$E(\tilde{r} = r_i \cup r_j | r_i, r_j) = E(\tilde{r}) + E(r_i, r_j) . \quad (1)$$

The single-segment properties, $E(\tilde{r})$, is the merged region property as defined in [10], it includes segment homogeneity (E_{hom}), segment convexity (E_{conv}), segment compactness (E_{comp}), and colour variances (E_{var_c}) within the segment:

$$E(\tilde{r}) = \frac{1}{E_{\text{hom}}(\tilde{r})} \cdot \sum_c E_{\text{var}_c}(\tilde{r}) \cdot (1 + |E_{\text{conv}}(\tilde{r})|)^{\text{sign}(E_{\text{conv}}(\tilde{r}))} \cdot (1 + |E_{\text{comp}}(\tilde{r})|)^{\text{sign}(E_{\text{comp}}(\tilde{r}))} . \quad (2)$$

The pair-wise property, $E(r_i, r_j)$, as defined in [10] includes the dynamics of the contour and the color difference between the neighboring regions.

In this work, considering the type of images we are dealing with, we propose the following merging criterion:

$$E(\tilde{r} = r_i \cup r_j \mid r_i, r_j) = \phi(c_i = c_j \mid r_i, r_j) \cdot (E(\tilde{r}) + E(r_i, r_j)) \quad . \quad (3)$$

where, $\phi(c_i = c_j \mid r_i, r_j)$ is a factor favoring the merging of regions with similar classes [11]; $E(r_i, r_j)$, the pair-wise region property, defined as:

$$E(r_i, r_j) = -\log \left(\sum_{k=1}^b \sqrt{P_{r_i}^{(k)} \cdot P_{r_j}^{(k)}} \right) \quad . \quad (4)$$

being the Bhattacharyya merging criterion proposed in [12], with the number of bins used $b = 32$.

Different from [11], the parameter $\phi(c_i = c_j \mid r_i, r_j)$, representing the potential of merging neighboring regions with similar class membership, is here defined as follows:

$$\phi(c_i = c_j \mid r_i, r_j) = \frac{1}{1 + \Pr(c_i = c_j \mid \mathbf{f}(r_i), \mathbf{f}(r_j))} \quad . \quad (5)$$

where $c_i, c_j \in \Omega = \{\omega_1 = \text{nucleus}, \omega_2 = \text{no-nucleus}\}$, are the classes for r_i and r_j , respectively, and $\Pr(c_i = c_j \mid \mathbf{f}(r_i), \mathbf{f}(r_j))$ is the probability that r_i and r_j belong to the same class, given the feature vectors $\mathbf{f}(r_i)$ and $\mathbf{f}(r_j)$. In our approach, $\Pr(c_i = c_j \mid \mathbf{f}(r_i), \mathbf{f}(r_j))$ is calculated using the method of Platt [13] from the output of a two classes SVM [14] trained using as feature vector, $\mathbf{f}(r) = [\mu(r(L)), \mu(r(a)), \mu(r(b))]^t$, consisting of the mean of the L , a , and b channels of the region r in the *Lab* colour space.

The parameters of the SVM classifier have been selected as follows. A linear kernel SVM and gaussian kernel SVMs (with different values for σ) were trained using a 10-fold cross-validation. A grid search method was used to select the best parameters of the SVM. The penalty parameter of the error C was tested in $C = \{2^i : i = -1..14, \infty\}$, as well as the parameter of the gaussian kernel σ in $\sigma = \{2^i : i = -3..4\}$. The best performance was obtained for $C = 1024$ and gaussian kernel SVM with $\sigma = 0.5$. Finally,

$$\Pr(c_i = c_j \mid \mathbf{f}(r_i), \mathbf{f}(r_j)) = p_1 p_2 + (1 - p_1)(1 - p_2) \quad . \quad (6)$$

with $p_1 = \Pr(c_i = \omega_1 \mid f)$ and $p_2 = \Pr(c_j = \omega_1 \mid f)$, are estimated using the method of Platt [13] which adjusts the output of a SVM by using a sigmoid function

$$\Pr(\text{class} = \omega_k \mid f) = \frac{1}{1 + \exp(Af + B)} \quad . \quad (7)$$

where f is the output of the SVM, and the parameters A and B are fitted using maximum likelihood estimation.

3 Pap-Smear Cell Images Segmentation and Classification

3.1 Segmentation Level Selection

As mentioned above, the output of the hierarchical segmentation is a set of partitions $\mathcal{P}^h = \{r_1^h, r_2^h, \dots, r_{m_h}^h\}; h = 1, \dots, n$. In order to select the best segmentation level, for further analysis, in this work we use the criterion of Borsotti *et al.* [15]:

$$\text{BOR}(\mathcal{P}^h) = 1 - \frac{\sqrt{m_h}}{10^4 \cdot \text{Card}(I)} \cdot \sum_{k=1}^{m_h} \left[\frac{E_k^2}{1 + \log(\text{Card}(r_k^h))} + \left(\frac{\chi(\text{Card}(r_k^h))}{\text{Card}(r_k^h)} \right)^2 \right] \quad (8)$$

where, $\text{Card}()$ is the size (area) of a region r_k^h or the image I ; $\chi(\text{Card}(r_k^h))$ is the number of regions having the same size (area) as region r_k^h ; and E_k is sum of the Euclidean distances between the RGB colour vector of the pixels of r_k and the colour vector attributed to the region r_k in the segmentation result. This criterion allows penalizing both over-segmentation (small regions) and under-segmentation (regions that have a large color error).

The best segmentation level is the one which produces the maximum value of the BOR criterion Eq (8).

3.2 SVM Region Classification and Merging

The Borsotti *et al.* [15] criterion Eq (8) is a good unsupervised segmentation quality measure, however most of the time the best value does not correspond to the best segmentation level according to the biologists criteria (Fig. 2(2) versus Fig. 2(6)). A suitable approach is to prune the segmentation, resulting from the hierarchical level selection criterion, by merging adjacent regions belonging to the same class. Indeed, as depicted in Fig. 2, the selected level shows a cell with 2 regions, after region-based classification and extra merging, the final segmentation/classification results has been refined.

Support vector machines (SVM) have been proven to be powerful and robust tools for tackling classification tasks [14]. Different from mostly used SVM pixel-based classification, we propose to apply SVM on region-based features and classify the segments of the selected level into nucleus and non-nucleus regions. A set of 116 region features were first calculated. In an attempt to optimize the dimensionality of the feature set, a subset of features was selected via stepwise discriminant analysis [16]. This method uses Wilks' λ statistic to iteratively determine which features are best able to separate the classes from one another in the feature space. Since it is not possible to identify a subset of features that are optimal for classification without training and testing classifiers for all combinations of the input features, optimization of Wilks' λ is a good choice.

Table 1, lists the identified nine (out of 116) features that were the most statistically significant in terms of their ability to separate the two considered classes, nucleus and no nucleus regions (cytoplasm and background).

Table 1. Selected features using stepwise discriminant analysis

F_1 . Mean of the green channel
F_2 . 0.1-trimmean of the blue channel
F_3 . Solidity
F_4 . Max value of the red channel
F_5 . Edge fraction of pixels along the edges
F_6 . Edge gradient intensity homogeneity
F_7 . Edge direction difference.
F_8 . Shape factor of the convex hull
F_9 . Region area

4 Results and Discussion

Fig. 3 illustrates the proposed approach in one of the tested images. The first row depicts some hierarchical levels along with their BOR criterion and number of regions. As it can be noticed, the hierarchical Level-1 is the best according to the BOR criterion. Moreover, the BOR criterion between the first three levels is almost identical. After SVM classification and the merging of neighboring regions belonging to the same class (second and third rows of Figure 3), the hierarchical Level-2 gives better segmentation results with respect to the BOR criterion, and the Vinet distance, V , [5]. This is also confirmed by classification results as shown in Table 2.

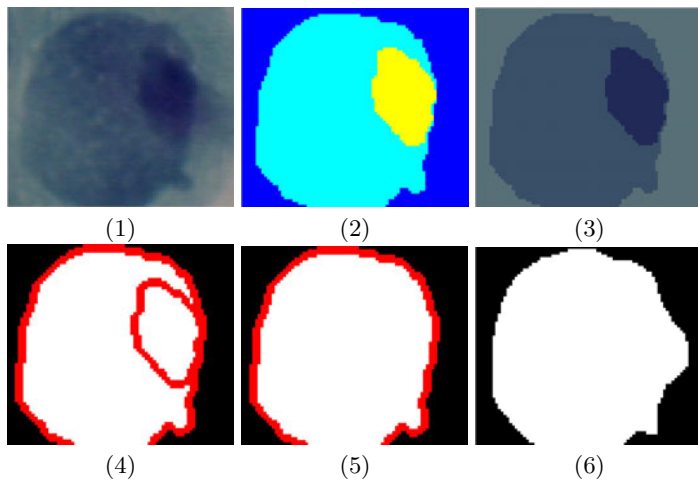


Fig. 2. Merging after classification. (1) Original image. (2) Hierarchical Level Selection Results: Labeled image. (3) Mosaic image. (4) Region Classification Results (white mean nucleus). (5) Merging of regions that belong to the same class. (6) Manually delineated nucleus.

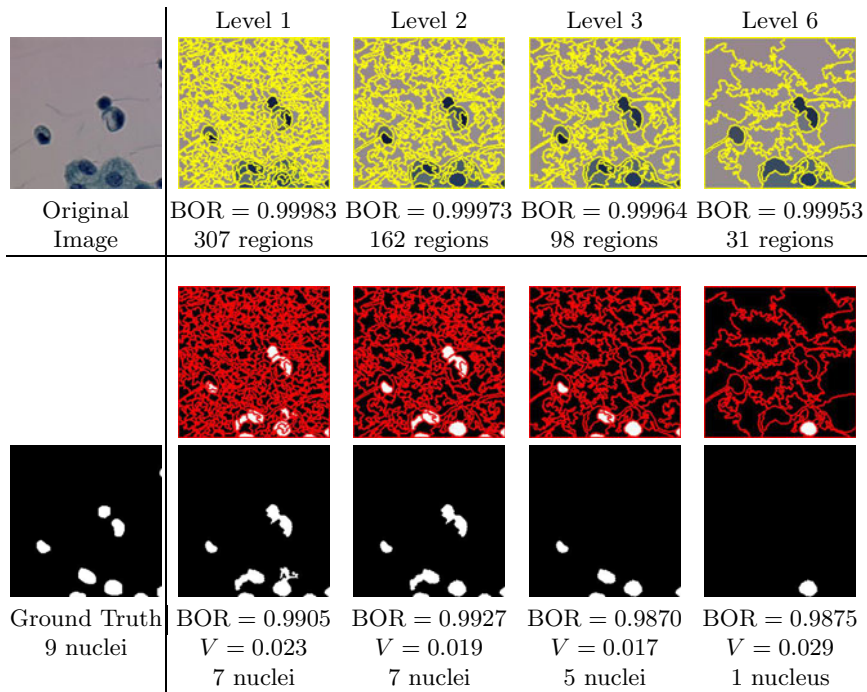


Fig. 3. Illustration of the approach. (Upper-Left) Original Image. (Bottom-Left) Ground Truth. (Upper-Right) Four Hierarchical Levels. (Bottom-Right) Results after classification and second merging.

Table 2. Confusion matrices - SVM classification of the regions shown in Fig. 3

Level		Nucleus	Non-Nucleus
1	Nucleus	290 (100.0 %)	0 (0.00 %)
	Non-Nucleus	4 (23.53 %)	13 (76.47 %)
2	Nucleus	151 (100.0 %)	0 (0.00 %)
	Non-Nucleus	1 (9.09 %)	10 (90.91 %)
3	Nucleus	91 (98.91 %)	1 (1.09 %)
	Non-Nucleus	0 (0.00 %)	6 (100.0 %)
6	Nucleus	29 (96.67 %)	1 (3.33 %)
	Non-Nucleus	0 (0.00 %)	1 (100.0 %)

The Vinet distance is a widely used measure to quantify the difference between two segmentations (one of them is frequently a ground truth). For an image I of N pixels and two segmentations A and B , with m and n regions, respectively. First, a label superposition table is computed: $T_{ij} = |A_i \cup B_j|$ with $0 \leq i \leq m$ and $0 \leq j \leq n$. The maximum of this matrix gives the two most similar regions extracted from A and B , respectively. The similarity criterion is defined by:

$C_0 = \max(T_{ij})$ with $0 \leq i \leq m$ and $0 \leq j \leq n$. The search of the second maximum (without taking into account the two last regions) gives the similarity criterion C_1 and so on to C_{k-1} , where $k = \min(m, n)$. The dissimilarity measure between the two segmentations A and B is given by:

$$D(A, B) = 1 - \frac{1}{N} \cdot \sum_{i=0}^{k-1} C_i . \quad (9)$$

The proposed approach was applied to twenty images that contains approximately 160 nuclei. The training of the SVM was made using the SVM-KM toolbox [17]. The evaluation of the proposed approach was made using a *leave-one-out* cross-validation using two different criteria:

- Segmentation quality: using the Vinet distance [5] according to a manually extracted ground truth.
- Classification quality: using the Accuracy and F-measures [18] according to a manually extracted ground truth.

Table 3. Overall Assesment

SVM Classifier	Vinet Measure	Accuracy	F-measure
Linear Kernel	0.0223	0.9733	0.9853
Gaussian Kernel $\sigma = 0.5$	0.0494	0.9109	0.9587
Gaussian Kernel $\sigma = 1$	0.0456	0.9235	0.9644
Gaussian Kernel $\sigma = 2$	0.0407	0.9448	0.9704
Gaussian Kernel $\sigma = 4$	0.0323	0.9592	0.9781
Gaussian Kernel $\sigma = 8$	0.0274	0.9668	0.9821
Linear Kernel $d = 2$	0.0375	0.9546	0.9750
Linear Kernel $d = 3$	0.0528	0.9552	0.9758
CCW	0.0455	0.9571	0.6445
GEE	0.0243	0.9784	0.9880
SVMP	0.0356	0.9743	0.8601

Table 3 summarizes the average of the Vinet measure, Accuracy, and F-measures, for all the testing images, versus SVM kernels. As it can be seen the best results are obtained using SVM classifier with a linear kernel. The Confusion matrix using a linear kernel is given in Table 4.

Table 4. Confusion Matrix for SVM classifier with a linear kernel

	Nucleus Non-Nucleus	
Nucleus	98.49 %	1.51 %
Non-Nucleus	9.16 %	90.84 %

To further assess our results, we give in the last rows of Table 3 the results obtained using three state of art methods, namely, the Cooperative colour watershed proposed in [5] (CCW), the Hierarchical Segmentation of [10] (GEE), and

a Pixel-based SVM classification (SVMP) [14]. As it can be seen from Table 3, the proposed approach produces good results.

5 Conclusions

In this work, we introduced a hybrid segmentation/classification approach which improves the automatic segmentation of nuclei for the purpose of the Papanicolaou test. First, a classification factor was introduced during the process of merging neighboring segments during the hierarchical segmentation process. Second we introduced a non supervised approach for the selection of the best hierarchical segmentation level. Finally, to prune most of the wrongly segmented cells and avoid over/under segmentation, we introduced a region-based SVM classifier able of improving the performance of the resulting segmentation. The SVM classifier was used to separate the two classes of regions: nucleus and no nucleus regions (cytoplasm and background) using an appropriate set of region features (morphometrics, edge-based, and convex hull-based). Our method is adapted to the segmentation of cellular objects. A *leave-one-out* cross-validation approach allowed proving that the proposed approach produces a segmentation closer to what is expected by human experts. In order to improve the segmentation results (separating cells) we will consider applying vector image restoration based on Partial Differential Equations (PDE) [19].

Acknowledgement

This work was partially supported by the Canadian International Development Agency Project Tier II-394-TT02-00 and by the Flemish VLIR-UOS Programme for Institutional University Co-operation (IUC).

References

1. Papanicolaou, G.: A new procedure for staining vaginal smears. *Science* 95, 438–439 (1942)
2. Pantanowitz, L., Hornish, M., Goulart, R.: The impact of digital imaging in the field of cytopathology. *Cytojournal* 6(1), 6–15 (2010)
3. Ricketts, I., Banda-Gamboa, H., Cairns, A., Hussein, K.: Automatic classification of cervical cells-using the frequency domain. In: *IEEE Colloquium on Applications of Image Processing in Mass Health Screening, IET*, p. 9 (2002)
4. Walker, R., Jackway, P.: Statistical geometric features extensions for cytological texture analysis. In: *Proceedings of 13th International Conference on Pattern Recognition*, vol. 2, pp. 790–794 (1996)
5. Lezoray, O., Cardot, H.: Cooperation of color pixel classification schemes and color watershed: a study for microscopic images. *IEEE transactions on Image Processing* 11, 783–789 (2002)
6. Bak, E., Najarian, K., Brockway, J.: Efficient segmentation framework of cell images in noise environments. In: *26th IEEE Annual International Conference on Engineering in Medicine and Biology Society (IEMBS 2004)*, vol. 1, pp. 1802–1805 (2005)

7. Beucher, S.: Watershed, hierarchical segmentation and waterfall algorithm. *Mathematical morphology and its applications to image processing*, 69–76 (1994)
8. Roerdink, J., Meijster, A.: The watershed transform: Definitions, algorithms and parallelization strategies. *Mathematical morphology* 187 (2000)
9. DiZenzo, S.: A note on the gradient of a multi-image. *Comput. Vision, Graphics. Image Proc.* 33(1), 116–125 (1986)
10. Geerinck, T., Sahli, H., Henderickx, D., Vanhamel, I., Enescu, V.: Modeling attention and perceptual grouping to salient objects. In: Paletta, L., Tsotsos, J.K. (eds.) *WAPCV 2008. Lecture Notes in Computer Science(LNAI)*, vol. 5395, pp. 166–182. Springer, Heidelberg (2009)
11. Lucchi, A., Smith, K., Achanta, R., Lepetit, V., Fua, P.: A Fully Automated Approach to Segmentation of Irregularly Shaped Cellular Structures in EM Images. In: *Medical Image Computing and Computer-Assisted Intervention–MICCAI 2010*, pp. 463–471 (2010)
12. Calderero, F., Marques, F.: General region merging approaches based on information theory statistical measures. In: *15th IEEE International Conference on Image Processing, ICIP 2008*, pp. 3016–3019 (2008)
13. Platt, J.C.: Probabilistic outputs for support vector machines and comparison to regularized likelihood methods. In: *Advances in large margin classifiers*, pp. 61–74 (1999)
14. Cristianini, N., Shawe-Taylor, J.: *Introduction to Support Vector Machines and other kernel-based learning methods*. Cambridge University Press, Cambridge (2000)
15. Borsotti, M., Campadelli, P., Schettini, R.: Quantitative evaluation of color image segmentation results. *Pattern Recognition Letters* 19(8), 741–747 (1998)
16. Jennrich, R., Sampson, P.: Stepwise discriminant analysis. In: *Mathematical methods for digital computers*, pp. 339–358 (1960)
17. Canu, S., Grandvalet, Y., Rakotomamonjy, A.: *SVM and Kernel Methods MATLAB Toolbox*. Perception de Syst émes et Information, INSA de Rouen, France (2003)
18. Joshi, M.V.: On evaluating performance of classifiers for rare classes. In: *Proceedings of the IEEE International Conference on Data Mining ICDM 2002*, p. 641. IEEE Computer Society, Washington (2002)
19. Vanhamel, I., Mihai, C., Sahli, H., Katartzis, A., Pratikakis, I.: Scale Selection for Compact Scale-Space Representation of Vector-Valued Images. *International Journal of Computer Vision* 84(2), 194–204 (2009)

Segmentation of Noisy Images Using the Rank M-Type L-Filter and the Fuzzy C-Means Clustering Algorithm

Dante Mújica-Vargas, Francisco J. Gallegos-Funes, and Rene Cruz-Santiago

Mechanical and Electrical Engineering Higher School

National Polytechnic Institute of Mexico

Av. IPN s/n, Edificio Z, acceso 3, 3^{er} piso; SEPI-Electronica, Col. Lindavista, 07738, México

D. F. México, Phone/Fax: (5255)57296000 ext. 54622

Abstract. In this paper we present an image processing scheme to segment noisy images based on a robust estimator in the filtering stage and the standard Fuzzy C-Means (FCM) clustering algorithm to segment the images. The main objective of paper is to evaluate the performance of the Rank M-type L-filter with different influence functions and to establish a reference base to include the filter in the objective function of FCM algorithm in a future work. The filter uses the Rank M-type (RM) estimator in the scheme of L-filter, to get more robustness in the presence of different types of noises and a combination of them. Tests were made on synthetic and real images subjected to three types of noise and the results are compared with six reference modified Fuzzy C-Means methods to segment noisy images.

Keywords: robust estimators, RM-estimator, L-filter, Fuzzy C-Means, segmentation, noise.

1 Introduction

Image segmentation is a key step forward image analysis and serves in a variety of applications, including pattern recognition, object detection, medical imaging, robot vision, military surveillance [1]. Image segmentation can be defined as the partition of an image into different meaningful regions with homogenous features using discontinuities or similarities of the image such as intensity, color, texture, and so on [2]. Numerous segmentation techniques have been developed and reported in the literature.

Fuzzy clustering as a soft segmentation method has been widely studied and successfully applied to image segmentation. Among the fuzzy clustering methods, Fuzzy C-Means (FCM) [3] algorithm is the most popular method because it is simple, easy to program, and can retain much more information than hard methods. Although fuzzy clustering methods work well on most noise-free images, they have a serious limitation: they do not incorporate any information about spatial context, which cause them sensitivity to the noise or outliers data. Then it is necessary to modify the objective function to incorporate local information of the image to get better results.

Following the image processing chain [4] to get a good segmentation stage, is necessary to have a good pre-filtering stage. The filter must be robust at the presence of different levels and types of noise, and in the extreme case, when the image is not noisy, the filtering stage does not distort the image in any way. Taking into consideration the above, in this paper we use a RML-estimator [5] to perform image filtering under the conditions mentioned the presence or absence of noise.

The outline of this paper is as follows. Section 2 presents information about the M, R and L estimators, and how to merge them. Section 3 gives a recall of the standard Fuzzy C-Means clustering algorithm. Experimental results are compared with some reference methods are shown in section 4. Finally, some conclusions are drawn in section 5.

2 Proposed Method

To segment noisy images we use the image processing chain shown in Figure 1. In this one the segmentation is the central point and crucial for next stages, and therefore it is necessary to have good results at this stage, but the segmentation has a total dependency of the early filter stage. The propose method is given by the RML-estimator in the filtering stage and then to segment by the standard Fuzzy C-Means clustering algorithm.

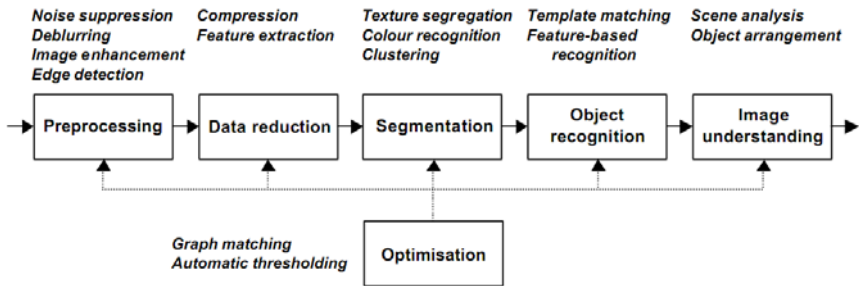


Fig. 1. The image processing chain containing the five different tasks: preprocessing, data reduction, segmentation, object recognition and image understanding

2.1 RM-Estimator

The R-estimators form a class of nonparametric robust estimators based on rank calculations [6,7,8,9]. They are known to be robust estimators and are used in the signal processing area like R-filters. The median estimator (median filter) is the best choice when any *a priori* information about data X_i distribution shape and its moments is unavailable [7],

$$\hat{\theta}_{med} = \{x_{(n+1/2)}, \quad \text{for odd } n\} \quad (1)$$

where, $x_{(i)}$ is the element with i rank, n is the size of the sample, and $1 \leq i \leq n$.

M-estimators are a generalization of maximum likelihood estimation (MLE) and were proposed by Peter Hubert [6,7,8,9]. Their definition is given by a robust loss function $\rho(x) = \ln(f(x))$, connected with the probability density function $f(x)$ for the sample data $x_i, i = 1, \dots, n$. The objective of M-estimators is to find an estimation $\hat{\theta}$ of θ such that,

$$\hat{\theta} = \arg \min_{\theta \in \Theta} \left(\sum_{i=1}^n \rho(x_i - \theta) \right) \quad (2)$$

The estimation of the localization parameter θ can be found by calculating the partial derivative of ρ (with respect to θ) introducing the function,

$$\psi(x, \theta) = \frac{\partial}{\partial \theta} \rho(x, \theta)$$

$$\sum_{i=1}^n \psi(x_i - \theta) = 0 \quad (3)$$

The robust M-estimator solution for θ is determined by imposing certain restrictions on the influence function $\psi(x)$ (see Table 1) or the samples $(x_i - \theta)$, called censorization or trimming. The standard technique used to calculate the M estimation is based on the iterative Newton method, but can be simplified by a single-step algorithm [9] to calculate the lowered M-estimate of the average θ value,

$$\theta_M = \frac{\sum_{i=1}^n x_i \tilde{\psi}(x_i - \text{med}\{x_n\})}{\sum_{i=1}^n 1_{[-r, r]} \tilde{\psi}'(x_i - \text{med}\{x_n\})} \quad (4)$$

where, $\tilde{\psi}$ is the normalized influence function $\psi: \psi(x) = x \tilde{\psi}(x)$. Is evident that (4) represents the arithmetic average of $\sum_{i=1}^n \psi(x_i - \text{med}\{x_n\}) = 0$, which is evaluated on the interval $[-r, r]$, r is the parameter connected with restrictions on the range of ψ . The simplest restriction on the range of ψ is the limit of the Huber's estimator $\tilde{\psi}_r(X) = \min(r, \max(X, r)) = [X]_{-r}^r$ [9].

Table 1. Influence functions used in the proposed filter

Influence function	Formulae	
simple cut	$\psi_{cut(r)}(x) = \begin{cases} x, & x \leq r \\ 0, & \text{otherwise} \end{cases}$	
Andrew's sine	$\psi_{sin(r)}(x) = \begin{cases} \sin(x/r), & x \leq r\pi \\ 0, & \text{otherwise} \end{cases}$	
Tukey's biweight	$\psi_{bi(r)}(x) = \begin{cases} \left[1 - \left(\frac{x}{r}\right)^2\right]^2, & x \leq r\pi \\ 0, & \text{otherwise} \end{cases}$	
Hampel's three part redescending	$\psi_{\alpha, \beta(r)}(x) = \begin{cases} x, & 0 \leq x < \alpha \\ \alpha \operatorname{sgn}(x), & \alpha \leq x < \beta \\ \alpha \frac{r - x }{r - \beta} \operatorname{sgn}(x), & \beta \leq x < r \\ 0, & r \leq x \end{cases}$	

The proposal to enhancement the robust properties of M-estimator (4) by using the R-estimator (1) consists of the application procedure similar to the median average instead of arithmetic one [9],

$$\theta_{medM} = med\{x_i\psi(x_i - med\{x_n\})\} \quad (5)$$

The properties of the RM-estimator (5) are increased by fusing the ability of the R-estimator to suppress the impulsive noise and the use of the different influence functions of the M-estimator to provide more robustness. Thus, it is expected that the combined RM-estimator can be better than the originals R-estimator and M-estimator.

2.2 RML- Estimator

We propose to use the RM-estimators into the linear combinations of order statistics defined by the L-filter. The proposed RM L (Rank M-type L) -filter employs the idea of RM-KNN algorithm [9]. The following representation of L-filter is often used [6],

$$\theta_L = \sum_{i=1}^n a_i x_{(i)} \quad \text{with} \quad a_i = \int_{\frac{i-1}{n}}^{\frac{i}{n}} h(\lambda) d\lambda \bigg/ \int_0^1 h(\lambda) d\lambda \quad (6)$$

where, $x_{(i)}$, $i = 1, \dots, n$ are the data sample and a_i , $i = 1, \dots, n$ are the weighted coefficients of filter, $h(\lambda)$ is the noise probability distribution function defined in $[0,1] \rightarrow R$, which satisfies $\int_0^1 h(\lambda) d\lambda \neq 0$, then, the L-filters satisfy the equation $\sum_{i=1}^n a_i = 1$ [7]. Using this method the weighted coefficients can be computed for different distribution functions (Exponential, Laplacian, Uniform, etc.) and window sizes, we used a 3x3 window because we obtained the best results and more detail preservation than 5x5 or 7x7 windows.

To introduce the RM-estimator in the scheme of L-filter, we should to present the ordered data sample of L-filter as function of an influence function. For this reason, the L-filter (6) is writing as [10,11]:

$$\theta_L = \sum_{i=1}^n a_i \psi(x_i) x_i \quad (7)$$

$$\psi(x_i) = \begin{cases} 1 & i \leq (2l+1)^2 \\ 0 & \text{otherwise} \end{cases} \quad (8)$$

where, $\psi(x)$ is the influence function used in the L-filter, $\psi(x_i)x_i$ is the ordered data sample according with the eq. (6), and $(2l+1)^2$ is the filtering window size. Then, the new filter can be obtained by merging the L-filter (7) and the RM-estimator (5). The Median M-type L (MML) -filter can be expressed by [10,11]:

$$\theta_{MML} = \frac{med\{a_i[x_i\psi(x_i - med\{x_n\})]\}}{a_{med}} \quad (9)$$

where, θ_{MML} is the output MML- filter, $x_i\psi(x_i - med\{x_n\})$ are the selected pixels in accordance with the influence function in the sliding filter window, and a_{med} is the median of coefficients a_i used as scale constant.

To improve the properties of impulsive noise suppression of the proposed filter we introduce an impulse detector, this one chooses that pixel is or not filter. The impulsive detector used is defined as [12]:

$$[(rank(x_{ij}) \leq s) \vee (rank(x_{ij}) \geq N - s)] \wedge (|x_{ij} - med\{x_n\}| \geq U_2) \quad (10)$$

where, x_{ij} is the central pixel in the filtering window, $s > 0$ y $U_2 \geq 0$ are thresholds, N is the length of the data and $med\{x_n\}$ is the median of pixels into the filtering window.

3 Classic Fuzzy C-Means Clustering

Fuzzy C-Means is a method for data classification, where each data belongs to a cluster to some degree, which is specified by a membership value [3]. This algorithm performs the iteration of two indispensable conditions to minimize the following objective function,

$$J_f(X; U_f, C) = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m d_{ij}^2 \quad \text{subject to} \quad \sum_{i=1}^N u_{ij} > 0 \quad \forall_j \quad \text{and} \quad \sum_{j=1}^c u_{ij} = 1 \quad \forall_i \quad (11)$$

where, $X = \{x_i | i = 1, \dots, N\}$ denotes the set of N feature vectors, c is the number of classes, $m \in [1, \infty)$ is a weighting exponent called the fuzzifier, d_{ij}^2 is the distance from feature vector x_i to the center of the class c_j and $C = (c_1, \dots, c_j)$ is a vector with all center classes. $U = [u_{ij}]$ is a $N \times c$ matrix denoting the constrained fuzzy c-partition. The value of u_{ij} denotes the degree of membership of x_i to the class c_j . Taking in account both constrains, the membership matrix and the cluster prototypes can be calculated with the following equations [13],

$$u_{ij} = \frac{1}{\sum_{k=1}^c \left(\frac{d_{ij}^2}{d_{ik}^2} \right)^{\frac{1}{m-1}}} \quad (12)$$

$$c_j = \frac{\sum_{i=1}^N u_{ij}^m x_i}{\sum_{i=1}^N u_{ij}^m} \quad (13)$$

4 Experimental Results

The performance of the proposed method was tested on synthetic and real images. In both cases the quantitative results were compared with FCM_S1, FCM_S2, EnFCM, FGFCMS1, FGFCM_S2 and the FGFCM algorithms taken from [14]. The comparison is done by the optimal segmentation accuracy (SA), where SA is defined as the number of correctly classified pixels divided by sum of the total number of pixels [14].

4.1 Results on a Synthetic Image

The algorithms were applied on the synthetic image as shown in Figure 2(a) (128x128 pixels, two classes with two gray levels taken as 0 and 90) corrupted by different levels of Gaussian and Salt & Pepper noise and a mixed noise of Gaussian white noise $N(0, 100)$ and unit dispersion, zero centered symmetric α -stable ($S\alpha S$) noise; for all algorithms $c=2$, according to [14] $\lambda_g=6$ and $\alpha S=3.8$, $r=5$ for all RML_FCM algorithms, $\alpha=0.16r$, $\beta=0.8r$ for Hampel influence function, $s=4$ and $U_2=5$ for the impulse detector. Table 2 and 3 show the SAs for the comparative and proposed algorithms, respectively, on the synthetic images and Figure 2 depicts the visual results.

Table 2. SA % of six reference algorithms on synthetic image

Algorithm	Gaussian 3%	Gaussian 5%	Gaussian 8%	S&P 5%	S&P 10%	S&P 15%	mixed $\alpha=0.3$	mixed $\alpha=0.5$	mixed $\alpha=0.7$
FCM_S1	99.14	96.42	92.32	98.69	97.14	94.78	93.80	98.68	99.59
FCM_S2	98.78	96.12	92.23	98.77	97.54	95.98	97.25	99.27	99.79
EnFCM	99.50	97.65	94.62	98.05	94.77	94.94	95.34	99.09	99.69
FGFCM_S1	99.57	98.20	95.41	99.07	96.47	92.40	95.82	99.44	99.83
FGFCM_S2	99.13	96.82	93.12	99.99	99.98	99.84	99.65	99.97	100.00
FGCM	99.51	98.10	95.10	99.91	99.47	98.36	97.95	99.84	99.96

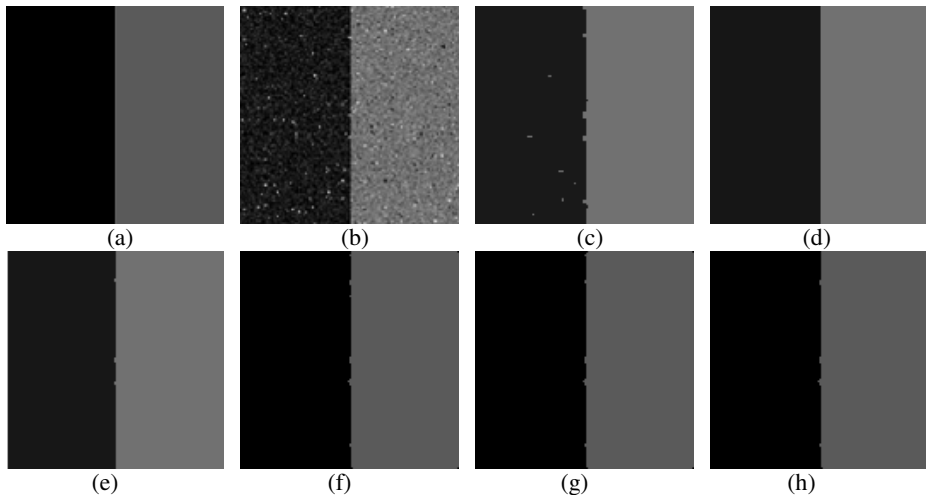


Fig. 2. Segmentation results in a synthetic image. (a) Original image. (b) Noisy image. (c) FGFCM_S1. (d) FGFCM_S2. (e) FGFCM. (f) RML_FCM H,U. (g) RML_FCM H,E. (h) RML_FCM H,L., where Hampel's three part redescending (H), Uniform (U), Exponential (E), Laplacian (L).

Table 3. SA % of RML_FCM algorithms on synthetic image

Influence function	RML distribution filter	Gaussian 3%	Gaussian 5%	Gaussian 8%	S&P 5%	S&P 10%	S&P 15%	mixed $\alpha=0.3$	mixed $\alpha=0.5$	mixed $\alpha=0.7$
Simple cut	Uniform	99.95	99.94	99.92	99.97	99.95	99.74	99.93	99.86	99.82
	Exponential	99.95	99.94	99.92	99.97	99.95	99.74	99.94	99.86	99.82
	Laplacian	99.95	99.94	99.92	99.97	99.95	99.74	99.93	99.89	99.84
Andrew's sine	Uniform	99.95	99.95	99.94	99.95	99.94	99.92	99.90	99.85	99.80
	Exponential	99.95	99.95	99.94	99.95	99.94	99.92	99.92	99.88	99.83
	Laplacian	99.95	99.95	99.94	99.95	99.94	99.92	99.89	99.86	99.81
Tukey's biweight	Uniform	99.95	99.93	99.92	99.93	99.88	99.86	99.89	99.84	99.78
	Exponential	99.95	99.93	99.92	99.93	99.88	99.86	99.88	99.84	99.74
	Laplacian	99.95	99.93	99.92	99.93	99.88	99.86	99.91	99.86	99.79
Hampel's three part redesding	Uniform	99.95	99.93	99.93	99.97	99.79	99.70	99.92	99.85	99.79
	Exponential	99.95	99.93	99.93	99.97	99.79	99.70	99.92	99.88	99.81
	Laplacian	99.95	99.93	99.93	99.97	99.79	99.70	99.91	99.86	99.80

4.2 Results on a Real Image

The robustness on a real image was tested using the *eight* real image corrupted by a mixed noise. The original image (Figure 3(a), 308x242 pixels) was corrupted simultaneously by Gaussian white noise $N(0,180)$ and unit dispersion, zero centered symmetric $\alpha(\alpha=0.9)$ -stable (*S α S*) noise. For all algorithms $c=3$, according to [14] $\lambda_g=2$ and $\alpha=8$ for reference algorithms, $r=5$ for all RML_FCM algorithms, $\alpha=0.16r$, $\beta=0.8r$ for Hampel influence function, $s=4$ and $U_2=5$ for the impulse detector. Table 4 gives the SAs for all algorithms on the *eight* image and Figure 3 presents the visual results.

Table 4. SA % of reference and RML_FCM algorithms on a real image

Influence function	RML distribution filter	Algorithm	SA %
-	-	FCM_S1	88.91
-	-	FCM_S2	88.64
-	-	EnFCM	82.18
-	-	FGFCM_S1	82.18
-	-	FGFCM_S2	89.11
-	-	FGFCM	91.87
Simple cut	Uniform	RML_FCM	90.52
	Exponential	RML_FCM	86.28
	Laplacian	RML_FCM	90.21
Andrew's sine	Uniform	RML_FCM	89.92
	Exponential	RML_FCM	88.86
	Laplacian	RML_FCM	89.67
Tukey biweight	Uniform	RML_FCM	90.47
	Exponential	RML_FCM	90.09
	Laplacian	RML_FCM	89.96
Hampel	Uniform	RML_FCM	89.51
	Exponential	RML_FCM	89.58
	Laplacian	RML_FCM	89.10

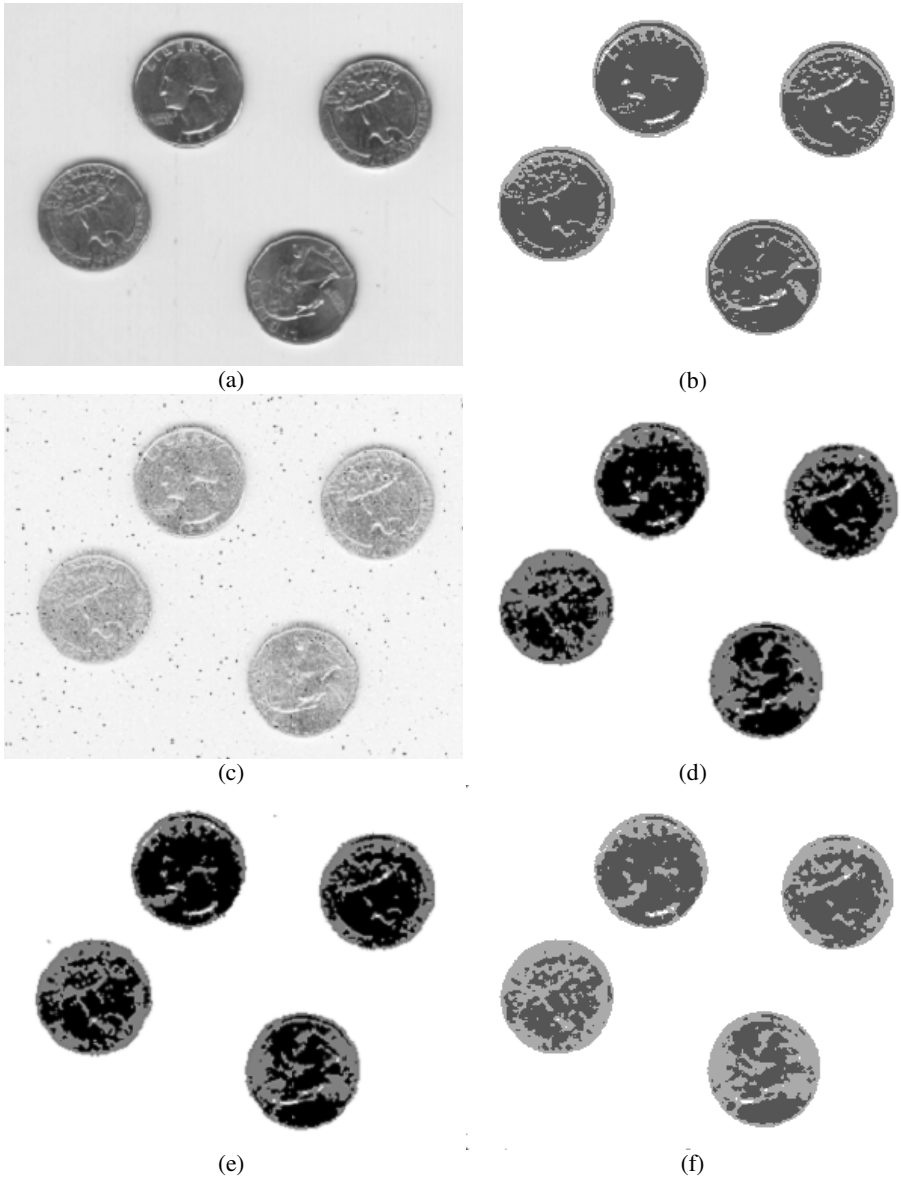


Fig. 3. Segmentation results on a real image. (a) Original image. (b) Original image segmentation. (c) Noisy image. (d) FGFCM_S2. (e) FGFCM. (f) RML_FCM H,U. (g) RML_FCM H,E. (h) RML_FCM H,L., where Hampel's three part redescending (H), Uniform (U), Exponential (E), Laplacian (L).

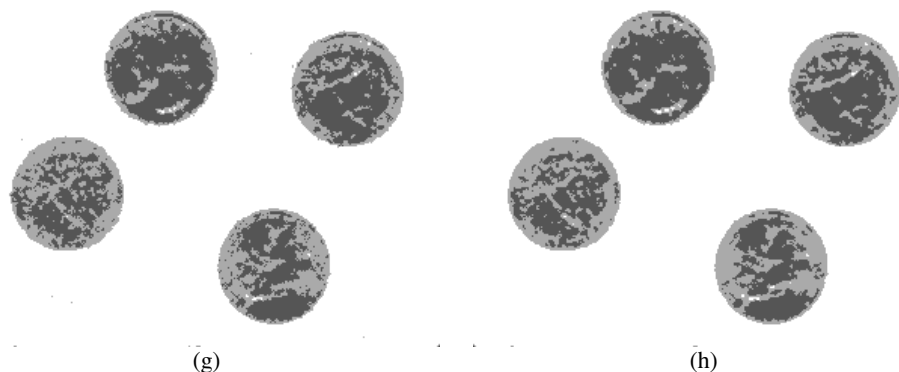


Fig. 3. *(continued)*

5 Discussion of Results

In tests in a synthetic image, one can see that the performance of the procedure RML_FCM presents better performance in the case of images corrupted by only one type of noise, but when the image has mixed noise the ability of method is comparable with the different reference algorithms. We must stress that although these algorithms include local information is the FGFCM algorithm the most robust in the presence of noise or outliers in the image. One can see that on a real image tests, the RML_FCM presents a SA% higher than the other algorithms. In the case of algorithms RML_FCM in all variations their performance is even close to FGFCM algorithm.

6 Conclusions

This paper presented the robust RM L-filters designed with different influence functions. These filters were the base of the FCM to segment noisy images. The performance of proposed RML_FCM is better than the comparative methods. To improve the properties of the FCM algorithm to segment free or noisy images as future work, the RM L-filter will be included in the function cost of the FCM algorithm to modify it. Besides, the segmentation will be extended on color images.

Acknowledgments

This work is supported by National Polytechnic Institute of Mexico and Conacyt.

References

1. Kim, J., Fisher, J.W., Yezzi, A., Cetin, M., Willsky, A.S.: A nonparametric statistical method for image segmentation using information theory and curve evolution. *IEEE Transactions on Image Processing*, 1486–1502 (2005)
2. Dong, G., Xie, M.: Color clustering and learning for image segmentation based on neural networks. *IEEE Transactions on Neural Networks* 16(4), 925–936 (2005)
3. Bezdek, J.C.: *Pattern Recognition with Fuzzy Objective Function Algorithms*. Plenum Press, New York (1981)
4. Egmont-Petersen, M., de Ridder, D., Handels, H.: Image processing with neural networks – a review. *Institute of Information and Computing, Utrecht University, Utrecht* (2002)
5. Gallegos-Funes, F.J., Linares, R., Ponomaryov, V., Cruz-Santiago, R.: Real-Time image processing using the Rank M-type L-filter. *Cientific* 11, 189–198 (2007)
6. Pitas, A.N.V.: *Nonlinear Digital Filters*. Kluwer Academic Publishers, Boston (1990)
7. Astola, J., Kousmanen, P.: *Fundamentals of Nonlinear Digital Filtering*. CRC Press, Boca Raton (1997)
8. Hampel, F.R., Ronchetti, E.M., Rouseew, P.J., Stahel, W.A.: *Robust Statistics*. In: *The approach based on influence function*. Wiley, NY (1986)
9. Gallegos-Funes, F.J., Ponomaryov, V.: Real-time image filtering scheme based on robust estimators in presence of noise impulsive. *Real Time Imaging* 8(2), 78–90 (2004)
10. Gallegos-Funes, F.J., Varela-Benitez, J.L., Ponomaryov, V.: Real-time image processing based on robust linear combinations of order statistics. In: *Proc. SPIE Real-Time Image Processing*, vol. 6063, pp. 177–187 (2006)
11. Varela-Benitez, J.L., Gallegos-Funes, F.J., Ponomaryov, V.: RML-filters for real time imaging. In: *Proc. IEEE 15th International Conference on Computing, CIC 2006*, pp. 43–48 (2006)
12. Aizenberg, L., Astola, J., Breging, T., Butakoff, C., Egiazarian, K., Paily, D.: Detectors of the impulsive noise and new effective filters for the impulse noise reduction. In: *Proc. SPIE Image Processing, Algorithms and Systems II*, vol. 5014, pp. 410–428 (2003)
13. Theodoridis, S., Koutroumbas, K.: *Pattern Recognition*, 4th edn. Elsevier, Amsterdam (2009)
14. Cai, W.L., Chen, S.C., Zhang, D.Q.: Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. *Pattern Recognition* 40, 383–825 (2007)

Design of Correlation Filters for Pattern Recognition Using a Noisy Training Image

Pablo M. Aguilar-González and Vitaly Kober

Department of Computer Science, Centro de Investigación Científica y de Educación
Superior de Ensenada,
Carretera Ensenada-Tijuana No. 3918, Zona Playitas, C.P. 22860, Ensenada, B.C.,
México

{paguilar, vkober}@cicese.mx

<http://www.cicese.edu.mx/>

Abstract. Correlation filters for object detection and location estimation are commonly designed assuming the shape and graylevel structure of the object of interest are explicitly available. In this work we propose the design of correlation filters when the appearance of the target is given in a single training image. The target is assumed to be embedded in a cluttered background and the image is assumed to be corrupted by additive sensor noise. The designed filters are used to detect the target in an input scene modeled by the nonoverlapping signal model. An optimal correlation filter, with respect to the peak-to-output energy ratio criterion, is proposed for object detection and location estimation. We also present estimation techniques for the required parameters. Computer simulation results obtained with the proposed filters are presented and compared with those of common correlation filters.

Keywords: correlation filters, pattern recognition.

1 Introduction

Since the introduction of the matched filter [1], correlation filters have been extensively used for pattern recognition [2-15]. Two tasks of interest in pattern recognition are detection of a target and the estimation of its location in an observed scene. With the help of correlation filters such tasks can be solved in two steps: detection is carried out by locating the highest peak in the filter output; then, the coordinates of the peak are taken as estimations of the position of the target in the observed scene [2].

The performance of correlation filters can be evaluated by quantitative performance criteria such as signal-to-noise ratio (SNR), peak sharpness, discrimination capability (DC), and probability of false alarms [3,4]. Location accuracy can be described in terms of the variance of location errors [5,6]. Correlation filters are designed by means of analytical optimization of one or more of these criteria. In order to perform such optimization, a mathematical model of the scene is chosen. The additive signal model is used when an input scene contains

a target distorted by additive noise. Optimizing the SNR criterion for the this model leads to the matched filter (MF) [1], while minimizing the probability of false alarms yields the optimal filter (OF) [4]. The nonoverlapping signal model is used when an opaque target is placed over a background that is spatially disjoint. Several filters have been derived for this scene model [6,7,8,9]. Maximizing the ratio of the square of the expected value of the correlation peak to the average output variance leads to the generalized matched filter [7]. Maximizing the peak-to-output energy ratio (POE) yields the generalized optimum filter [7] (GOF).

Because correlation filters are designed using the expected appearance of the target, their performance degrades rapidly if the target appears distorted in the scene. Distortions can be caused by changes of scale, rotation or perspective; blurring or defocusing, or incomplete information about the appearance of the target. Several correlation filters were proposed that take into account linear degradations of the input scene and the target [10]. Composite filters have been used to consider geometric distortions [11,12,13]. However, the design of these filters is done assuming that a knowledge of the target shape is explicitly known. In practical situations, the target may be given in a noisy reference image with a cluttered background. Recently [14,15], a signal model was introduced that accounts for additive noise in the image used for filter design. In this paper, we extend that work to account for the presence of a nonoverlapping background in a training image that is corrupted by additive noise. We derive a correlation filter optimized with respect to the POE criterion. The performance of this filter is compared to that of classical correlation filters for the nonoverlapping signal model.

2 Design of Filters

The nonoverlapping signal model is used for the reference image and the input scene. We use one-dimensional notation for simplicity. Integrals are taken between infinite limits. Throughout this section we use the same notation for a random process and its realization.

Formally, the input scene and the reference image are given, respectively, by

$$s(x) = t(x - x_s) + b_s(x) \bar{w}(x - x_s) + n_s(x) , \quad (1)$$

$$r(x) = t(x - x_r) + b_r(x) \bar{w}(x - x_r) + n_r(x) , \quad (2)$$

where $t(x)$ is the target, located at unknown coordinates x_s and x_r in the input scene $s(x)$ and in the reference image $r(x)$, respectively; $b_s(x)$ and $b_r(x)$ are the disjoint backgrounds, and $n_r(x)$ and $n_s(x)$ are the additive noise signals due to sensor noise. $\bar{w}(x)$ is the inverse support region for the target, that is, it takes a value of unity outside the target area and a value of zero inside. We make the following assumptions:

- The nonoverlapping backgrounds, $b_s(x)$ and $b_r(x)$, are treated as the realization of stationary random processes that have mean values μ_s and μ_r , respectively, and have power spectral densities $B_s^0(\omega)$ and $B_r^0(\omega)$, respectively.
- The additive noise processes $n_s(x)$ and $n_r(x)$ are assumed to be stationary random processes with mean zero and spectral densities $N_s(\omega)$ and $N_r(\omega)$, respectively.
- All random processes and random variables are treated as statistically independent.
- $s(x)$ and $r(x)$ are real-valued images with Fourier transforms $S(\omega)$ and $R(\omega)$, respectively.

The goal of the filter design process is to obtain a filter frequency response $H(\omega)$ of the form

$$H(\omega) = A(\omega) R^*(\omega), \quad (3)$$

where $A(\omega)$ is a deterministic function and $*$ denotes complex conjugate. Since the obtained filter frequency response contains non deterministic components, the filter expression represents a bank of transfer functions. A specific realization of the filter is fixed by the realization of the noises processes $b_r(x)$ and $n_r(x)$.

In (2), the location of the target in the reference image x_r , is unknown and not necessarily located at the origin. Therefore, the correlation peak is expected to be present at the coordinate $x_0 = x_s - x_r$. If x_r is close to 0, the location estimation of the target in the input scene will be in the close vicinity of its true location in the input scene. Even if the exact location of the target cannot be precisely determined, the relative position is useful for applications such as tracking [16], where the goal is to determine the relative movement of the target.

We derive the modified generalized optimum filter for the *nonoverlapping-nonoverlapping* model (GOF_{NN}) by maximizing the POE criterion, formally defined as

$$\text{POE} = |\text{E}\{y(x_0)\}|^2 / \text{E}\{\overline{|y(x)|^2}\}, \quad (4)$$

where $\text{E}\{\cdot\}$ denotes statistical averaging and the over-bar denotes spatial averaging, i.e. $\overline{y(x)} = (1/L) \int y(x) dx$, L is the spatial extent of the signal $y(x)$. The expected value of the correlation peak in the filter output plane is

$$\text{E}\{y(x_0)\} = (2\pi L)^{-1} \int A(\omega) \text{E}\left\{(R(\omega) e^{i\omega x_r})^* S(\omega) e^{i\omega x_s}\right\} d\omega. \quad (5)$$

The denominator of the POE represents the average energy in the correlation plane. It can be calculated as

$$\text{E}\{\overline{|y(x)|^2}\} = (2\pi L)^{-1} \int |A(\omega)|^2 \text{E}\{|R^*(\omega) S(\omega)|^2\} dx. \quad (6)$$

Using (5) and (6) in (4) we get

$$\text{POE} = \frac{L \left| \int A(\omega) \text{E}\left\{(R(\omega) e^{i\omega x_r})^* S(\omega) e^{i\omega x_s}\right\} d\omega \right|^2}{2\pi \int |A(\omega)|^2 \text{E}\{|R^*(\omega) S(\omega)|^2\} dx}. \quad (7)$$

Applying the Cauchy-Schwartz inequality to (7) and substituting the optimum value for $A(\omega)$ into (3), we obtain the following frequency response for the GOF_{NN} :

$$\text{GOF}_{\text{NN}}^*(\omega) = \frac{\text{E} \{ R(\omega) e^{i\omega x_r} \}^* \text{E} \{ S(\omega) e^{i\omega x_s} \} R(\omega)}{\text{E} \{ |R(\omega)|^2 \} \text{E} \{ |S(\omega)|^2 \}}. \quad (8)$$

The expected value of the power spectra of the input scene and reference image can be calculated as follows:

$$\text{E} \{ |S(\omega)|^2 \} = |T(\omega) + \mu_s \bar{W}(\omega)|^2 + \frac{1}{2\pi} B_s^0(\omega) \bullet |\bar{W}(\omega)|^2 + N_s(\omega), \quad (9)$$

$$\text{E} \{ |R(\omega)|^2 \} = |T(\omega) + \mu_r \bar{W}(\omega)|^2 + \frac{1}{2\pi} B_r^0(\omega) \bullet |\bar{W}(\omega)|^2 + N_r(\omega), \quad (10)$$

where $\bullet$ denotes the convolution operation and $\bar{W}(\omega)$ is the Fourier transform of the inverse support function. It can be seen that the obtained filter requires knowledge of the Fourier transform of the target and its support function. However, in the problem model we assume that this information is not available. Therefore, estimations need to be designed from the available information. A smoothing Wiener filter [17] can be used to partially suppress the background and attenuate the effects of noise in the reference image. After filtering, we can apply a threshold to the resulting image and obtain an approximate support function as follows:

$$\hat{w}(x) = \begin{cases} 1 & \mu_r > \mu_t \text{ and } \tilde{r}(x) \geq \tau(t, b_r, n_r) \\ 1 & \mu_r < \mu_t \text{ and } \tilde{r}(x) \leq \tau(t, b_r, n_r) \\ 0 & \text{otherwise} \end{cases}, \quad (11)$$

where $\tilde{r}(x)$ denotes the reference image after Wiener filtering, $\hat{w}(x)$ denotes the estimation of the inverse support function and $\tau(t, b_r, n_r)$ is the optimum threshold for separating the distributions of $t(x) + n_r(x)$ and $b_r(x) + n_r(x)$ after filtering. We make the simplifying assumption that the noise processes are approximately normal after filtering. Therefore, we can estimate their statistics, using the spectral densities we assumed known, and use them calculate the optimum threshold. When the statistics of the target are known, they can be used to improve the threshold calculation. If the statistics of the target are unknown, the optimum threshold can be determined with respect to the statistics of the background and additive noise.

Once an estimation of the inverse support function is obtained, we can estimate the expected value of the input scene as

$$T(\omega) + \mu_s \bar{W}(\omega) \approx \tilde{R}(\omega) + (\mu_s - \mu_r) \hat{W}(\omega), \quad (12)$$

where $\tilde{R}(\omega)$ is the Fourier transform of the reference image after Wiener filtering. This estimation is then used to design the frequency response of the GOF_{NN} .

It is worth noting that when there is no noise present in the reference image, the GOF_{NN} is equal to the GOF.

3 Computer Simulations

In this section we present computer simulation results. The performance of the proposed filters is evaluated in terms of discrimination capability (DC) and location errors (LE). The DC is formally defined [4] as the ability of a filter to distinguish a target from other objects in the scene. The DC is formally defined as

$$DC = 1 - \frac{|C_B|^2}{|C_T|^2}, \quad (13)$$

where C_B is the maximum value in the correlation plane over the background area, and C_T is the maximum value in the correlation plane over the target area in the scene. The background area and the target area are complementary. Ideally, values of the DC should be close to unity, indicating a good capacity to discriminate the target against unwanted objects in the background. Negative values of the DC indicate a failure to detect the target. The location accuracy can be characterized by means of the location errors defined as

$$LE = \sqrt{(x_T - \hat{x}_T)^2 + (y_T - \hat{y}_T)^2}, \quad (14)$$

where (x_T, y_T) are the exact coordinates where the correlation peak is expected to occur and $(\hat{x}_T, \hat{y}_T)$ are the coordinates where it is actually located after filtering.

The size of the images used in the experiments is 256×256 pixels with intensity values in the range $[0-255]$. We use the image of the toy car shown in Fig. 1(a) as the target. Two types of backgrounds are used: deterministic and stochastic. The stochastic backgrounds are realizations of colored random processes with correlation coefficients of 0.70 and 0.95 for the reference image and input scene, respectively. The size of the target is 62×50 pixels. The target mean and Std. Dev. have values 105 and 45, respectively. To guarantee statistically correct results, 30 statistical trials of each experiment for different positions of a target and realization of random processes were carried out. All scenes are corrupted by additive white Gaussian noise.

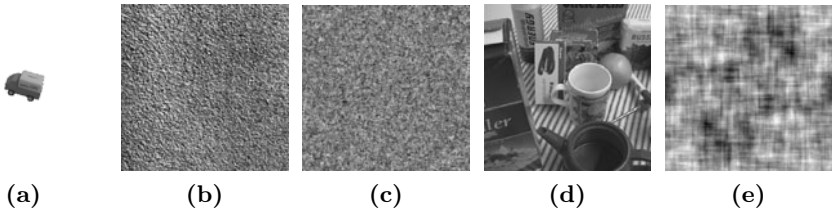


Fig. 1. (a) The target used in the experiments, (b) deterministic reference image background, (c) sample stochastic reference image background, (d) deterministic scene background, and (e) sample stochastic scene background

For comparison purposes, we include the ideal GOF, designed with all known parameters, to establish an upper bound on performance; the proposed GOF_{NN} filter when using $\hat{w}(x)$ as the estimation of the inverse support function; and a switching version of the GOF_{NN} , labeled sGOF_{NN} , which is designed by using only part of the GOF_{NN} . The GOF_{NN} can be regarded as the sum of two filters, corresponding to the part of the target and the part of the expected value of the background; that is, we can write the frequency response of the GOF_{NN} as

$$\text{GOF}_{\text{NN}}(\omega) = G(\omega)T^*(\omega) + G(\omega)\mu_s\bar{W}(\omega), \quad (15)$$

for a suitably defined $G(\omega)$. This is effectively an approximation of the ideal GOF when the reference image is corrupted by noise. However, when the estimation of the support function is degraded, it proves better to design a switching filter that uses either or both terms of (15) depending on the statistics of the target and the noise processes in a given problem instance.

We first perform experiments using synthetically generated backgrounds while varying the location of the target in the scene. We need to determine how robust the filters are with respect to the noise in the reference image. The simulation results are shown in Fig. 2 when the mean and Std. Dev. of the background in the reference image are 100 and 40, respectively. Since the statistics of the background and of the target are so similar, the estimations of the support function are severely degraded. The performance of the GOF is constant because this filter is designed with all parameters known. It can be seen that detection of the target is possible in the presence of noise with a Std. Dev. of up to 20. Location errors are small values of the Std. Dev. up to 25. In this case, the performance of the switching filter is not significantly better since there is very little of the support function available for the design of the filters.

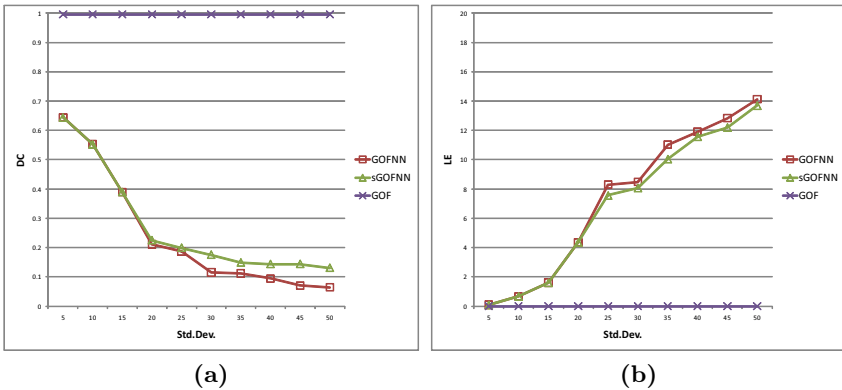


Fig. 2. Performance of filters in terms of (a) DC and (b) LE while varying the Std. Dev. of the reference image noise. The input scene is corrupted by additive noise with Std. Dev. of 10.

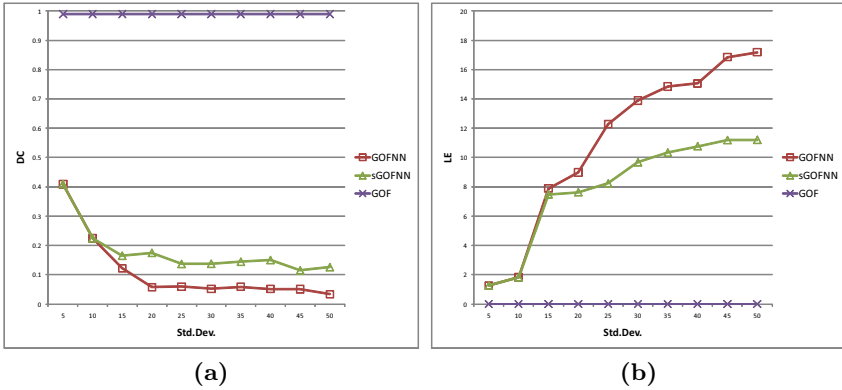


Fig. 3. Performance of filters in terms of (a) DC and (b) LE while varying the Std. Dev. of the reference image noise using deterministic backgrounds. The input scene is corrupted by additive noise with Std. Dev. of 10.

While stochastic backgrounds closely match the signal model, it is interesting to investigate the performance of the proposed filters when natural images are used as backgrounds. The simulation results are shown in Fig. 3. Because of the increased complexity of the backgrounds, it becomes harder to detect and locate the target. Thus, the performance for the GOF_{NN} is lower than when using stochastic backgrounds. However, in this case the switching filter consistently outperforms the GOF_{NN} . We can say that the target can be detected when the noise has a Std. Dev. of up to 10, while location errors are small up to a Std. Dev. of 20. When the noise levels increase, we can no longer consider the detection results to be reliable.

4 Conclusions

In this paper we proposed a novel filter for detecting and locating a target in nonoverlapping background noise by using a noisy image. The filter expression is derived from a new signal model that accounts for the presence of a cluttered background in the training image. Filter instances are designed using only information available in the reference image and statistical information of the noise processes in the model. Estimations were given for the parameters assumed unknown in the model. A switching filter was also proposed that, under certain conditions, performs better than the approximation of the ideal filter. With the help of computer simulations, we showed that the filters along with the proposed estimations, yield good detection results in the presence of moderate levels of noise.

References

1. VanderLugt, A.: Signal detection by complex spatial filtering. *IEEE Transactions on Information Theory* 10(2), 139–145 (1964)
2. Kumar, B.V.K.V., Mahalanobis, A., Juday, R.: *Correlation pattern recognition*. Cambridge University Press, Cambridge (2005)
3. Kumar, B.V.K.V., Hassebrook, L.: Performance measures for correlation filters. *Applied Optics* 29(20), 2997–3006 (1990)
4. Yaroslavsky, L.P.: The theory of optimal methods for localization of objects in pictures. In: Wolf, E. (ed.) *Progress in Optics*, pp. 145–201. Elsevier, Amsterdam (1993)
5. Kumar, B.V.K.V., Dickey, F.M., DeLaurentis, J.M.: Correlation filters minimizing peak location errors. *Journal of the Optical Society of America A* 9(5), 678–682 (1992)
6. Kober, V., Campos, J.: Accuracy of location measurement of a noisy target in a nonoverlapping background. *Journal of the Optical Society of America A* 13(8), 1653–1666 (1996)
7. Javidi, B., Wang, J.: Design of filters to detect a noisy target in nonoverlapping background noise. *Journal of the Optical Society of America A* 11(10), 2604–2612 (1994)
8. Javidi, B., Zhang, G., Parchekani, F.: Minimum-mean-square-error filters for detecting a noisy target in background noise. *Applied Optics* 35, 6964–6975 (1996)
9. Javidi, B.: *Real-Time Optical Information Processing*. Academic Press, London (1994)
10. Ramos-Michel, E.M., Kober, V.: Design of correlation filters for recognition of linearly distorted objects in linearly degraded scenes. *Journal of the Optical Society of America A* 24(11), 3403–3417 (2007)
11. Mahalanobis, A., VijayaKumar, B.V.K., Song, S., Sims, S.R.F., Epperson, J.F.: Unconstrained correlation filters. *Applied Optics* 33(17), 3751–3759 (1994)
12. González-Fraga, J., Kober, V., Álvarez-Borrego, J.: Adaptive synthetic discriminant function filters for pattern recognition. *Optical Engineering* 45, 057005 (2006)
13. Ramos-Michel, E.M., Kober, V.: Adaptive composite filters for pattern recognition in linearly degraded and noisy scenes. *Optical Engineering* 47, 047204 (2008)
14. Aguilar-González, P.M., Kober, V.: Correlation filters for pattern recognition using a noisy reference. In: Ruiz-Shulcloper, J., Kropatsch, W.G. (eds.) *CIARP 2008*. LNCS, vol. 5197, pp. 38–45. Springer, Heidelberg (2008)
15. Aguilar-González, P.M., Kober, V.: Correlation pattern recognition in nonoverlapping scene using a noisy reference. In: Bayro-Corrochano, E., Eklundh, J.-O. (eds.) *CIARP 2009*. LNCS, vol. 5856, pp. 555–562. Springer, Heidelberg (2009)
16. Yilmaz, A., Javed, O., Shah, M.: Object tracking: A survey. *ACM Computing Surveys* 38(4) (2006)
17. Pratt, W.K.: *Digital Image Processing*. John Wiley & Sons, Chichester (2007)

Image Fusion Algorithm Using the Multiresolution Directional-Oriented Hermite Transform

Sonia Cruz-Techica and Boris Escalante-Ramirez

Facultad de Ingeniería, Universidad Nacional Autónoma de México,
Edif. Bernardo Quintana, Circuito exterior, Cd. Universitaria, México, D.F. 04510
cruz_s@uxmcc2.iimas.unam.mx, boris@servidor.unam.mx

Abstract. The Hermite transform is introduced as an image representation model for multiresolution image fusion with noise reduction. Image fusion is achieved by combining the steered Hermite coefficients of the source images, then the coefficients are combined with a decision rule based on the linear algebra through a measurement of the linear dependence. The proposed algorithm has been tested on both multi-focus and multi-modal image sets producing results that exceed results achieved with other methods such as wavelets, curvelets [1], and contourlets [2] proving that our scheme best characterized important structures of the images at the same time that the noise was reduced.

Keywords: image fusion, Hermite transform, multiresolution, linear dependence.

1 Introduction

Image fusion can be defined as the process of combining information from different sources, in order to detect strong salient features in the input images and fuse these details into the fused image. In general, image fusion proposes the integration of disparate and complementary data to improve the information that appears on images as well as increased reliability and performance, which results in greater accuracy of data.

Fusion techniques can be divided into spatial domain and transform domain techniques [7]. In the first case, the input images are fused into spatial domain, the fusion process deals with the original pixel values. In contrast, in the transform domain techniques it is possible to use a framework where the salient features of the images are clearer than in the spatial domain.

In the literature several methods of pixel-level fusion have been reported which use a transformation to perform data fusion, some of these transformations are: the discrete wavelet transform (DWT) [1], the contourlet transform (CW) [15], the curvelet transform (CUW) [8], and the Hermite transform (HT) [4, 5].

The wavelet transform has been the most used technique for the fusion process but it is the technique with more problems in the analysis of signals from two or

more dimensions; an example of this is the points of discontinuity that sometimes are undetected; another drawback is its limitation to capture directional information. The contourlet and the curvelet transforms have shown better results than the wavelet transform due to multi-directional analysis, but they require an extensive orientation search at each level of the decomposition. Because of this, the Hermite transform provides significant advantages to the process of image fusion: first this model of representation includes some properties of human visual system such as the local orientation analysis and the Gaussian derivative model of primary vision [16] and it also has the additional advantage of reducing noise without introducing artifacts.

In this work, we take it as a prerequisite that the source images must be registered so that the corresponding pixels are aligned. The proposed scheme fuses images on a pixel-level using a multiresolution directional-oriented Hermite transform of the source images by means of a decision map. This map is based on a linear dependence test of the rotated Hermite coefficients.

The rest of the paper is organized as follows: Section 2 presents the basic concepts of Hermite Transform. Section 3 describes the proposed image fusion algorithm. Section 4 focuses on experiments, evaluation criteria and analysis of results. Finally conclusions are introduced in section 5.

2 The Hermite Transform (HT)

The Hermite transform (HT) [9] is a special case of polynomial transform, which is a technique of local decomposition of signals and can be regarded as an image description model. In this, the input image $L(x, y)$ is windowed with a local Gaussian function $\omega(x - p, y - q)$ at the positions (p, q) that conform the sampling lattice S . By replicating the window function over the sampling lattice, we can define the periodic weighting function as $W(x, y) = \sum_{(p, q) \in S} \omega(x - p, y - q)$. Then, the local information of each analysis window is expanded in terms of a family of orthogonal polynomials defined as

$$G_{n-m, m}(x, y) = \frac{1}{\sqrt{2^n (n-m)! m!}} H_{n-m}\left(\frac{x}{\sigma}\right) H_m\left(\frac{y}{\sigma}\right) . \quad (1)$$

where $H_i\left(\frac{x}{\sigma}\right)$ denotes the i th Hermite polynomial orthogonal to the Gaussian window with standard deviation σ .

In every window function, the signal content is described as the weighted sum of polynomials $G_{m, n-m}(x, y)$ of m degree in x and $n - m$ in y . In a discrete implementation, the Gaussian window function may be approximated by the binomial window function and in this case, its orthogonal polynomials are known as Krawtchouk's polynomials.

In either case, the polynomial coefficients $L_{m, n-m}(p, q)$ are calculated convolving the original image $L(x, y)$ with the analysis filters $D_{m, n-m}(x, y) = G_{m, n-m}(-x, -y)\omega^2(-x, -y)$, followed by subsampling at position (p, q) of the sampling lattice S . That is,

$$L_{m,n-m}(p,q) = \int_{-\infty}^{+\infty} \int_{-\infty}^{+\infty} L(x,y) D_{m,n-m}(x-p,y-q) dx dy . \quad (2)$$

The recovery process of the original image consists of interpolating the transform coefficients with the proper synthesis filters. This process is called inverse transformed polynomial and it is defined by

$$\hat{L}(x,y) = \sum_{n=0}^{\infty} \sum_{m=0}^n \sum_{(p,q) \in S} L_{m,n-m}(p,q) P_{m,n-m}(x-p,y-q) . \quad (3)$$

where $P_{m,n-m}(x,y) = (G_{m,n-m}(x,y) \omega(x,y) / W(x,y))$ are the synthesis filters of order m and $n-m$ for $m = 0, \dots, n$ and $n = 0, \dots, \infty$.

2.1 The Steered HT

The Hermite transform has the advantage of high-energy compaction by adaptively steering the HT [12], [10]. Steerable filters are a class of filters that are rotated copies of each filter, constructed as a linear combination of a set of basis filters. The steering property of the Hermite filters explains itself because this filters are products of polynomials with a radially symmetric window function. The $N+1$ Hermite filters of Nth-order form a steerable basis for each individual Nth-order filter. Because of this property, the Hermite filters at each position in the image adapt the local orientation content. In terms of orientation frequency functions, this property of the Hermite filters can be expressed by

$$g_{m,n-m}(\theta - \theta_0) = \sum_{k=0}^n c_{m,k}^n(\theta_0) g_{n-k,k}(\theta) . \quad (4)$$

where $c_{m,k}^n(\theta_0)$ is the steering coefficient. The orientation selectivity for the filter is expressed by

$$g_{m,n-m}(\theta) = \sqrt{\binom{n}{m}} \cos^m \theta \sin^{n-m} \theta . \quad (5)$$

For the directional Hermite decomposition, first, a HT was applied and then the coefficients were rotated toward the estimated local orientation θ , according to a criterion of maximum oriented energy at each window position. This angle can be approximated as $\theta = L_{01}/L_{10}$, where L_{01} and L_{10} are a good estimated to optimal edge detectors in the horizontal and vertical directions respectively (Fig. 1 shows the HT and the steered HT over an image).

2.2 The Multiresolution Directional-Oriented HT

A multiresolution decomposition using the HT can be obtained through a pyramid scheme [6]. In a pyramidal decomposition, the image is decomposed into a

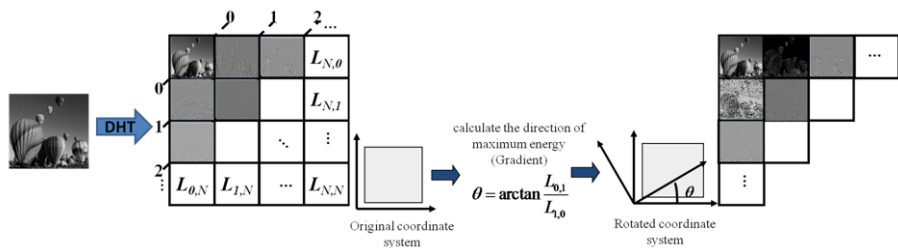


Fig. 1. The Discrete Hermite Transform (DHT) and the steered Hermite Transform over an image

number of band-pass or low-pass subimages, which are then subsampled in proportion to their spatial resolution. In each layer the zero order coefficients are transformed to obtain -in a lower layer- a scaled version of the one above. Once the coefficients of Hermite decomposition of each level are obtained, the coefficients can be projected to one dimension by its local orientation of maximum energy. In this way we obtain the multiresolution directional-oriented Hermite transform, which provides information about the location and orientation of the structure of the image at different scales.

3 Proposed Image Fusion Algorithm

Our approach aims at analyzing images by means of the HT, which allows us to identify perceptually relevant patterns to be included in the fusion process while discriminating spurious artifacts. As we have mentioned, the steered HT allows us to focus energy in a smaller number of coefficients, and thus the information contained in the first-order rotated coefficient may be sufficient to describe the edge information of the image in a particular spatial locality. If we extend this strategy to more than one level of resolution, then it is possible to obtain a better description of the image.

However, the success of any fusion scheme depends not only on the image analysis model but also on the fusion rule, therefore, instead of choosing for the usual selection operators based on the maximum pixel value, which often introduce noise and irrelevant details in the fused image, we seek a rule to consider the existence of a pattern in a region defined by a fixed-size window.

The general framework for the proposed algorithm includes the following stages. First a multiresolution HT of the input images is applied. Then, for each level of decomposition, the orientation of maximum energy is detected so that the coefficients can rotate, thus the first order rotated coefficient has the most information about edges. Afterwards, taking this rotated coefficient of each image we apply a linear dependence test. The result of this test is then used as a decision map to select the coefficients of the fused image in the multiresolution HT domain of the input images. If the original images are noisy, the decision map is applied on the multiresolution HT directional-oriented. The approximation coefficients in the case of HT are the zero order coefficients. In most multifocal and

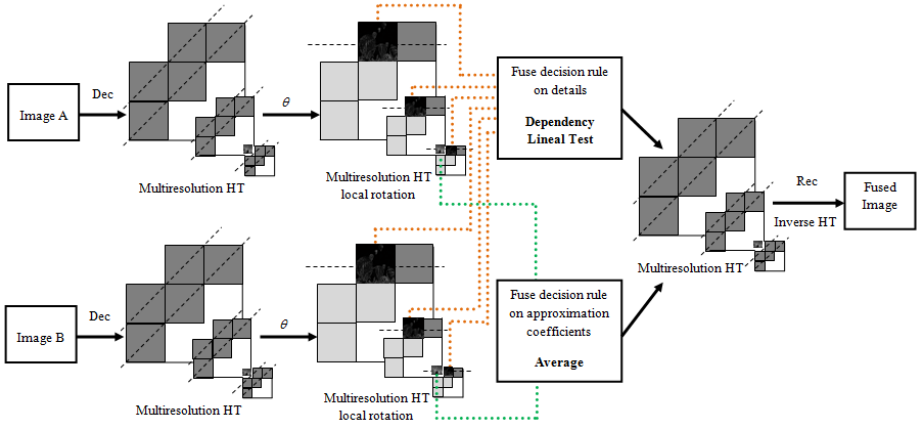


Fig. 2. Fusion scheme with the multiresolution directional-oriented Hermite Transform

multimodal applications the approximation coefficients of the input images are averaged to generate the zero order coefficient of the fused image, but this always depends on the application context. Finally, the fused image is obtained by applying the inverse multiresolution HT (Fig. 2 shows a simplified representation of this method).

3.1 The Fusion Rule

The linear dependence test evaluates the pixels inside a window of $w_s \times w_s$, if those pixels are linearly independent, then there is no relevant feature in the window. However, if the pixels are linearly dependent, it indicates the existence of a pattern. The fusion rule selects the coefficient with the highest dependency value. A higher value will represent a stronger pattern.

This approach has been reported in the literature in image fusion schemes that use the wavelet transform [1] and the curvelet transform [8]; their basis is an empirical method proposed in [3], where the image is analyzed in small regions and each neighborhood of a pixel is expressed as a vector in which the linear dependence is calculated.

A simple and rigorous test for determining the linear dependence or independence of vectors is the Wronskian determinant, which is defined for functions but can also be applied to vectors. The dependency of the window centered at a pixel (i, j) is described in

$$D_A(i, j) = \sum_{m=i-w_s}^{i+w_s} \sum_{n=j-w_s}^{j+w_s} L_A^2(m, n) - L_A(m, n) . \quad (6)$$

where $L_A(m, n)$ is the first order steered Hermite coefficient of the source image A with spatial position (m, n) . The coefficient of the fused HT is selected as the one with largest value of the dependency measure, then

$$L_F(i, j) = \begin{cases} L_A(i, j) & \text{if } D_A(i, j) \geq D_B(i, j) \\ L_B(i, j) & \text{if } D_A(i, j) < D_B(i, j) \end{cases} . \quad (7)$$

We apply this rule to all detail coefficients and average the zero order Hermite coefficients as

$$L_{00_F}(i, j) = \frac{1}{2} [L_{00_A}(i, j) + L_{00_B}(i, j)] . \quad (8)$$

4 Experiments and Results

The proposed algorithm was tested on several sets of multi-focus and multi-modal images. Fig. 3 shows one of the multi-focus image sets used and the results of image fusion achieved with different fusion methods, all of them using the linear dependence test with a window size of 3×3 , and two decomposition levels. By the HT, we used a Gaussian window with spread $\sigma = \sqrt{2}$, a subsampling factor $T = 2$ between each pyramidal level. The DWT used was db4 and in the case of the CW, the McClellan transform of 9-7 filters were used as directional filters and the wavelet db4 was used as pyramidal filters.

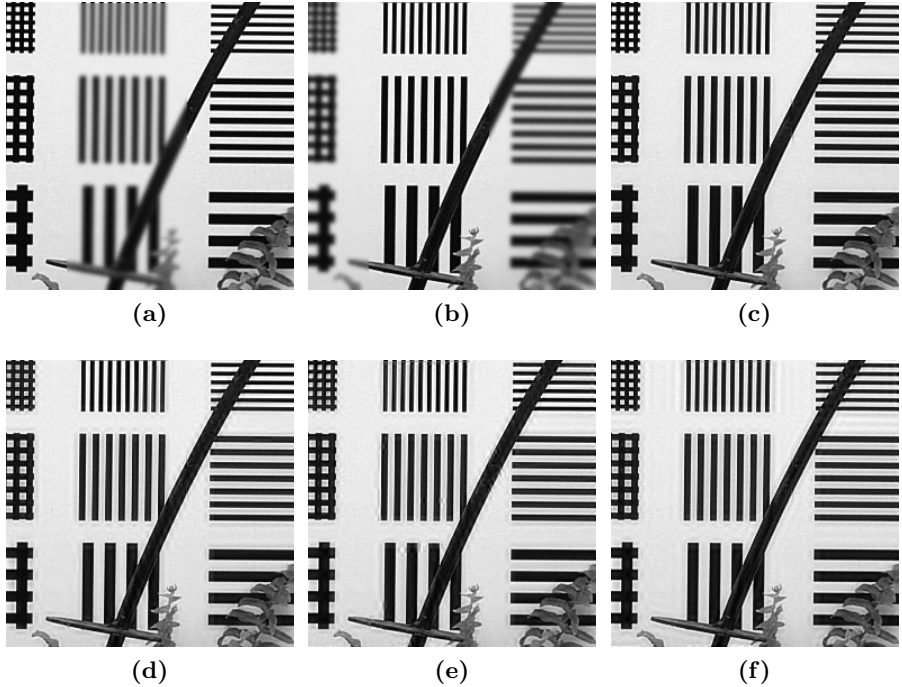


Fig. 3. Results of image fusion in multi-focus images, using different analysis techniques. (a) and (b) as the source images, (c) HT, (d) DWT, (e) CW and (f) CUW.

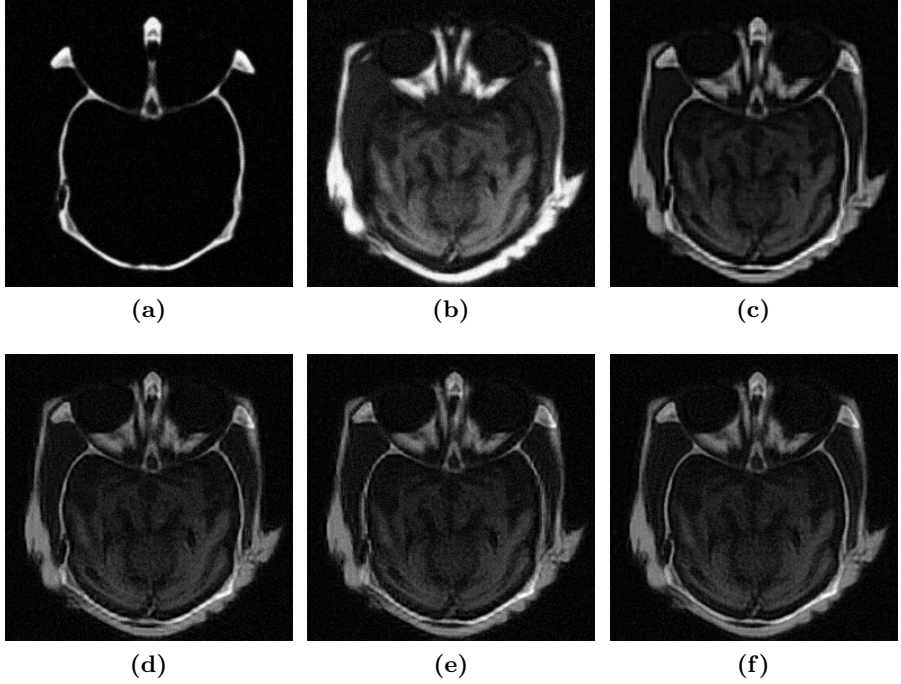


Fig. 4. Results of image fusion in noisy medical images, using different analysis techniques. (a) computed tomography (CT) and (b) magnetic resonance (MR) as the source images, c) HT, d) DWT, e) CW and f) CUW.

On the other hand, Fig. 4 shows the application in noisy medical images with the same described parameters. In this case, Gaussian noise with $\sigma = 0.001$ was introduced to the original images.

From Figs. 3 and 4, we can notice that the image fusion method based on the Hermite transform preserved the spatial resolution and information content of both images better. Moreover, our method shows a superior performance in noise reduction.

In order to quantitatively compare the proposed algorithm with the others, we evaluated our fusion results with several quality metrics: the peak signal to noise ratio (PSNR) defined in Eq. 9, the mean square error (MSE) defined in Eq. 10, the measure of structural similarity (SSIM) [13] defined in Eq. 11 and the Mutual information (MI) [14] defined in Eq. 12.

$$PSNR = 10 \log_{10} \frac{255^2 (MN)}{\sum_{i=1}^M \sum_{j=1}^N [F(i, j) - R(i, j)]^2} . \quad (9)$$

$$MSE = \frac{\sum_{i=1}^M \sum_{j=1}^N [F(i, j) - R(i, j)]^2}{MN} . \quad (10)$$

where $F(i, j)$ denotes the intensity of the pixel of the fused image and $R(i, j)$ denotes the intensity of the pixel of the original image.

$$SSIM(R, F) = \frac{\sigma_{RF}}{\sigma_R \sigma_F} \frac{2\mu_R \mu_F}{(\mu_R)^2 + (\mu_F)^2} \frac{2\sigma_R \sigma_F}{\sigma_R^2 + \sigma_F^2}. \quad (11)$$

where μ_R is the original image mean and μ_F the fused image mean; σ is the variance and σ_{RF} is the covariance.

$$MI_F^{AB} = MI_{FA}(F, A) + MI_{FB}(F, B). \quad (12)$$

where $MI_{FA}(F, A) = \sum P_{FA}(F, A) \log [(P_{FA}(F, A)) / (P_F(F) P_A(A))]$ is the amount of information that belongs to image A contained in the fused image, where P_F and P_A are the marginal probability density functions of images F and A respectively, and P_{FA} is the joint probability density function of both images.

Tab. 1 shows the performance of the method using different image analysis techniques with the same fusion rule. The values are the average of the tests performed on multifocal and medical images. Altogether we used 11 sets of images, in 6 of which, we compared with a ground truth. These ground truths were obtained from synthetic images.

Table 1. Performance measurement applying the fusion rule based on linear dependency with different methods

Fusion method	MSE	$PSNR$	$SSIM$	MI_F^{AB}
Hermite Transform	127.055	36.425	0.9640	6.130
Wavelet Transform	148.889	34.627	0.9574	5.595
Contourlet Transform	177.466	30.836	0.9477	5.535
Curvelet Transform	164.296	31.608	0.9496	5.609

5 Conclusions

We have presented a multiresolution image fusion method based on the directional-oriented HT, which uses a linear dependency test as fusion rule. We have experimented with this method for multi-focus and multi-modal images and we have obtained good results, even in the presence of noise. Both subjective and objective results show that the proposed scheme outperforms other existing methods.

The HT has proved to be an efficient model for the representation of images because derivatives of Gaussian are the basis functions of this transform, which optimally detect, represent and reconstruct perceptually relevant image patterns, such as edges and lines.

Acknowledgments. This work was supported by UNAM grants IN113611 and IX100610.

References

1. Aguilar-Ponce, R., Tecpanecatli-Xihuitl, J.L., Kumar, A., Bayoumi, M.: Pixel-level image fusion scheme based on linear algebra. In: IEEE International Symposium on Circuits and Systems ISCAS 2007, New Orleans, pp. 2658–2661 (2007)
2. Contourlet toolbox, <http://www.mathworks.com/matlabcentral/fileexchange/8837>
3. Durucan, E., Ebrahimi, T.: Change detection and background extraction by linear algebra. *Proceedings of the IEEE* 89(10), 1368–1381 (2001)
4. Escalante-Ramírez, B.: The Hermite transform as an efficient model for local image analysis: An application to medical image fusion. *Comput. Electr. Eng.* 34(2), 99–110 (2008)
5. Escalante-Ramírez, B., López-Caloca, A.: The Hermite transform: an efficient tool for noise reduction and image fusion in remote sensing. In: book: *Image Processing for Remote Sensing*, pp. 539–557. CRC Press, Boca Raton (2006)
6. Escalante-Ramírez, B., Silván-Cárdenas, J.L.: Advanced modeling of visual information processing: A multi-resolution directional-oriented image transform based on Gaussian derivatives. *Signal Processing: Image Communication* 20(9-10), 801–812 (2005)
7. Hill, P., Canagarajah, N., Bull, D.: Image Fusion using Complex Wavelets. In: *Proc. 13th British Machine Vision Conference*, pp. 487–496 (2002)
8. Mahyari, A., Yazdi, M.: A novel image fusion method using curvelet transform based on linear dependency test. In: *International Conference on Digital Image Processing*, pp. 351–354 (2009)
9. Martens, J.-B.: The Hermite transform-theory. *IEEE Transactions on Acoustics, Speech and Signal Processing* 38(9), 1595–1606 (1990)
10. Martens, J.-B.: Local orientation analysis in images by means of the Hermite transform. *IEEE Transactions on Image Processing* 6(8), 1103–1116 (1997)
11. The Curvelet.org team, <http://www.curvelet.org/software.html>
12. Van Dijk, A., Martens, J.-B.: Image representation and compression with steered Hermite transforms. *Signal Processing* 56(1), 1–16 (1997)
13. Wang, Z., Bovik, A., Sheikh, H., Simoncelli, E.: Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing* 13, 600–612 (2004)
14. Wang, Q., Yu, D., Shen, Y.: An overview of image fusion metrics. In: *Conference on Instrumentation and Measurement Technology*, pp. 918–923 (2009)
15. Yang, L., Guo, B.L., Ni, W.: Multimodality medical image fusion based on multi-scale geometric analysis of contourlet transform. *Neurocomputing* 72(1-3), 203–211 (2008)
16. Young, R.: The Gaussian derivative theory of spatial vision: analysis of cortical cell receptive field line-weighting profiles. *Integration. Technical report*, General Motors Research (1986)

Normalized Cut Based Edge Detection

Mario Barrientos and Humberto Madrid

Applied Mathematics Research Center
Autonomous University of Coahuila
Camporeddondo Unit, Building “S”, Postcode 25000
Saltillo, Coahuila, Mexico
{mbarrientosmate,hmadrid}@gmail.com
www.cima.uadec.mx

Abstract. This work introduces a new technique for edge detection based on a graph theory tool known as normalized cut. The problem involves to find certain eigenvector of a matrix called normalized laplacian, which is constructed in such way that it represents the relation of color and distance between the image’s pixels. The matrix dimensions and the fact that it is dense represents a trouble for the common eigensolvers. The power method seemed a good option to tackle this problem. The first results were not very impressive, but a modification of the function that relates the image pixels lead us to a more convenient laplacian structure and to a segmentation result known as edge detection. A deeper analysis showed that this procedure does not even need of the power method, because the eigenvector that defines the segmentation can be obtained with a closed form.

Keywords: Edge detection, normalized cut, power method, image segmentation.

1 Normalized Cut

A graph G its conformed by a pair of sets (V, E) , where V is a finite set of points $v_1, v_2, \dots, v_n$ called nodes, and E is the set of edges $e(i, j)$ that connects the nodes v_i and v_j . This edges will have an assigned weight w_{ij} and will be non directed, which means that $w_{ij} = w_{ji}$. The idea to construct a graph from an image is consider each pixel of the image as a node of the graph. The weights of the edges will be assigned with a function that relates pairs of pixels, taking into account characteristics like the color similarity and the distance among them.

A graph $G = (V, E)$ can be splitted in two disjoint sets A and B , with $A \cup B = V$ y $A \cap B = \emptyset$, simply by removing the edges connecting both parts. We say that A and B are a bipartition of G . Their grade of dissimilarity can be calculated as the sum of the weights of the removed edges. In graph theory this process is called cut:

$$\text{cut}(A, B) = \sum_{u \in A, v \in B} w(u, v), \quad (1)$$

with $w(i, j) = w_{ij}$. The optimal bipartition is the one that minimizes the cut value. Wu and Leahy [1] proposed a grouping method based on the minimum cut criteria that produced good segmentations for some images. They also noted that the minimum cut criteria favors the formation of small sets containing only a few isolated nodes. To avoid this tendency, Shi and Malik [2] proposed a new measure of dissociation of groups called *Normalized Cut* (Ncut)

$$\text{Ncut}(A, B) = \frac{\text{cut}(A, B)}{\text{assoc}(A, V)} + \frac{\text{cut}(A, B)}{\text{assoc}(B, V)}, \quad (2)$$

where $\text{assoc}(A, V) = \sum_{u \in A, t \in V} w(u, t)$ is the total of connections of the nodes from A to the nodes of the graphic and $\text{assoc}(B, V)$ is defined similarly.

1.1 Calculating the Optimal Partition

Given a partition of the nodes of a graph G in two sets A and B , we make:

- $\mathbf{x}$ an indicator vector of dimension $N = |V|$, with $\mathbf{x}_i = 1$ if the node i is in A and $\mathbf{x}_i = -1$ if the node belongs to B .
- $\mathbf{W}$ a symmetric matrix of $N \times N$ with $\mathbf{W}(i, j) = w_{ij}$, known as *adjacency matrix*.
- $\mathbf{D}$ a diagonal matrix of $N \times N$ whose diagonal elements are the row-wise sum of the entries of $\mathbf{W}$, $\mathbf{D}(i, i) = \sum_j \mathbf{W}(i, j)$.

In [2] is shown that minimize (2) is equivalent to solve

$$\min_x \text{Ncut}(\mathbf{x}) = \min_y \frac{\mathbf{y}^T (\mathbf{D} - \mathbf{W}) \mathbf{y}}{\mathbf{y}^T \mathbf{D} \mathbf{y}}, \quad (3)$$

with $\mathbf{1}$ a vector of all-one elements of dimension N such that $\mathbf{y}^T \mathbf{D} \mathbf{1} = 0$ and $\mathbf{y}_i \in \{1, -b\}$, where b depends of the proportion of the cardinalities of A and B . The procedure to obtain (3) and how b is defined can be found in [3].

Note that (3) is a Rayleigh quotient [4]. If the domain restriction of $\mathbf{y}$ is relaxed and it is allowed to take real values we can minimize Ncut solving the eigensystem

$$(\mathbf{D} - \mathbf{W})\mathbf{y} = \lambda \mathbf{D} \mathbf{y}, \quad (4)$$

that can be rewritten as the standard eigensystem

$$\mathbf{D}^{-\frac{1}{2}} (\mathbf{D} - \mathbf{W}) \mathbf{D}^{-\frac{1}{2}} \mathbf{z} = \lambda \mathbf{z}, \quad (5)$$

with $\mathbf{z} = \mathbf{D}^{\frac{1}{2}} \mathbf{y}$.

Is easy to verify that $\mathbf{z}_0 = \mathbf{D}^{\frac{1}{2}}$ is the eigenvector of (5) with associated eigenvalue zero. Moreover, $\mathbf{D}^{-\frac{1}{2}} (\mathbf{D} - \mathbf{W}) \mathbf{D}^{-\frac{1}{2}}$, known as *normalized Laplacian*, is symmetric positive-semidefinite because is known that $(\mathbf{D} - \mathbf{W})$ is positive-definite [5]. Therefore $\mathbf{z}_0$ is the eigenvector associated to the smallest eigenvalue, also called smallest eigenvector of (5) and all of the eigenvectors of (5) are perpendicular to each other. Retaking one of the main characteristics of the Rayleigh

quotient, the second smallest eigenvector $\mathbf{z}_1$ of the generalized eigensystem (4) is the real-valued solution of the normalized cut problem (4).

In (3) is shown that the bipartition defined by the signs of the elements of $\mathbf{y}$ and $\mathbf{z}_1$ is the same. This property is not necessarily preserved numerically. To deal with this, some criterion should be adopted to split $\mathbf{z}_1$ in two sets.

2 Power Method

The practical difficulty of segmentation using the normalized cut method lies in the enormous amount of data that needs to be generated and stored. An image of $m \times n$ generates a Laplacian of $N \times N$, with $N = m \cdot n$. The Lanczos method has been used formerly to find $\mathbf{z}_1$ (2). We proposed to use power method to get this vector, because this method has shown good performance in other problems involving big dimensions matrices, such as PageRank (6).

The power method is enounced as follows:

Let $\mathbf{A}$ be a $n \times n$ diagonalizable matrix with eigenvalues $\lambda_1, \lambda_2, \dots, \lambda_n$, such that $\|\lambda_1\| > \|\lambda_j\|$ for $j = 2, \dots, n$ (λ_1 is the dominant eigenvalue), and $\mathbf{x}_0$ an initial vector of dimension n . The sequence

$$\mathbf{x}_k = \mathbf{A}\mathbf{x}_{k-1}$$

converge to the eigenvector $\mathbf{v}_1$ associated to λ_1 ($\mathbf{v}_1$ is the dominant eigenvector).

The initial vector $\mathbf{x}_0$ can be chosen randomly. The stop criterion we used is the relative error between the iterations k and $k - 1$.

The eigenvector that we need to obtain the segmentation is not the dominant one, so we need to use deflation and shift (4) to obtain it with the power method. We know that the normalized laplacian $\mathbf{L}$ is symmetric positive semi-definite, so its smallest eigenvector is $\mathbf{u}_1 = \mathbf{D}^{\frac{1}{2}}\mathbf{1}$ with $\lambda_1 = 0$, and that its eigenvalues are on the interval $[0, 2]$, (7). Considering this, the procedure to calculate the subdominant eigenvector of $\mathbf{L}$ is:

1. Apply a shift to $\mathbf{L}$ with $s = 2$ to obtain $\mathbf{L}_s = \mathbf{L} - s\mathbf{I}$, with $\mathbf{I}$ identity matrix of $N \times N$.
2. Deflate $\mathbf{L}_s$ using $\bar{\mathbf{u}}_1 = \mathbf{u}_1 / \|\mathbf{u}_1\|$ and $\lambda'_1 = -2$, obtaining $\mathbf{L}_d = \mathbf{L}_s - \lambda'_1 \bar{\mathbf{u}}_1 \bar{\mathbf{u}}_1^T$.
3. Use power method with $\mathbf{L}_d$ to get $\mathbf{u}_2$.

To avoid the explicit construction of $\bar{\mathbf{u}}_1 \bar{\mathbf{u}}_1^T$ required for the deflation, we will do the next modification to the procedure. After shift and deflation the matrix $\mathbf{L}$ becomes

$$\mathbf{L}_d = \mathbf{L}_s - \lambda'_1 \bar{\mathbf{u}}_1 \bar{\mathbf{u}}_1^T. \quad (6)$$

Multiplying (6) by $\mathbf{x}_0$ produces

$$\begin{aligned} \mathbf{L}_d \mathbf{x}_0 &= (\mathbf{L}_s - \lambda'_1 \bar{\mathbf{u}}_1 \bar{\mathbf{u}}_1^T) \mathbf{x}_0 \\ &= \mathbf{L}_s \mathbf{x}_0 - \lambda'_1 \bar{\mathbf{u}}_1 (\bar{\mathbf{u}}_1^T \mathbf{x}_0). \end{aligned}$$

We make $\mathbf{x}_1 = \mathbf{L}_d \mathbf{x}_0$. Now the product of $\mathbf{L}_d \mathbf{x}_1$ is

$$\mathbf{L}_d \mathbf{x}_1 = \mathbf{L}_d^2 \mathbf{x}_0 = \mathbf{L}_s \mathbf{x}_1 - \lambda'_1 \bar{\mathbf{u}}_1 (\bar{\mathbf{u}}_1^T \mathbf{x}_1).$$

Continuing in the same way $\mathbf{x}_k$ will be

$$\mathbf{x}_k = \mathbf{L}_d^k \mathbf{x}_0 = \mathbf{L}_s \mathbf{x}_{k-1} - \lambda'_1 \bar{\mathbf{u}}_1 (\bar{\mathbf{u}}_1^T \mathbf{x}_{k-1}). \quad (7)$$

Hence, we will do the power method iterations as is indicated by (7).

3 Weight Function and Segmentation

A crucial point for the segmentation result is the election of the weight function with which the edges weights will be assigned. In [2] the proposed function is:

$$w_{ij} = e^{\frac{-\|F(i)-F(j)\|^2}{\sigma_I}} * \begin{cases} e^{\frac{-\|X(i)-X(j)\|^2}{\sigma_X}} & \text{if } \|X(i) - X(j)\| \leq r \\ 0 & \text{otherwise} \end{cases}. \quad (8)$$

F represents the color characteristics of the pixels. For an RGB image $F(i) = (r(i), g(i), b(i))$, considering that the pixels of the image are reshaped as an N size vector and being $r(i)$, $g(i)$ and $b(i)$ the corresponding values for the pixel i in the red, green and blue layers; for images on HSV format F will be $F(i) = [v(i), v \cdot s \cdot \sin(2\pi \cdot h), v \cdot s \cdot \cos(2\pi \cdot h)]$, being $h(i)$, $s(i)$ and $v(i)$ the corresponding hue, saturation and value components of each pixel; finally, F will be the intensity of the pixel for a grayscale image. For all the cases, $X(i)$ will be the location of the pixel on the image, being $(1, 1)$ the upper left corner. The r parameter will define a neighborhood for each pixel. It is considered that the pixels that are further than r have not significative relationship with the central pixel. σ_I and σ_X are parameters that need to be calibrated.

Once the weight function has been set, we only need to define a criterion to split $\mathbf{u}_2$ in two sets, a threshold in this case, to finally have the complete segmentation process defined. As is supported in [3], the best results are obtained splitting the elements according to its sign, that is

$$\mathbf{x}(i) = \begin{cases} 1 & \text{if } \mathbf{u}_2(i) > 0 \\ -1 & \text{if } \mathbf{u}_2(i) \leq 0 \end{cases} \quad (9)$$

The results obtained using power method and the weight function (8) were good, but the memory and computing time requirements were not improved significantly, as can be seen in [8].

4 Edge Detection

Our first modification to the segmentation procedure was the introduction of the weight function

$$w_{ij} = e^{\frac{-\|F(i)-F(j)\|}{\sigma_I}} * \begin{cases} e^{\frac{-\|X(i)-X(j)\|}{\sigma_X}} & \text{if } \|X(i) - X(j)\| \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (10)$$

The differences with (8) are that the arguments of the exponential functions are not squared, and that r is fixed to 1, this last being the most important one. The segmentation obtained with a grayscale image using (10) is presented on Fig. 1

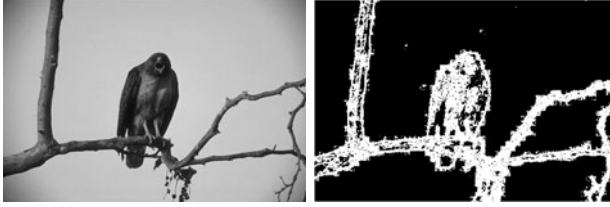


Fig. 1. First example of edge detection using a 354×451 pixels grayscale image with $\sigma_I = 255$ and $\sigma_X = \sqrt{354^2 + 451^2}$

This kind of segmentation is known as edge detection. The distance factor does not modify the resulting segmentation, so we can drop it and define our weight function for edge detection as

$$w_{ij} = \begin{cases} e^{\frac{-\|F(i)-F(j)\|}{\sigma_I}} & \text{if } \|X(i) - X(j)\| \leq 1 \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

The segmentation results are greatly improved by applying a median filter to the image. This filter has the property of remove noise preserving the edges [9]. For images on RGB and HSV formats, the filter is applied to each layer of the image. The obtained edges are thick, but they can be thinned using methods like non maxima suppression [10] or the edge thinning algorithm implemented in the MATLAB's Image Processing Toolbox [11], if necessary. To show the segmentation results at several σ_I values, the results are presented as soft boundary maps in Fig. 2

4.1 Simplifying the Method

An interesting fact of this edge detection method is that the power method always needs only one iteration to converge with 1×10^{-5} precision when

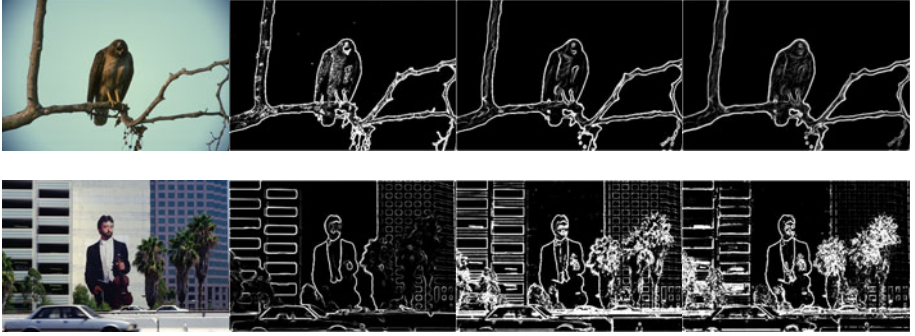


Fig. 2. Soft boundary maps obtained with images from the Berkeley Segmentation Dataset [12]. The second column shows the results obtained from the grayscale version of the images; the third column contains the segmentation obtained from the RGB format images; and the fourth column presents the result obtained from the HSV format images. All the images were preprocessed with a 7×7 neighborhood size median filter and using 30 different evenly spaced σ_I values in the intervals $[255, 7650]$ for the grayscale case, $[1, 30]$ for the RGB case and $[2, 60]$ for the HSV case.

$\mathbf{x}_0 = \mathbf{1}$. Searching for an explanation to this behavior, we noted that the adjacency matrix $\mathbf{W}$ has some characteristics that can be exploited to simplify our edge detection scheme. Our strongest supposition is that the adjacency matrix $\mathbf{W}$ can be approximated by $\mathbf{W} \approx \mathbf{I} + \mathbf{P}$, where $\mathbf{I}$ is an identity matrix of $N \times N$ and $\mathbf{P}$ is a matrix of $N \times N$ with all-one entries on the diagonals $n, -1, 1$ and n , where the zero index corresponds to the main diagonal and the negative indexes correspond to the diagonals below the main diagonal and vice versa. Based on the former, we can take the liberty of approximate $\mathbf{D}$ by $\mathbf{D}^* = 5\mathbf{I}$.

Using both suppositions, we can approximate the normalized Laplacian by

$$\mathbf{L}^* = \frac{4}{5}\mathbf{I} - \frac{1}{5}\mathbf{P}.$$

Applying a shift with $s = 2$ and deflation with $\lambda'_1 = -2$ and $\bar{\mathbf{u}}_1 = \mathbf{u}_1 / \|\mathbf{u}_1\|$, we obtain

$$\mathbf{L}_d^* = \mathbf{L}_s^* + 2\bar{\mathbf{u}}_1\bar{\mathbf{u}}_1^T. \quad (12)$$

It is easy to verify that the second iteration of the power method using (12) will return a vector that is very close to a multiple of the first iteration, and the same will happen with the next iterations. Being like that, the first iteration of the power method is a good enough approximation of the second smallest eigenvector of $\mathbf{L}$. This means that we can get $\mathbf{u}_2$ as

$$\mathbf{u}_2^* = -2\mathbf{1} + 2\bar{\mathbf{u}}_1\bar{\mathbf{u}}_1^T\mathbf{1}. \quad (13)$$

This means that our edge detection method is no longer an iterative process, because (13) give us a closed form to obtain the segmentation. The segmentations obtained with (13) and the ones obtained with power method are visually

undistinguishable. The former and the derivation of (13) are explained in detail in [8]. Our final version of the normalized cut based edge detection method is synthesized on Algorithm 1.

Input: Image $\mathbf{A}$, σ_I
Output: Segmentation $\mathbf{S}$
 $m, n \leftarrow$ Dimensions of $\mathbf{A}$
 $\mathbf{W} \leftarrow$ Build $\mathbf{W}$ from $\mathbf{A}$ using (11) with σ_I
 $\mathbf{D}(i, i) \leftarrow \sum_j \mathbf{W}(i, j)$
 $\mathbf{u}_1 \leftarrow \mathbf{D}^{\frac{1}{2}} \mathbf{1}$
 $\mathbf{u}_1 \leftarrow \mathbf{u}_1 / \|\mathbf{u}_1\|$
 $c \leftarrow \mathbf{u}_1^T \mathbf{1}$
 $\mathbf{u}_2 \leftarrow -\mathbf{1} + c\mathbf{u}_1$ // same signs that $-\mathbf{21} + 2c\mathbf{u}_1$
 $\mathbf{x} \leftarrow \text{sign}(\mathbf{u}_2)$
 $\mathbf{S} \leftarrow$ Reshape $\mathbf{x}$ as a $m \times n$ matrix

Algorithm 1. Normalized cut based edge detection

It is relevant to highlight some characteristics of the Algorithm 1. $\mathbf{W}$ is a symmetric pentadiagonal matrix with all-one elements on its main diagonal, which means that we only need to calculate two of the diagonals of $\mathbf{W}$. In fact, is not necessary to explicitly build $\mathbf{W}$, because is possible to obtain the entries of $\mathbf{D}$ directly. Moreover, $\mathbf{D}$ can be handled as a vector of dimension N . An appropriate implementation of this edge detection method can reduce the required storing space to a vector of N elements. Also, the complexity of the algorithm is $O(N)$ with small constant.

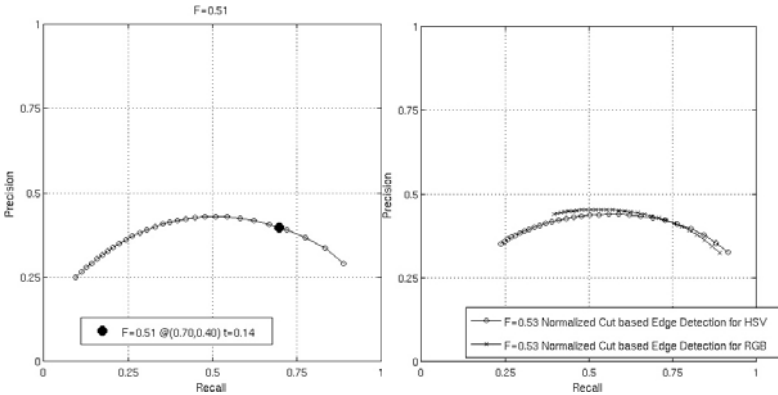


Fig. 3. Results obtained by our algorithms (*grayscale on left and color on right*) on the Berkeley Segmentation Benchmark using the same specs for the soft boundary maps as in the results of Fig. 2. F corresponds to the F-measure, which is the harmonic mean of precision and recall calculated at each level of the soft boundary map. The maximum F-measure value across an algorithm's precision-recall curve is reported as its summary statistic.

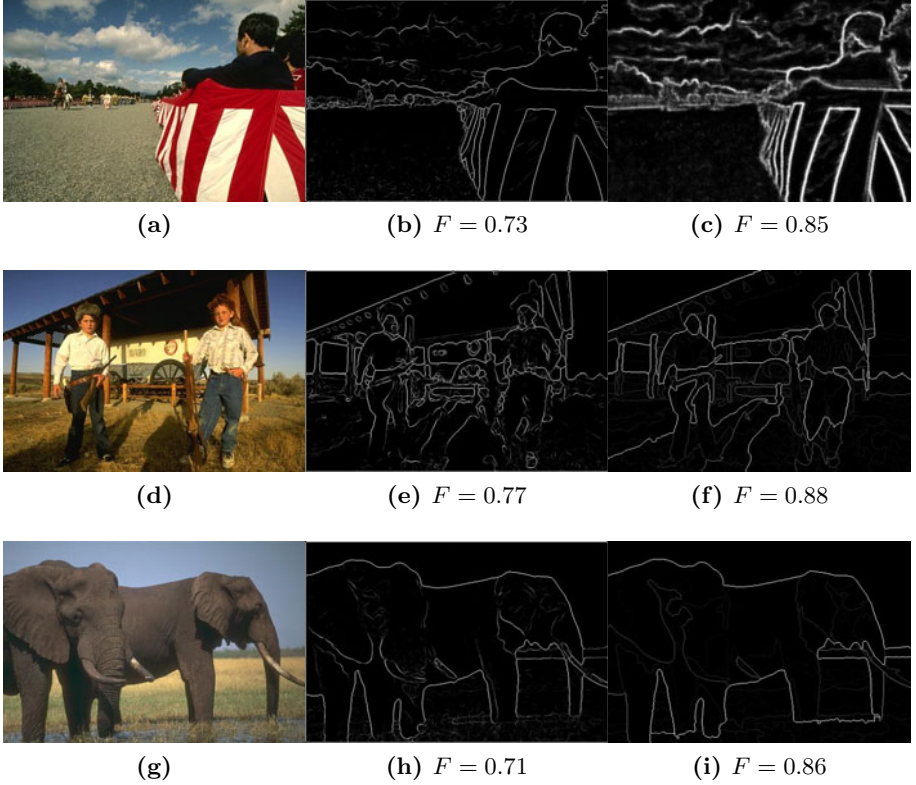


Fig. 4. Images from the BSD300 dataset segmented with the RGB version of the algorithm. Central columns shows our results using the specs indicated in Fig. 2 and the edge thinning algorithm included in MATLAB [11]. Right column shows top results obtained with (c) *boosted edge learning* [13] and (f) and (i) with *ultrametric contour maps* [14]. The F measure is shown for every result (greater values means better qualification).

The segmentations obtained with this simplified method are graphically identical to the obtained with our first version of the method, but the simplified one has a remarkable superior performance. The results obtained on the Berkeley Segmentation Benchmark are shown on Fig. 3. According to the obtained scores, our method is ranked tenth for the grayscale version and ninth for the RGB and HSV version on the list of reported edge detection algorithms. Figure 4 shows some comparisons of our method results with those of the best methods reported so far.

5 Conclusions

This paper presented a novel edge detection technique based on the original segmentation scheme introduced by Shi-Malik. The method was developed mainly

by moderately modifying the function that relates the pixels and by considering only the relation among the pixels on a neighborhood of radius one. Using a very simple approximation of the adjacency matrix lead us to obtain the edge detection with a very simple closed form, basically a sum of vectors. The most remarkable characteristic of the method is its simplicity, which translates in economy of computational resources.

The method was capable of work with images on color and grayscale images. Overall results are on the level of those of the gradient methods, but ours have a high variability. We considered that this notorious variation of scores between images is related with the textures, since the best results are obtained with images containing simple textures.

References

1. Wu, Z., Leahy, R.: An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 15(11) (1993)
2. Shi, J., Malik, J.: Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 22(8) (2000)
3. Avalos, V.: Segmentación de imágenes usando técnicas espectrales de grafos (2007)
4. Golub, G., Van-Loan, C.: *Matrix Computations*, 3rd edn. John Hopkins Press, Baltimore (1996)
5. Pothén, A., Simon, H.D., Liou, K.P.: Partitioning sparse matrices with eigenvectors of graphs. *SIAM Journal on Matrix Analysis and Applications* 11(3), 430–450 (1990)
6. Langville, A.N., Meyer, C.D.: *Google's PageRank and Beyond: The Science of Search Engine Rankings*, pp. 40–41. Princeton University Press, Princeton (2006)
7. Chung, F.: *Spectral Graph Theory*. Number 92 in CBMS Regional Conference Series in Mathematics. American Mathematical Society, Providence (1997)
8. Madrid, H., Barrientos, M.: Detección de bordes basada en corte normalizado. In: *Latin American Conference on Networked and Electronic Media* (2010)
9. Arce, G.R.: *Nonlinear signal processing: a statistical approach*. Wiley, Chichester (2005)
10. Lindeberg, T.: Edge detection and ridge detection with automatic scale selection. *International Journal of Computer Vision* 30(2), 117–154 (1996)
11. MathWorks: *Morphological operations on binary images* (2011)
12. Martin, D., Fowlkes, C., Tal, D., Malik, J.: A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics. In: *Proc. 8th Int'l Conf. Computer Vision*, vol. 2, pp. 416–423 (2001)
13. Dollar, P., Tu, Z., Belongie, S.: Supervised learning of edges and object boundaries. In: *IEEE Computer Vision and Pattern Recognition, CVPR 2006* (2006)
14. Arbeláez, P.: *Boundary extraction in natural images using ultrametric contour maps*. Technical report, Université Paris Dauphin (2006)

Homogeneity Cues for Texel Size Estimation of Periodic and Near-Periodic Textures

Rocio A. Lizarraga-Morales, Raul E. Sanchez-Yanez, and Victor Ayala-Ramirez

Universidad de Guanajuato DICIS
Salamanca, Guanajuato, Mexico

rocio.lizarraga@laviria.org, {sanchezzy,ayalav}@ugto.mx

Abstract. Texel size determination on periodic and near-periodic textures, is a problem that has been addressed for years, and currently it remains as an important issue in structural texture analysis. This paper proposes an approach to determine the texel size based on the computation and analysis of the texture homogeneity properties. We analyze the homogeneity feature computed from difference histograms, while varying the displacement vector for a preferred orientation. As we vary this vector, we expect a maximum value in the homogeneity data if its magnitude matches the texel size in a given orientation. We show that this approach can be used for both periodic and near-periodic textures, it is robust to noise and blur perturbations, and its advantages over other approaches in computation time and memory storage.

Keywords: Texel size detection, Textural periodicity, Difference histogram, Similarity test.

1 Introduction

Visual texture is a perceived property on the surface of all objects around us and can be an important reference for their characterization. From the structural point of view, it is widely accepted to define the texture as a conjunction of two components: i) a texture element (texel), which is the fundamental microstructure in the image [22], and ii) a set of rules for texel placement into the field of view. Such components can be used in several applications like shape from texture [2], texture synthesis [13,11,14], texture compression [14], among others. Furthermore, the texel can be used as a reference to improve the performance in classification [6,12] and segmentation [18] tasks, and to achieve scale invariant texture analysis [21].

Texel size determination on periodic and near-periodic textures, is a problem that has been addressed for years. A typical approach is the use of the co-occurrence matrix (CM) proposed by Haralick [5]. This methodology has been widely used, mainly by exploiting its parametrization. Selkainaho *et al.* [17] detect texture periodicity by using κ statistics emphasizing its computational advantages over χ^2 statistics. Oh *et al.* [16], have proposed a fast determination of textural periodicity using a binary co-occurrence matrix, improving the

processing time in comparison with the CM in gray level images. But Recently, other non-CM-based approaches have been proposed. Grigorescu and Petkov [3], determine texel size of periodic and near-periodic images based on the calculation of Renyi's generalized entropies by assuming a square texel for all textures. Other studies for texture periodicity detection are found in the literature, we can mention those based on the wavelet transform [8], autocorrelation [10,9], or regular bands [15].

In this work, we explore the use of difference histograms, originally proposed by Unser [19], to detect the texel size of both periodic and near-periodic visual textures. On the proposed approach, we specifically use the homogeneity property computed from the difference histogram (DH). Our method exploits the fact that the homogeneity attains its maximum value when the parameter of DH takes the value of the texel size or any positive integer multiple of it. Moreover, DH computation can be done more efficiently than CM both in memory usage and algorithmic complexity.

This paper is structured as follows: Section 2 describes the homogeneity feature used for the proposed method and our approach to estimate the texel size. In Section 3 we present the experiments and results performed to validate our method on a set of corrupted textures and natural near-periodic textures. In this section, it is also presented a computation time comparison with other approaches. Section 4 presents a summary of this work and our conclusion.

2 Texel Size Estimation Using Homogeneity Cues

Sum and Difference Histograms (SDH), were introduced by Unser [19] as an alternative to the usual co-occurrence matrix (CM). Unlike the CM that occupies K^2 memory elements for an image with K gray levels, SDH present memory storage reduction, since they only occupy two arrays of $2K - 1$ integers. To obtain the SDH, let us define an image I of $M \times N$ pixels size, which has K gray levels $k = 0, 1, \dots, K - 1$. Consider a pixel positioned in the coordinates (m, n) with intensity $I_{m,n}$ and a second pixel in the relative position $(m + v_m, n + v_n)$ with intensity $I_{m+v_m, n+v_n}$. The non-normalized sum and differences, of two pixels associated with the relative displacement vector $V = (v_m, v_n)$, are defined as:

$$s_{m,n} = I_{m,n} + I_{m+v_m, n+v_n} \quad (1)$$

$$d_{m,n} = I_{m,n} - I_{m+v_m, n+v_n}. \quad (2)$$

Sum and Difference Histograms h_s and h_d , with displacement vector $V = (v_m, v_n)$ over the image domain D , are defined as:

$$h_s(i) = \text{Card}\{(m, n) \in D, s_{m,n} = i\} \quad (3)$$

$$h_d(j) = \text{Card}\{(m, n) \in D, d_{m,n} = j\}. \quad (4)$$

The normalized sum and differences histograms are estimations of the sum and difference probability functions defined by $P_s(i)$ and $P_d(j)$.

A number of features computed from the probability functions have been proposed to be used as textural features, but the result largely depends on the size of the displacement vector V . These features for texture description have been proposed as a simplification of those proposed by Haralick [5]. In this study we specifically use the homogeneity, defined as:

$$\text{homogeneity} = G = \sum_j \frac{1}{1 + j^2} \cdot P_d(j) \quad (5)$$

This feature has two advantages, it is normalized in the range $[0, 1]$, and it only uses the difference probability distribution, reducing memory and time consumption.

In this paper, we analyze the difference histogram behavior to estimate the period of a given texture. Differences $d_{m,n}$, resulting from the use of a V that matches the period only take values of zero, as the reference pixel $I(m, n)$ and the relative pixel $I(m + v_m, n + v_n)$ have the same value. If the difference histogram has recorded only one value, the homogeneity function G reaches its maximum value.

Estimation of the period in a given direction can be done by setting a component to 0. That is, to detect the horizontal period of the texture, we set $v_n = 0$. We can detect periodicity values for T_m ranging from 2 to $\frac{M}{2}$. In a similar way, the detection of the texture period in the vertical direction T_n can be done by setting $v_m = 0$. We can detect values of $T_n \in [2, \frac{N}{2}]$.

As an example of this method, we show an artificial and periodic texture with a texel size of 60×40 pixels in Fig. 1a, and its homogeneity plot in Fig. 1b. We can see the periodicity of both plots as they present maxima values at the corresponding texel size multiples, 60 and 120 for horizontal detection, and 40, 80 and 120 for vertical detection.

3 Experiments and Results

In this section, we present an experimental evaluation of the proposed approach. We have separated our experiments in two parts: the first part consists in evaluating our approach with a corrupted regular pattern and the second part consists in using natural near-periodic images as inputs to our method.

3.1 Texel Size Estimation of a Corrupted Regular Pattern

In order to evaluate the limits of our approach under different conditions, we have corrupted a periodic texture pattern (Fig. 1a) with varying blur and noise levels. With this pattern, we have a benchmark, as we know that the pattern has a texel size of 60×40 pixels.

Blur. In this section, we have applied a simple low-pass filter a number of times in order to obtain a blur effect. The blur effect causes the loss of detail in the

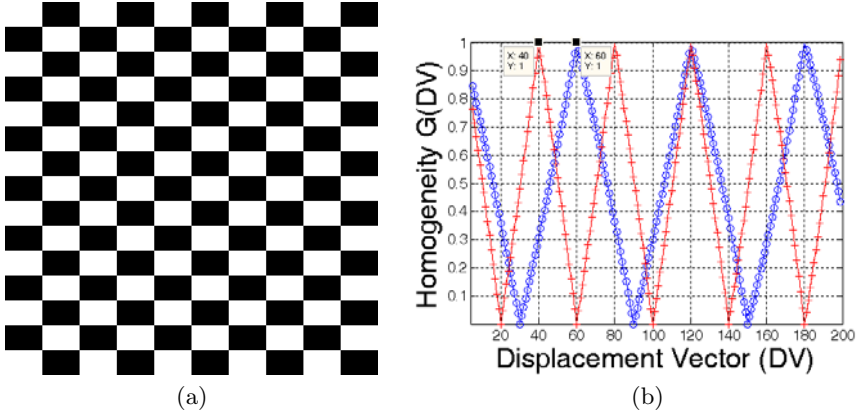


Fig. 1. An artificial texture (a) and its homogeneity function (b) in both, horizontal (o) and vertical (+) directions

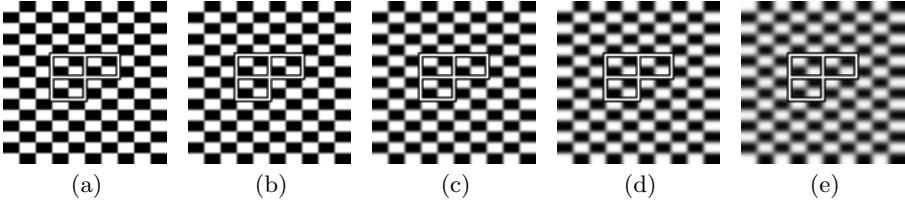


Fig. 2. Synthetic images with blur variations and the estimated texel size (60×40 in all cases). The blur filter is applied (a) 2 times, (b) 4 times, (c) 8 times, (d) 16 times, (e) 32 times.

image, making difficult the texel detection. This filter has been applied 2, 4, 8, 16 and 32 times. Resulting images are shown in Fig. 2. In this figure, the detected texel is highlighted twice in each direction for comparison purposes. As we can see, the texel is accurately detected despite the blur effect in all the images, so we can infer that these blur levels do not affect the performance and accuracy of our approach.

Salt and Pepper Noise. In order to evaluate the performance of our approach with noise, we have corrupted the same periodic texture pattern with salt and pepper noise in different occupancy levels. Image noise is usually regarded as undesirable, which is why it is important to evaluate our approach under noisy conditions. Occupancy levels of noise considered in tests are 5%, 10%, 20%, 40% and 80%, but in order to extend the test we have randomly built 100 images for each occupancy level. These 500 images are used as inputs to our method. A sample of the aspect of the resulting images is shown in Fig. 3, where the texel detected is also shown for each image. Results in terms of percentage of correctly detected texels show that our approach detects properly the texel size

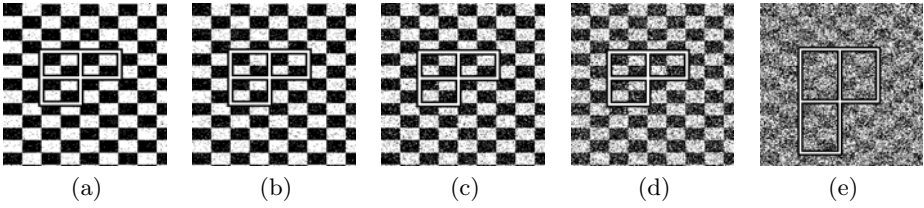
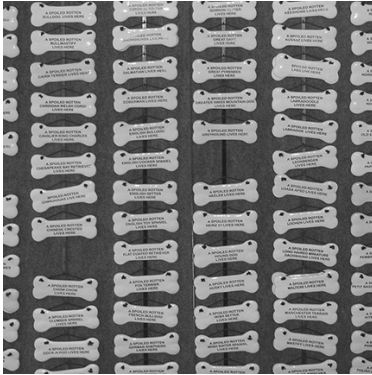
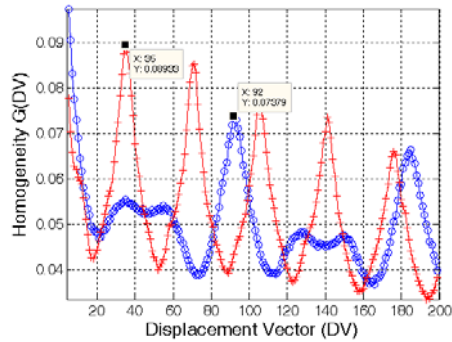


Fig. 3. A sample of the image corrupted with each noise occupancy level and the texel size detected by our method. Noise occupancy levels are (a) 5%, (b) 10%, (c) 20%, (d) 40%, (e) 80%.



(a)



(b)

Fig. 4. A natural texture (a) and its homogeneity function (b) in both directions horizontal (o) and vertical (+)

with 5% and 10% noise occupancy in the 100 images. With occupancy levels of 20% and 40% an error occurs in only 6 images. With 80% of noise occupancy, in 62 times the texels were accurately detected. Most of the errors are related with the detection of multiples of the texel size, as we can see in Fig. 3e, where our system detects a texel size of 60×80 .

3.2 Texel Size Estimation of Natural Images

After tests on synthetic images, we have evaluated the performance of our approach using near-periodic natural images. Near-periodic textures are those that are not strictly periodic, showing irregularities in color, intensity, noise, global or local deformations, resolution, etc. [20] presenting a challenge for any system. Texel size of a periodic texture pattern is estimated by finding the first maximum, but it is more difficult when a natural texture is analyzed (see Fig. 4). In order to generalize our method, we say that texel size is estimated by the V where the global maximum from homogeneity functions is found.

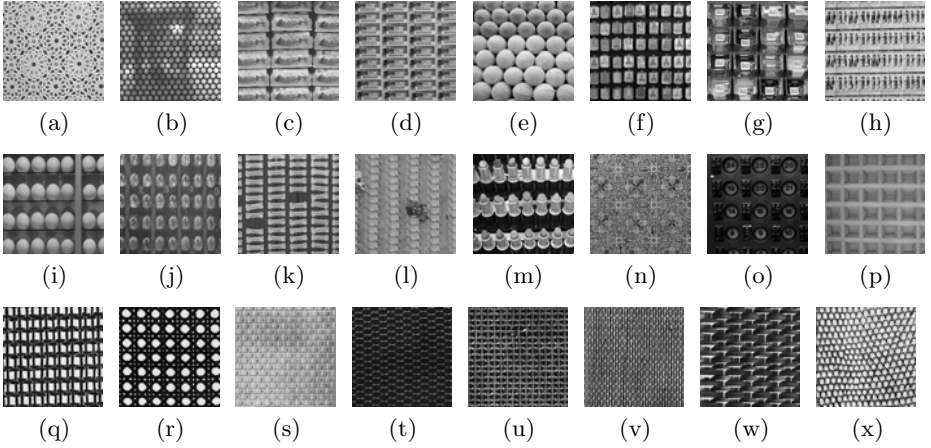


Fig. 5. Natural images set used in experiments

We have evaluated the performance of our method using a set of 16 natural images extracted from the album proposed by Klette [7] and 8 images from the Brodatz album [1]. At a first glance, these 24 textures (see Fig. 5) seem to be periodic. Nevertheless, a thorough inspection shows that the texture periodicity descriptors (texel size, texel shape and texel placement rules) and the intensity properties (illumination, blur, contrast, noise), vary through the image because of its natural origin.

Table 1 presents the results of texel size estimation using our method for each image. In order to evaluate the goodness of our method, we have implemented a simple texture synthesis by using a tiling algorithm. In this algorithm, a sample is tiled up to fill an image of the same dimensions of the original texture. Some qualitative results are shown in Fig. 6, where the original image is given, highlighting the detected texel. These results show that even when a tiling algorithm seems not appropriate for near-periodic textures, the accuracy in our texel detection allows to both textures, original and synthetic, seem not identical but quite similar.

A quantitative evaluation is carried out with a cosine similarity measure in order to quantify the similarity between the original image and the synthetic image. Cosine similarity measure is a common vector similarity metric, that guarantees, as do other similarity metrics, that $0 \leq \cos(h_1, h_2) \leq 1$, so we can have an intuitive result where 1 is the value for two images that match exactly. Cosine similarity metric is computed as: $\cos(h_1, h_2) = \frac{\sum_k h_1(k)h_2(k)}{\sqrt{\sum_k h_1^2(k) \sum_k h_2^2(k)}}$, where h_1 and h_2 are representative histograms of two images. In this paper, each image is represented by the Sum and Difference Histograms with different arbitrary displacement vectors. Therefore, we have two similarity values and the final result is the average of these two values.

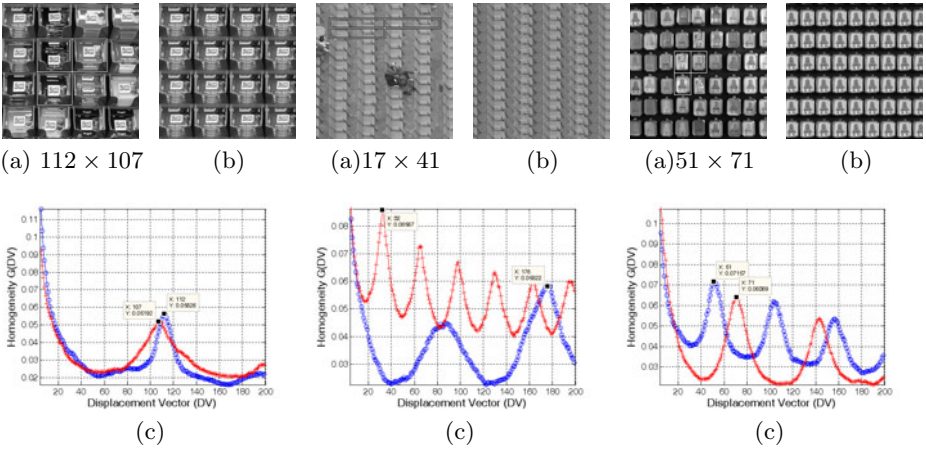


Fig. 6. Some qualitative results. (a) Original image with the estimated texel overlaid, (b) Synthetic texture resulting from the tiling of the detected texel, and (c) corresponding homogeneity plot for the image in (a).

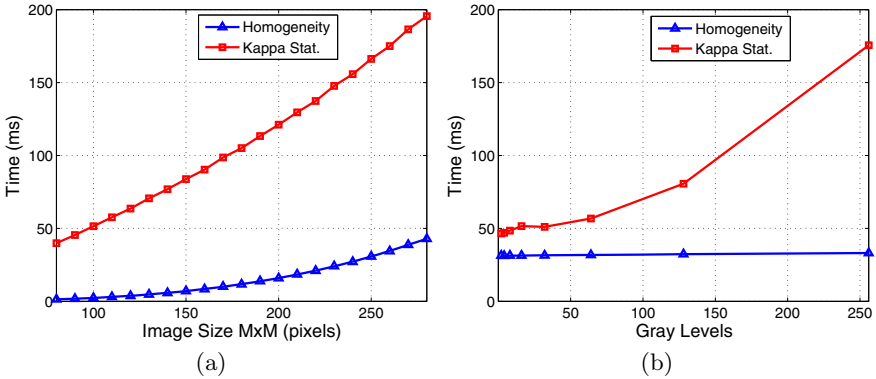


Fig. 7. Computation time of the proposed algorithm and the Kappa statistics for (a) Different Image Sizes and (b) Varying Gray Levels in the image

Results of cosine similarity measure are also presented in Table I. These values show that the synthetic image is very similar to the original one with an average value of 0.8879 (88.8% similar). Synthetic images corresponding with (a), (c), (d), (f), (j), (k), (l), (n), (q), (r) and (u), exhibit good similarity values, higher than 0.9, this is, images are similar in more that 90%. The lowest value of 0.704 (marked in Table I in bold) is obtained with the image (i). In general, there are slight differences between the original and synthetic images, due to natural irregularities of texture.

Table 1. Texel size determination for textures in Fig. 5 and the similarity measure for the corresponding synthesized texture

Texture	Texel Size	Similarity	Texture	Texel Size	Similarity
(a)	194×198	0.967	(m)	56×137	0.846
(b)	27×47	0.880	(n)	151×153	0.913
(c)	180×78	0.947	(o)	123×97	0.842
(d)	117×47	0.973	(p)	81×78	0.846
(e)	84×132	0.829	(q)	33×33	0.915
(f)	51×71	0.892	(r)	38×38	0.990
(g)	112×107	0.920	(s)	25×31	0.872
(h)	22×95	0.825	(t)	41×17	0.877
(i)	64×117	0.704	(u)	32×29	0.973
(j)	110×91	0.951	(v)	14×19	0.805
(k)	92×35	0.918	(w)	69×28	0.875
(l)	176×32	0.980	(x)	15×31	0.770

3.3 Computation Time Evaluation

In this section, we have compared in computation time our algorithm with different well-known approaches. The tested approaches are: a CM-based method using κ statistics, an auto-correlation method and the method based on Renyi's generalized entropies. We have evaluated these methods in their computation time dependance on (1) Image size and (2) Gray levels in an image. For each method in both tests, we obtain the average computation time of 100 executions.

For the first experiment (see Fig. 7a) we have tested square images of varying sizes of $M \times M$ pixels, with $M \in [80, 280]$, and in the second experiment (see Fig. 7b) we have tested images of 256×256 pixels size with varying the gray levels K with $K \in \{2, 4, 8, 16, 32, 64, 128, 256\}$. As can be seen in the figures, the time curves are noticeably different. Our approach time consumption is considerably lower than the CM-based method in both cases, depending on the image size and depending on the gray levels. We can observe that the corresponding curve for the CM-based method increases more and faster than the curve for our method. Results of auto-correlation-based and Renyi's generalized entropies-based methods were obtained but they have not been plotted because they are out of scale. These methods are time-consuming, they are measured in minutes for an image of 256×256 pixels size.

All of our data were obtained by using non-optimized C implementations on an ordinary Intel(R) Core(TM)2 Duo 3.05GHz CPU with 2GB of RAM.

4 Summary and Conclusions

Texel size detection is a classical problem in structural texture analysis. In this paper, the use of homogeneity cues to detect the texel size in periodic, corrupted

periodic and near-periodic textures has been discussed. Homogeneity was presented as a function of a displacement vector, which determines the histogram to be used. When the displacement vector matches the texel size, the homogeneity reaches its maximum value. With this in mind, we can easily detect the basic pattern repeated along a specific texture surface. Natural textures lose periodicity because of usual surface irregularities. However, homogeneity function still has local maxima corresponding to the texel size and multiples.

The algorithm, yet simple, is a robust method with respect to blur distortions and noise corruption near to 80%. This robustness is also shown using texture synthesis with near-periodic natural textures as inputs. In most cases, we obtain a good similarity index between the original image and the synthesized one. Other advantage of the proposed method is the ability of detection of period in both horizontal and vertical directions. Therefore, we can easily detect both square and rectangular texels. This approach, is fast enough to be considered for practical applications, since it takes 0.015s to detect the texel size in a 200×200 pixels size and 256 gray levels image with a non-optimized implementation.

Acknowledgments

R.A. Lizarraga-Morales acknowledges the Mexican CONACyT and CONCyTEG for the financial support via scholarships, grant numbers 206622 and 08-16-K119-139, respectively.

References

1. Brodatz, P.: Textures: A Photographic Album for Artists and Designers. Dover Publications, New York (1966)
2. Forsyth, D.: Shape from texture without boundaries. In: Heyden, A., Sparr, G., Nielsen, M., Johansen, P. (eds.) ECCV 2002. LNCS, vol. 2352, pp. 225–239. Springer, Heidelberg (2002)
3. Grigorescu, S., Petkov, N.: Texture analysis using Renyi's generalized entropies. In: Proc. of the IEEE (ICIP 2003), vol. 1, pp. 241–244. IEEE, Los Alamitos (2003)
4. Gui, Y., Ma, L.: Periodic pattern of texture analysis and synthesis based on texels distribution. *The Visual Computer* 26(6-8), 951–964 (2010)
5. Haralick, R.: Statistical and Structural Approaches to texture. In: Proc. on the IEEE 4th. Int. Joint Conf. Pattern Recognition, pp. 45–60 (1979)
6. Jan, S.R., Hsueh, Y.C.: Window-size determination for granulometrical structural texture classification. *Pattern Recogn. Lett.* 19(5-6), 439–446 (1998)
7. Klette, R.: Basic multimedia imaging (2002), http://www.cs.auckland.ac.nz/~rklette/TeachAuckland.html/mm/Pictures/220Textures_RK.zip
8. Lee, K.L., Chen, L.H.: A new method for extracting primitives of regular textures based on wavelet transform. *Int. J. of Patt. Recogn. and Artif. Intell.* 16, 1–25 (2002)
9. Leu, J.G.: On indexing the periodicity of image textures. *Image and Vision Computing* 19(13), 987–1000 (2001)

10. Lin, W.C., Hays, J., Wu, C., Liu, Y., Kwatra, V.: Quantitative Evaluation of Near Regular Texture Synthesis Algorithms. In: IEEE Computer Society Conf. on Computer Vision and Pattern Recognition , vol. 18(5), pp. 427–434 (2006)
11. Liu, Y., Lin, W.C., Hays, J.: Near-regular texture analysis and manipulation. In: SIGGRAPH 2004, pp. 368–376. ACM Press, New York (2004)
12. Lizarraga-Morales, R.A., Sanchez-Yanez, R.E., Ayala-Ramirez, V.: Optimal spatial predicate determination of a local binary pattern. In: Proc. of the (VIIP 2009), pp. 41–46. Acta Press (2009)
13. Lobay, A., Forsyth, D.: Recovering shape and irradiance maps from rich dense texton fields. In: Proc. of the (CVPR 2004) , pp. 400–406 (2004)
14. Menegaz, G., Franceschetti, A., Mecocci, A.: Fully automatic perceptual modeling of near regular textures. In: SPIE Human Vision and Electronic Imaging XII, vol. 6492, pp.64921B.1–64921B.12. SPIE, San Jose (2007)
15. Ngan, H.Y.Y., Pang, G.K.: Regularity analysis for patterned texture inspection. IEEE Trans. on Automation Science and Engineering 6(1), 131–144 (2009)
16. Oh, G., Lee, S., Shin, S.Y.: Fast determination of textural periodicity using distance matching function. Pattern Recogn. Lett. 20(2), 191–197 (1999)
17. Selkainaho, K., Parkkinen, J., Oja, E.: Comparison of χ^2 and κ statistics in finding signal and picture periodicity. In: Proc. 9th Int. Conf. Patt. Recogn., pp. 1221–1224 (1988)
18. Todorovic, S., Ahuja, N.: Texel-Based texture Segmentation. In: Proc. of the (ICCV 2009), pp. 841–848 (2009)
19. Unser, M.: Sum and difference histograms for texture classification. IEEE Trans. on Pattern Anal. Mach. Intell. 8(1), 118–125 (1986)
20. Liu, Y., Tsin, Y., Lin, W.C.: The Promise and Perils of Near-Regular Texture. Int. J. of Computer Vision 62(1-2), 145–159 (2005)
21. Zhang, J., Tan, T.: Brief review of invariant texture analysis methods. Pattern Recognition 35, 735–747 (2002)
22. Zhu, S., Guo, C., Wang, Y., Xu, Z.: What are Textons? Int. J. of Computer Vision 62, 121–143 (2005)

Adaptive Thresholding Methods for Documents Image Binarization

Bilal Bataineh¹, Siti N.H.S. Abdullah², K. Omar³, and M. Faizul³

Center for Artificial Intelligence Technology
Faculty of Information Science and Technology
Universiti Kebangsaan Malaysia, 43600, Bangi, Selangor, Malaysia
¹bilal_bataineh82@yahoo.com, {²mimi, ³ko,mfn}@ftsm.ukm.my

Abstract. Binarization process is easy when applying simple thresholding method onto good quality image. However, this task becomes difficult when it deals with degraded image. Most current binarization methods involve complex algorithm and less ability to recover important information from a degradation image. We introduce an adaptive binarization method to overcome the state of the art. This method also aims to solve the problem of the low contrast images and thin pen stroke problems. It can also enhance the effectiveness of solving all other problems. As well as, it does not need to specify the values of the factors manually. We compare the proposed method with known thresholding methods, which are Niblack, Sauvola, and NICK methods. The results show that the proposed method gave higher performance than previous methods.

Keywords: binarization, document image, thresholding method, local binarization.

1 Introduction

This binarization of document images is a necessary step in pre-processing stage of document analysis applications. However, Kefali et al. claimed that the aim of binarization is to reduce unwanted information to increase the visibility of the desired information [1]. The process of binarization divides the values of pixels in the image into two levels such as black pixels represent as the foreground whereas the white pixels represent as the background. Based on previous studies [1-2], the binarization techniques are classified into two ways. Firstly, it involves hybrid or complex algorithms based on compound steps and existing techniques [3] and secondly, it applies simple or automatic thresholding methods to determine the thresholding value [4]. Comparing to both forms, the simple methods are easier to design and implement, also they give higher performance in different cases. In general, the simple thresholding methods are classified into two categories [1-2]: local thresholding methods [4] and global thresholding methods [5]. The local thresholding methods determine different thresholding values based on region of interest of an image. On the other hand, the global methods determine a single thresholding value for the whole image.

Kefali et al. has conducted an assessment about twelve outstanding methods on historical Arabic document images [1]. Similar to above competition, they used 150

images containing different problems. The result showed that Nick and Sauvola methods achieved the first and second place respectively. Both the Nick and Sauvola fall into simple local thresholding method. They claimed that Nick method could perform extremely good, because it is based on thresholding shifting value whilst Sauvola's method worked better on solving the binarization noise. This initial study is one of the rare studies that focused on the historical Arabic documents.

Apart from that, Stathis et al. have written a deep evaluation paper based on a binarization competition [2]. At the same time, they also proposed a new evaluation technique namely pixel error. The competition involved about 30 well-known algorithms from different categories such as global, local and hybrid. Each method was tested on 150 different level of degraded document images. The competition results indicated that Sauvola method was outperformed on maximum intensity document images, whereas Johansen method obtained the best accuracy on normal intensity document images. As a result, this competition has given a clear view on the binarization methods categories and performance. Furthermore, we can also conclude that most prominent recent methods are only sufficient to tackle specific image cases.

As conclusion, we can assume that simple method outperform compared to hybrid or complex method. Unlike others, the simple method does not require high complexity cost. Nevertheless, it is an independence method that does not require other process in advance. Also, it is easy to construct and implement. On the other view, the simple thresholding methods can extraordinarily outperform if they the image includes pre-processing. Otherwise, the simple global thresholding approach can become ineffective on degraded images such as poor quality, illumination inconsistency and scanning error [1, 3].

Generally, performance of binarization process affects document image properties. Ntogas have summed up those challenges as: dirty spots, poor quality, low contrast between text and background (Fig.1 (a)), multi-color, thin strokes of pen, ink seeping from other documents and multi-size text [6] (Fig.1 (b, c)). In general, each method can deal well with some of these challenges but fails with the other.

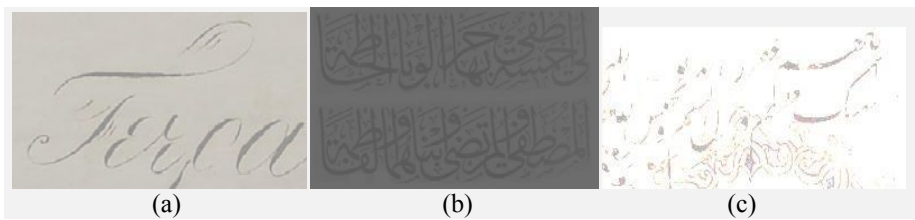


Fig. 1. (a) is a thin pen stroke and (b, c) a low contrast image

The aim of this work is to propose a binarization method for document images. This method adopts local approach to find the thresholding value of the windows in the document images. It has been achieved by introducing an adaptive thresholding method that able to find automatic thresholding value based on each window. The proposed method aims to solve the low contrast images and thin pen stroke problems. We compare the proposed method with other current existing methods such as Niblack, Sauvola, and NICK methods. We test them using selected images for visual experiments and other

benchmark dataset with evaluation techniques for binarization methods. This paper is organized as follows. Section 2 reviews on the state of art of the most local binarization methods used. Section 3 explains the proposed method, Section 4 presents and analyses the experimental results. Finally, conclusions are presented in Section 5.

2 The State of Art

In this study we emphasize on simple and local thresholding approach. Out of many available techniques, we select only the most outperform and recent methods. They are Niblack, Sauvola and the NICK methods.

2.1 Niblack's Method

This method is proposed by Niblack in 1986 [7]. The thresholding value for each window is determined based on mean, m and standard deviation, σ values of pixels in that window as the following:-

$$T = m + k \times \sigma, \quad (1)$$

where k is -0.2 by suggested by Niblack [7], and the window size are pre-determined by the user. Based on experiments, this method can strongly identify the text body. However, it also generate binarization black noise in empty windows.

2.2 Sauvola's Method

This method has been proposed by Sauvola et al. in 1997 [8]. This approach is actually inherited from Niblack method. It can successfully overcome the black noise problem. The thresholding formula is as following:-

$$T = m \times \left(1 - k \times \left(1 - \frac{\sigma}{R}\right)\right), \quad (2)$$

where k is a control factor in the range of [0.2, 0.5], R is a predetermined image gray-level value. The author suggested $k=0.2$, $R= 125$. Unfortunately, this method is less effective when image contrast between the text and background is relatively small.

2.3 Nick Method

This method is proposed by Khurshid et al. in 2010 [5]. NICK method was developed from Niblack method. It tried to solve low contrast problem by shifting down the thresholding value. The thresholding formula is as the following:-

$$T = m + k \times \sqrt{\frac{(\sum P_i^2 - m^2)}{NP}}, \quad (3)$$

where k is a control factor in the range of [-0.1, -0.2], P_i = the image pixel grey-scale value and NP = the total pixels number in the image. The author suggested the $k = -0.1$ [7]. Kefali et al. [1] claimed that NICK method gave the best performance compared with previous methods. However, low contrast images problem are still remained unsolved.

In general, neither of these methods able to deal with all the problems. The methods deal well on some problems and fail on others. Furthermore, most of recent methods requires manual parameters/factors setting. Prior to this limitation, we introduce an automatic parameter setting for determining adaptive threshold value.

3 The Proposed Method

We aim to solve the problems and weaknesses of the previous. To achieve that, two factors have used in our method. The first one is the mean value, m_g of all image's pixels, the other one is the adaptive standard deviation $\sigma_{Adaptive}$.

In Niblack method, the different properties of each windows led to binarization problem, whereas the binarization noises found in empty window. To solve this problem, the global mean value is used to round the extremist values of windows. In Sauvola and NICK methods, binarization faced problems when the contrast of the image is low. In images representation, the contrast value of images is denoted by the standard deviation σ . If image contrast is low, the standard deviation value will be too small and does not effective in the binarization process. To solve this problem, we adapted the standard deviation values of each image. Which leads to an equal effect regardless of the images contrast value. To present the proposed thresholding method on all gray-scale values, we fixed pixels values in the range of [0,1]. Where, the minimum scale value are 0 and the maximum scale value are 1. Then, we define the proposed method formula as follows:

$$T = m_W - \frac{m_W^2 - \sigma_W}{(m_g + \sigma_W) \times (\sigma_{Adaptive} + \sigma_W)}, \quad (4)$$

where, T is the thresholding value, m_W is the mean value of the widow's pixels, σ_W is the standard deviation of the widow's pixels, m_g is the mean value of all pixels in the image and $\sigma_{Adaptive}$ is the adaptive standard deviation of the window. The adaptive standard deviation method for each window is given by equation (5).

$$\sigma_{Adaptive} = \frac{\sigma_W - \sigma_{\min}}{\sigma_{\max} - \sigma_{\min}}, \quad (5)$$

where $\sigma_{Adaptive}$ is the fixed standard deviation of the window, σ_W is the standard deviation of the window, $\sigma_{\min}$ is the minimum standard deviation value and $\sigma_{\max}$ is the maximum standard deviation value of all windows in the document image. We calculate $\sigma_{Adaptive}$ to represent the most optimum σ value among all windows in an image. This $\sigma_{Adaptive}$ value can relatively change according to the natural of the image. It gives an idea about nature of the image contrast. Based on our experiments, sometimes the standard deviation values σ is insignificant applying on brighter or low contrast images. Therefore in some cases, that drives to inapplicable/ ineffective standard deviation values. For that reason, we adaptive the standard deviation values of each image in the range of [0,1]. Based on this T values, the binarization process is defined in Equation (6).

$$I(x, y) = \begin{cases} \text{black,} & i(x, y) < T_W, \\ \text{white,} & \text{Else} \end{cases} \quad (6)$$

where $I(x, y)$ is the image and $i(x, y)$ is the input pixel value of the image.



Fig. 2. (a) is a thin pen stroke (b) and (c) are low contrast image. (d),(e) and (f) are the binarization results of the Niblack's method, (g), (h) and (i) are the binarization results of the Sauvola's method, (j), (k) and (l) are the binarization results of the NICK method (m), (n) and (i) are the results of the proposed method.

4 The Experiments and Results

We organize the experiments into two phases; training and testing. In the training phase, we observe relationship between factors and the performance of each method on thin pen stroke text and low contrast images. Then, we identify the most optimum R , k and

window size for each method. Some of them are a pre-determined factors by the proposed authors. The values identified as: $k = -0.2$ and 25×25 window size for Niblack method [1], $k = 0.2$, $R = 128$ and 15×15 window size for Sauvola method [8], and $k = -0.2$ and 19×19 window size for Nick method [5] while 20×20 window size for the proposed method. Below in Table 1 the parameters setting for each method. Some of visual results are shown according to Niblack’s method, Sauvola’s method, NICK and proposed method successively in Fig. 2.

Table 1. The factors values of the Niblack, Sauvola, NICK and proposed methods

	Niblack [7]	Sauvola [8]	Nick [5]	proposed
k	-0.2	0.2	-0.2	-
R	-	128	-	-
Window size	25×25	15×15	19×19	20×20

To give a clearer perception of the performance of previous methods, the previous methods have been tested on benchmark dataset and evaluation technique. We test and evaluate existing methods based on benchmark dataset namely Document Image Binarization Contest (DIBCO, 2009), it was prepared for International Conference on Document Analysis and Recognition ICDAR 2009 [9, 10]. This dataset [11] contains 10 document images. These images are color and gray-scale also divided into 5 handwritten and 5 printed document images. These dataset included the general challenges of binarization process. The evaluation technique are based on the *F-mean* measurement, it is available in [12]. As below, the *F-mean* denotes to the percentage of the binary image accourcy.

$$F\text{-mean} = 2 \times \text{Recall} \times \text{Precision} / \text{Recall} + \text{Precision}, \tag{7}$$

where, $\text{Recall} = TP / TP + FN$, $\text{Precision} = TP / TP + FP$ and, TP is the true-positive value, FN is the false-positive value and FP false-negative values.

Three experiments conducted on DIBCO, (2009) dataset [11]. The first experiment conducted based on selected samples containing thin pen stroke and low contrast problems, the selected samples are: H01 and H05. As shown in Table 2 and Fig. 3, the results of the proposed method is better than other methods. The average of the *F-mean* are 82.425% for the proposed method. While 24.883%, 33.342% and 73.6145% for the Niblack, Sauvola, and NICK methods successively.

Table 2. The *F-mean* of the Niblack, Sauvola, NICK and proposed methods of the selected samples

	H01	H05	Average
proposed	82.123%	82.727%	82.425%
Niblack [7]	32.086%	17.68%	24.883%
Sauvola [8]	18.53%	48.154%	33.342%
NICK [5]	71.015%	76.214%	73.6145%

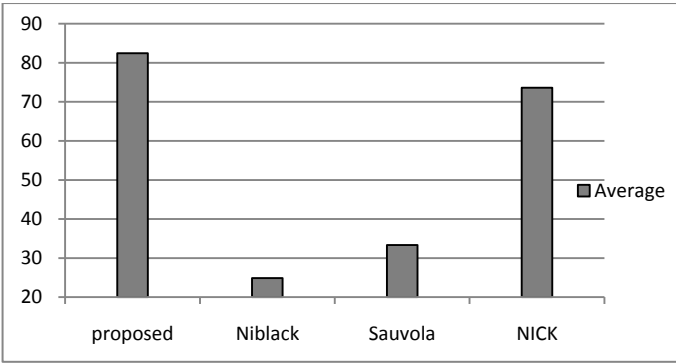


Fig. 3. The *F-mean* average of the proposed, Niblack, Sauvola and NICK methods

To view the performance of methods on all types of challenges. The next experiment was conducted on dataset images. As exhibited in Table 3 and Fig. 4, the results of the proposed method on both hand written or printed written are higher than the other methods. In the printed written category, the highest *F-mean* the Niblack's value was achieved by the proposed method about 83.3416%. while the rest achieved about 83.3416%, 71.7396% and 52.363% to NICK, Sauvola and Niblack methods. Retrieving to previous, we can conclude the average performance for both categories are 84.968%, 80.0546%, 63.25% and 38.97% with respect to the proposed, NICK, Sauvola and Niblack methods subsequently.

Table 3. The *F-mean* of the Niblack, Sauvola, NICK and proposed methods with authers' parameters values

	Hand written F-mean	Printed written F-mean	Average
proposed	82.82%	87.12%	84.97%
Niblack [7]	25.57%	52.36%	38.97%
Sauvola [8]	54.77%	71.74%	63.25%
NICK [5]	76.77%	83.342%	80.1%

For the results of methods' performance without the influence of windows size. In the following experiment, all methods have applied on the same windows size. The 20 ×20 window size was chosen because it is the more optimum windows size to all methods. As exhibited in Table 4 and Fig. 5, the average of the *F-mean* results of the proposed method are higher than the other methods. The average of the *F-mean* is 84.968% for the proposed method. However, 39.5853%, 66.9045% and 80.0546% are *F-mean* results for the Niblack method, Sauvola's method, and NICK method successively.

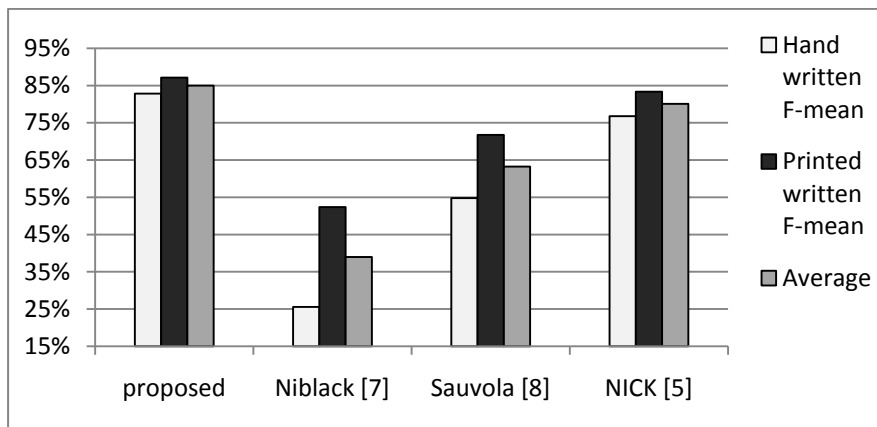


Fig. 4. The F-mean of the proposed, Niblack, Sauvola and NICK methods with authers' parameters values

Also, The proposed method gave the second best performance among each method in each of Recall (the accuracy rate of the text body in the result out image) and Precision (the percentage of binarization noise in the result out image). However, Sauvola's method achieved the best performance in Precision term (97.6%), but it achieved the lowest Recall performance about 55.2%. On the other hand, Niblack method achieved the best performance in Recall term about 91.9%, whereas, Niblack method achieved the lowest Precision performance is 27%. However, the proposed method achieved a proportional values in terms of Recall and Precision are 83.3% and 88.4% successively. Which makes the proposed method gives the best performance in whole binarization process.

Table 4. The F-mean of the Niblack, Sauvola, NICK and proposed methods with 20 ×20 window

	<i>Recall</i>	<i>Precision</i>	<i>F-mean</i>
proposed	83.3%	88.4%	84.968%
Niblack [7]	91.9%	27%	39.585%
Sauvola [8]	55.2%	97.6%	66.905%
NICK [5]	74.5%	89.5%	80.055%

Based on previous experiments, we have found that the proposed method effective with all types of document images. Apart from successfully solving close contrast image problem, the proposed method is also perform better in thin pen stroke text problems. In addition, the proposed method skipped the problem of identifying the factors manually.

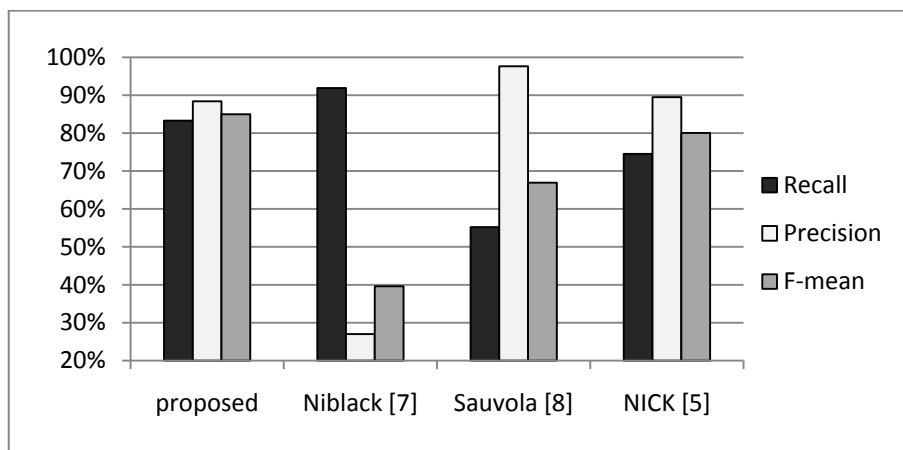


Fig. 5. The F-mean of the proposed, Niblack, Sauvola and NICK methods with 20×20 window

5 Conclusion

The objective of this work is to propose an enhanced binarization method based on local thresholding approach. It presents a new thresholding method that can determine effective threshold values for each windows regions on the document images. The proposed method can deal with all kinds of challenges including the low contrast images and thin pen stroke text. Also, it is skipped the problem of identifying the parameters of the previous methods manually. In the experiments, we have evaluated the proposed method by comparing Niblack's, Sauvola's, and NICK methods. The experiments was conducted on selected documents images and benchmark dataset that dedicated to the problems of the of binarization. In summary, the proposed method has given better performance in comparison to other state of the art methods. Also, it is easy to implement and deals with binarization challenges types.

Acknowledgments. Special thanks are due to Dr. Khurram Khurshid, University Paris Descartes, France, for his assistance and cooperation. This project research was funded by the UKM-TT-03-FRGS0129-2010 grant entitled “Determining adaptive threshold for image segmentation”, UKM-TT-03-FRGS0130 entitled “Automatic Background and Subtraction for Image Enhancement”.

References

1. Kefali, A., Sari, T., Sellami, M.: Evaluation of several binarization techniques for old Arabic documents images. In: The First International Symposium on Modeling and Implementing Complex Systems MISC 2010, Constantine, Algeria, pp. 88–99 (2010)
2. Stathis, P., Kavallieratou, E., Papamarkos, N.: An Evaluation Technique for Binarization Algorithms. *Journal of Universal Computer Science* 14(18), 3011–3030 (2008)

3. Gatos, B., Pratikakis, I., Perantonis, S.J.: Adaptive degraded document image binarization. *Pattern Recognition* 39(3), 317–327 (2006)
4. Otsu, N.: A threshold selection method from gray-level histogram. *IEEE Transactions on Systems, Man and Cybernetics* 9(1), 62–66 (1979)
5. Khurshid, K., Siddiqi, I., Faure, C., Vincent, N.: Comparison of Niblack inspired Binarization methods for ancient documents. In: 16th International conference on Document Recognition and Retrieval. SPIE, USA (2010)
6. Ntogas, N., Ventzas, D.: A Binarization Algorithm For Historical Manuscripts. In: Proceedings of the 12th WSEAS international conference on Communications, Heraklion, Greece, pp. 41–51 (2008)
7. Niblack, W.: An introduction to digital image processing (1985)
8. Sauvola, J., Seppanen, T., Haapakoski, S., Pietikainen, M.: Adaptive document binarization. In: Fourth International Conference Document Analysis and Recognition (ICDAR), Ulm, Germany (1997)
9. Gatos, B., Ntirogiannis, K., Pratikakis, I.: DIBCO 2009: document image binarization contest. *International Journal on Document Analysis and Recognition* (2009)
10. Gatos, B., Ntirogiannis, K., Pratikakis, I.: ICDAR 2009 Document Image Binarization Contest. In: 10th International Conference on Document Analysis and Recognition, Beijing, China (2009)
11. Document Image Binarization Contest (DIBCO 2009), National Center for Scientific Research. Demokritos, Greece (September 2010), <http://www.iit.demokritos.gr/~bgat/DIBCO2009/benchmark>
12. Document Image Binarization Contest (DIBCO 2009), National Center for Scientific Research. Demokritos, Greece (September 2010), <http://users.iit.demokritos.gr/~bgat/DIBCO2009/Evaluation.html>

Foveated ROI Compression with Hierarchical Trees for Real-Time Video Transmission

J.C. Galan-Hernandez^{1,*}, V. Alarcon-Aquino¹, O. Starostenko¹,
and J.M. Ramirez-Cortes²

¹ Department of Computing, Electronics, and Mechatronics
Universidad de las Americas Puebla
Sta. Catarina Martir, Cholula, Puebla. C.P. 72810, Mexico
{juan.galanhz,vicente.alarcon}@udlap.mx

² Department of Electronics
Instituto Nacional de Astrofisica, Optica y Electronica
Tonantzintla, Puebla Mexico

Abstract. Region of interest (ROI) based compression can be applied to real-time video transmission in medical or surveillance applications where certain areas are needed to retain better quality than the rest of the image. The use of a fovea combined with ROI for image compression can help to improve the perception of quality and preserve different levels of detail around the ROI. In this paper, a fovea-ROI compression approach is proposed based on the Set Partitioning In Hierarchical Tree (SPIHT) algorithm. Simulation results show that the proposed approach presents better details in objects inside the defined ROI than the standard SPIHT algorithm.

Keywords: Compression, Fovea, ROI, SPIHT, Wavelet Transforms.

1 Introduction

Video and image compression can help in reducing the communication overhead. Lossy compression is a common tool for achieving high compression ratios; however, more information from the image is lost as the compression rate increases. Compression algorithms based on regions with different compression ratios are important for applications where is needed to preserve the details over a particular object or area. Given a ratio compression n , such algorithms isolate one or several regions of interest (ROI) from the background and then the background is compressed at higher ratios than n while all ROIs are compressed at lower ratios than n achieving a better reconstruction of the ROIs. Standards like MPEG4 and JPEG2000 define an operation mode using ROIs.

The proposed approach for ROI coding over real-time video transmission is to take advantage of the structure of the human retina, called fovea, for increasing

* The authors gratefully acknowledge the financial support from the National Council of Science and Technology and the Puebla State Government, under the contract no. 109417.

the quality of the perception of each reconstructed frame while maintaining a high data quality over the ROI. Such ROI is defined by a motion detection algorithm. This approach is based on the use of a Lifting Wavelet Transform and a modified version of the SPIHT algorithm that allows to define Foveated areas of the image.

1.1 Previous Works

Proposals for wavelet based fovea compression are presented in [1]-[2]. The idea of these approaches is to modify the continuous wavelet transform that decimate the coefficients using a weight function. Another approach using fovea points over a wavelet is discussed in [3]. Instead of using a Fovea operator over the Continuous Wavelet Transform (CWT), a quantization operator $q(x)$ is applied to each coefficient of the discrete wavelet transform (DWT). Such quantization operator is defined by a weight window. Figure 1 depicts the results from both methods applied to the image *lenna*. Figure 1a shows the results of the foveated continuous wavelet transform applied to the image. Figure 1b shows the results of foveation by applying a quantized operator applied to the DWT coefficients of the image. It can be seen that the CWT-based fovea approach shows a softer behavior in the image (especially in the upper right corner) than the DWT-based fovea algorithm.



(a) Foveated Wavelet Transform using CWT (b) Quantized Wavelet Coefficients using DWT

Fig. 1. Different foveating methods using wavelet transforms

The Set Partitioning In Hierarchical Tree algorithm (SPIHT) does not allow to define ROIs. In [2] and [4], different proposals for ROI compression with the SPIHT algorithm are presented. In this paper we report a fovea-ROI compression approach based on a modified version of the SPIHT algorithm. The remainder of this paper is organized as follows. Section 2 reports a description of classical video

compression. In Section 3 an overview of foveated compression is given. Section 4 describes the SPIHT algorithm. Section 5 reports the proposed approach, Section 6 presents results and Section 7 reports conclusions and future work.

2 Video Compression

Lately, video coding has evolved into two dominant standards: MPEG and ITU-T H.26x. Such recommendations are based on a classic video encoding framework [5] shown in figure 2.

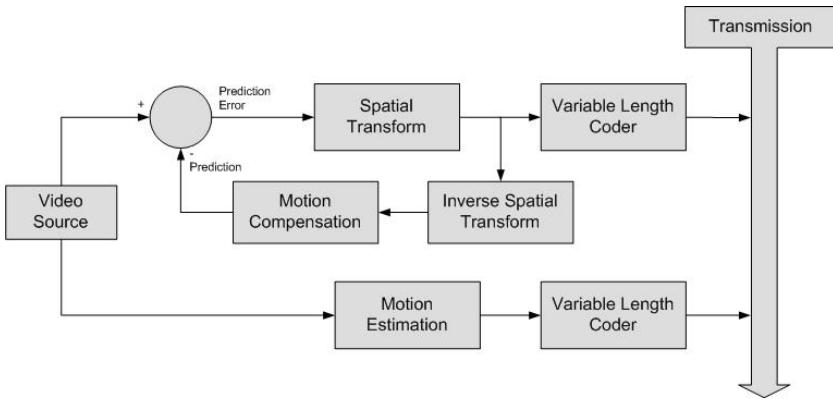


Fig. 2. Classic Video Encoding Framework

Two main parts in video compression shown in figure 2 are spatial transform and motion estimation. Spatial transformation is successively applied to individual video frames to take advantage of the high degree of data correlation in adjacent image pixels (spatial correlation).

Motion estimation exploits temporal correlation. Classic video coding takes the difference from two frames $e_n = f_{n-1} - f_n$ where f_i is the video frame i and e is called Motion Compensation Error Residual (MCER) [6]. Usually, a video sequence does not change but only in small segments from frame to frame. Using the MCER instead of the original frame reduces the amount of data to be transmitted because MCER will have more redundant data (zeros) overall.

However, the use of motion estimation adds an accumulative error over the coding because the coder uses the original frames for calculating the MCER and the decoder only have the decoded frames that, when used lossy compression algorithm, are not a perfect reconstruction of the original frame. For improving the quality of the compression, a feedback from the video encoded frames are used for calculating a motion compensation vector. Such motion compensation can be calculated either from the coder alone such in classic compression or by the coder using feedback from the decoder such in Distributed Video Coding (DVC) [78].

3 Foveated Compression

Foveated images are images which have a non-uniform resolution [1]. Researchers have demonstrated that the human eye experiments a form of aliasing from the fixation point or fovea point to the edges of the image [9]. Such aliasing increases in a logarithmic rate on all directions. This can be seen as concentric cutoff frequencies from the fixation point. Foveated images have been exploited in video and image compression. The use of fovea points yields reduced data dimensionality, which may be exploited within a compression framework. A foveated image can be represented by [10]

$$I_0(x) = \int I(t)C^{-1}(x)s\left(\frac{t-x}{w(x)}\right)$$

where $I(x)$ is a given image and $I_0(x)$ is the foveated image. The function s is called the weighted translation of s by x . There are several weighted translation functions, such the ones defined in [11].

For wavelets, foveation can be applied in both the Wavelet Transform [1], and the wavelet coefficients [12]. Given a foveation operator T with a weight function $w(x) = \alpha|x|$, and a smooth function $g(x)$ with support on $[-\alpha^{-1}, \alpha^{-1}]$, a 2D wavelet transform is defined by

$$\theta_{j,m,k,n} = \langle T\psi_{j,m}, \psi_{k,n} \rangle = \int_{-\infty}^{\infty} \int_{-\infty}^{\infty} \psi_{j,m}(t)\psi_{k,n}(x) \frac{1}{|x|} g\left(\frac{t-x}{\alpha|x|}\right) dt dx$$

where $\{\phi_{l_0,n}\}_{0 \leq n \leq 2^{l_0}} \cup \{\psi_{j,n}\}_{j < l_0, 0 < n < 2^{-j}}$ define an orthonormal wavelet basis, $\phi_{l_0,n}(\cdot)$ and $\psi_{j,n}(\cdot)$ represent scaled and translated versions of the mother wavelet $\psi(\cdot)$.

4 SPIHT

The algorithm called Set Partitioning In Hierarchical Trees (SPIHT) is a compression scheme based on wavelets proposed in [13]. It has important properties such as high compression ratios and progressive transmission.

The SPIHT is based on bit-plane encoding and takes advantage of a property wavelet coefficient. When a one level wavelet decomposition is applied to an image, four bands are obtained: an LL band or approximation coefficients band, and three detail coefficients called HL , HH , LH . Higher levels of decomposition yields into more detail coefficients HL_n , LH_n and HH_n where n is the level of decomposition in which the sub band belongs. If $C(\cdot)$ is a set of wavelet coefficients from a wavelet decomposition of the image I and L is the level of decomposition, there is a relation [14] between a coefficient $C(i, j)$ with $C(i, j) \in HL_n \cup LH_n \cup HH_n$ and $1 < n \leq L$ and the coefficients $C(2i, 2j)$, $C(2i, 2j + 1)$, $C(2i + 1, 2j)$ and $C(2i + 1, 2j + 1)$. $C(i, j)$ is known as a parent and $C(2i, 2j)$, $C(2i, 2j + 1)$, $C(2i + 1, 2j)$ and $C(2i + 1, 2j + 1)$ coefficients are called *Offsprings*. Applying recursively such property from the highest band L to $L - 1$ yields into a hierarchical tree known as quadtree with root $C(i, j)$.

Given a threshold T if all coefficients from a quadtree are lower than T , such quadtree is called a zerotree [15]. Zerotrees are common in wavelet decompositions and the SPIHT exploits such property for compression. If a quadtree is a zerotree, the SPIHT only outputs a zero instead of sending bits for each coefficient of the zerotree. The SPIHT defines three lists: list of insignificant pixels (LIP), list of insignificant sets (LIS) and list of significant pixels (LSP). LIP stores the position of each pixel that are lower than a given threshold, LIS stores the position of each root of all zerotrees for a given threshold and LSP stores the position of all coefficients higher than a given threshold. Such lists are used for the two main steps of the algorithm: significance pass and refinement pass.

The significance pass checks all elements $C(i, j)$ with $(i, j) \in LIS$, if $|C(i, j)|$ is higher than a threshold T_l outputs a 1 followed by the sign of $C(i, j)$ and (i, j) are deleted from LIS and stored into LSP, also, a matrix of thresholds W_Q is updated with $W_Q(i, j) = T_l$. Then, checks all the coefficients from quadrees that its root $(i, j) \in LIS$. If a quadtree is not a zerotree, at least one of the coefficients $|C(i', j')|$ that belongs to that quadtree is higher than the threshold T_l . If $C(i', j') \in HL_\delta \cup LH_\delta \cup HH_\delta$ with $1 \leq \delta \leq L$, each coefficients $C(k, m) \in HL_\phi \cup LH_\phi \cup HH_\phi$ with $1 \leq \phi < \delta$ are classified and its position inserted into LIS if $|C(k, m)|$ are lower than T_l or if $|C(k, m)|$ is higher than T_l the significance pass outputs a 1 and its sign, the matrix of thresholds W_Q is updated with $W_Q(k, m) = T_l$ and inserted into LSP. All positions (k, m) where $C(k, m) \in HL_\delta \cup LH_\delta \cup HH_\delta$ are also stored in LIS if $C(k, m) < T_l$.

In the refinement pass, with a given threshold T_l , each coefficient $C(i, j)$ with $(i, j) \in LSP$ is evaluated. If $|C(i, j)| \in [W_Q(i, j), W_Q(i, j) + T_l)$ then outputs a 0 else if $|C(i, j)| \in [W_Q(i, j) + T_l, W_Q(i, j) + 2T_l)$ then outputs a 1 and $W_Q(i, j) = T_l$.

SPIHT is defined as a five steps algorithm:

1. Initialization. The threshold $T = 2^{\lfloor \log_2(\max(|C(i, j)|)) \rfloor}$ with $C(i, j) \in LL \cup HL_n \cup LH_n \cup HH_n$ and $1 \leq n \leq L$. Each $(i, j) \in LL \cup HL_L \cup LH_L \cup HH_L$ is inserted into LIP and each $(i, j) \cup HL_L \cup LH_L \cup HH_L$ is inserted into LIS.
2. Significance pass
3. Refinement pass
4. $T = T/2$
5. Return to 2

The algorithm can be stopped either on an arbitrary value of T or if a bit per pixel (bpp) ratio is met for the output.

5 Proposed Approach

Our proposal is to mix fovea compression with ROI compression using the SPIHT algorithm. In this paper this algorithm is referred to as FVHT (Fovea Hierarchical Trees), which is a modified version of the SPIHT. The FVHT is applied to individual frames of a video stream for a real-time video transmission. The mix

of both will yield into a better quality perception of each individual frame and preserve information around the ROI that can be useful for an observer instead of only making the ROI bigger. The wavelet decomposition is calculated with the Lifting Wavelet Transform (LWT) [16]. The LWT uses the lifting scheme and factorizes orthogonal and biorthogonal wavelet transforms into elementary spatial operators called liftings. The advantage of the LWT is three reduced operations to nearly two for its calculation [17]. The block diagram of the proposed approach is depicted in figure 3.

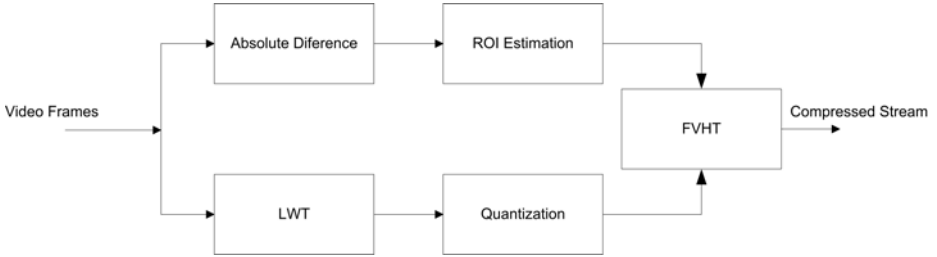


Fig. 3. Block diagram of the proposed approach

Given a video transmission of a moving object with frames $F_0, F_1, \dots, F_n$ taken by a steady camera. The proposed algorithm is described in the following steps:

1. If $i \neq 0$ then $ROI_i = F_i - F_{i-1}$
2. Calculate the centroid c_i of the ROI if any
3. Calculate the wavelet decomposition W_i of F_i using the LWT.
4. W_i is quantized into integers generating the new coefficients set W_i^q
5. FVHT is applied to W_i^q and outputs the resultant bit stream
6. Return to step 1 until no more frames are left.

The first step of the proposed algorithm obtains the absolute difference between the last frame and the current one for motion detection, other algorithms may also be considered if better accuracy or motion compensation is needed. Then, if a moving object is detected by the pixel difference, the centroid and size is estimated. This will be the ROI of the frame determined by step 2. The next step calculates the LWT wavelet coefficients of the current frame. The LWT is used because of its low computational complexity in order to meet a good overall performance. The following step will quantize the wavelet coefficients as integers with a fixed q . Then, FVHT is applied to the quantized coefficients. Note that the FVHT algorithm is a proposed modified version of SPIHT that allows fovea regions.

The proposed algorithm is intended for a DVC scheme [78] where no feedback is allowed from the decoder or for IP cameras. Over DVC without feedback,

exploiting motion compensation error residual is unreliable because the displaced frame difference cannot be calculated either by the coder or the decoder. Without displaced frame difference, an accumulative error is added over time on the decoding step reducing the quality of the decoded frames over time [6]. So, time correlation is not included in the proposed algorithm. However, if feedback is possible, the proposed algorithm can be easily implemented over a classic video scheme.

5.1 Fovea Hierarchical Trees

As stated previously, Fovea Hierarchical Trees is a modified version of SPIHT that compress using fovea regions. The algorithm is fed with the coefficients, ROI centroid and radius, fovea decaying length and a monotonic increasing function $g(x)$ with $g : R \rightarrow (b, L]$ with b as the lowest bit rate and L is the highest bit rate of the compression that defines how the compression bit rate will increase as the pixels move farther from the ROI centroid, the resultant decompressed image will have L bpp.

On each pass of the algorithm, a distance function $D(i, j)$ of each coefficient position is evaluated and used to determine the bit rate of encoding using the decaying function $g(D(i, j))$. If the current bit rate is lower than $g(D(i, j))$ then the coefficient is encoded, otherwise it is discarded. Each quadtree will be evaluated besides the distance of the root to avoid loss of information if an element of the quadtree should still be encoded besides its root distance. The distance is evaluated on each pass in order to not increase the memory usage of the algorithm. The resultant image will have a bit-rate of L .

The distance evaluation should be done in both the significance pass and the refinement pass. However, on the significance pass, the positions of the coefficients are discarded from the *LIP* and in the refinement pass are discarded from the *LSP*. The list *LIS* will remain the same as in the SPIHT algorithm.

6 Results

The proposed approach was implemented with a logarithmic decaying function and an arbitrary uniform scalar quantization $\delta = 0.01$ and a biorthogonal wavelet 9/7 with five levels of decomposition using the LWT as in the standard JPEG2000 which is considered as benchmark [18], other wavelets may also be considered. In figure 4 is shown a comparative analysis of a frame of the video sequence “walk” compressed using the SPIHT and the FVHT algorithms with different bit rates.

Figure 4a is the original frame with the fovea area marked with a white circle, figure 4b, 4c and 4d are the frame compressed with SPIHT at 0.06bpp ratio, 1bpp ratio and 3bpp ratio respectively. Figure 4e is the frame compressed with FVHT with a logarithmic decaying function from 0.06 bpp to 1 bpp with a centroid in (256, 256) and a ROI area of 50px and a fovea decaying of 50px. Figure 4f is the frame compressed with FVHT with a logarithmic decaying function

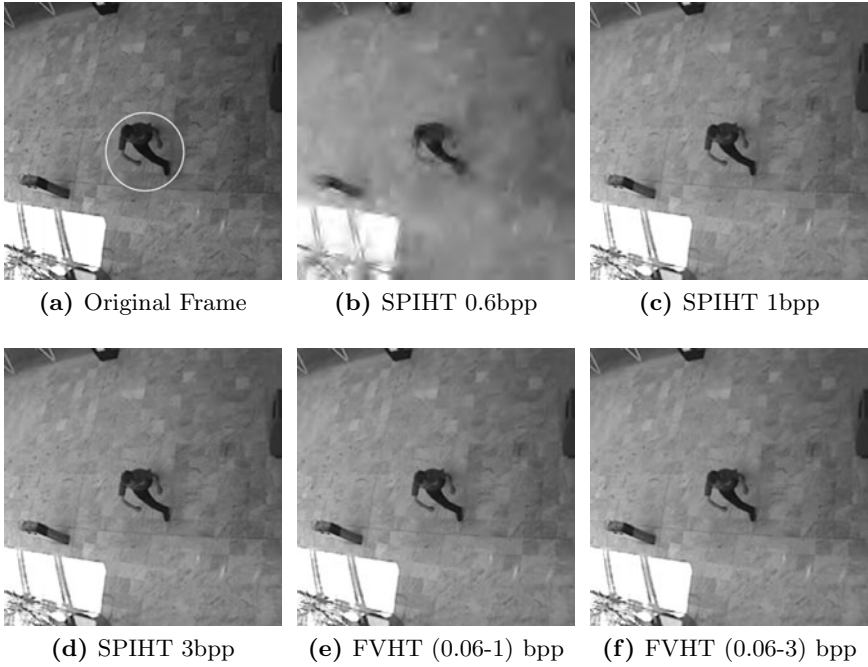


Fig. 4. Comparison of a frame from “walk” sequence with two compression algorithms (SPIHT and FVHT) and several bpp

Table 1. PSNR comparison of different areas of the frame walk with two compression algorithms (SPIHT and FVHT) and several bpp

Algorithm	bpp	25x25 px	50x50 px	75x75 px	100x100 px	125x125 px
SPIHT	3bpp	48.5818	49.0957	49.9044	50.0628	51.1757
SPIHT	1bpp	41.3977	42.1692	43.3924	43.6584	44.8385
FVHT	0.06-3bpp	41.5309	42.2027	43.4104	43.6687	44.8451
FVHT	0.06-1bpp	41.3977	42.1692	43.3924	43.6584	44.8385

from 0.06 bpp to 3 bpp with a centroid in (256, 256) and a ROI area of 50px and a fovea decaying of 50px. Figures 4e and 4f present better detail in objects inside the defined ROI area like the floor texture. The same texture is less detailed as the pixels are farther from the ROI, while the images compressed with SPIHT, 4b, 4c and 4d, presents the same level of details over all. In table 1 a comparative of peak to noise signal ratio (PSNR) for the different compressions used in figure 4 is shown.

Table 1 shows the PSNR of the different compression methods and bpp against the original video frame at different steps. The first step takes a square of the decompressed frame of 25×25 pixels dimension with centroid at the fovea center

(256, 256) and compares it with a same squared region of the original video frame. Each step increases the dimensions of the compared region by 25. The proposed method shows a lower PSNR than the classic SPIHT compression. This is due to the fact that PSNR penalizes heavily the compression resolution drop from the fovea. However, PSNR cannot reflect the antialiasing effect that occurs on the human eye with a fovea compression [1]. Such antialiasing can be perceived in figure 4 as an increased image quality when examining it directly with the eyes fixed at the fovea center.

7 Conclusions and Future Work

Fovea compression together with ROI allows to control the bit rate compression of given areas and preserve image perception quality to the human eye. Compression with hierarchical trees can be further improved as described in [2] by labeling beforehand each coefficient and using unbalanced quadrees, however it will decrease the memory performance of the algorithm. Future work will be focused on defining and evaluating different distance functions and estimation of individual bpp on different regions of the foveated image as well as determining a better metric for measuring the quality of a foveated compression method. Other compression algorithms may also be considered for reducing the high memory usage of the reported algorithms.

References

1. Chang, E.C., Yap, C.K.: Wavelet Approach to Foveating Images. In: Proceedings of the thirteenth annual symposium on Computational geometry - SCG 1997, pp. 397–399 (1997)
2. Cuhadar, A., Tasdoken, S.: Multiple arbitrary shape ROI coding with zerotree based wavelet coders. In: Proceedings of the IEEE International Conference on Multimedia and Expo, ICME 2003, pp. 157–160. IEEE, Los Alamitos (2003)
3. Galan-Hernandez, J.C., Alarcon-Aquino, V., Starostenko, O., Ramirez-Cortes, J.M.: DWT Foveation-Based Multiresolution Compression Algorithm. *Research in Computing Science*, 197–206 (2010)
4. Park, K.-H., Park, H.W.: Region-of-interest coding based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology* 12(2), 106–113 (2002)
5. Bovik, A.: *The Essential Guide to Video Processing*. Academic Press, London (2009)
6. Hanzo, L., Cherriman, P.J., Streit, J.: *Video Compression and Communications*. John Wiley & Sons, Ltd, Chichester (2007)
7. Girod, B., Aaron, A., Rane, S., Rebollo-Monedero, D.: Distributed Video Coding. *Proceedings of the IEEE* 93, 71–83 (2005)
8. Martinez, J.L., Weerakkody, W.A.R.J., Fernando, W.A.C., Fernandez-Escribano, G., Kalva, H., Garrido, A.: Distributed Video Coding using Turbo Trellis Coded Modulation. *The Visual Computer* 25(1), 69–82 (2008)
9. Silverstein, L.D.: *Foundations of Vision*. *Color Research & Application* 21, 142–144 (2008)

10. Ciocoiu, I.B.: ECG signal compression using 2D wavelet foveation. In: Proceedings of the 2009 International Conference on Hybrid Information Technology - ICHIT 2009, vol. 13, pp. 576–580 (2009)
11. Bovik, A.C.: Fast algorithms for foveated video processing. *IEEE Transactions on Circuits and Systems for Video Technology* 13(2), 149–162 (2003)
12. Galan-Hernandez, J.C., Alarcon-Aquino, V., Starostenko, O., Ramirez-Cortes, J.M.: Wavelet-Based Foveated Compression Algorithm for Real-Time Video Processing. In: Robotics and Automotive Mechanics Conference 2010 IEEE Electronics, september 2010, pp. 405–410. IEEE, Los Alamitos (2010)
13. Said, A., Pearlman, W.: A new, fast, and efficient image codec based on set partitioning in hierarchical trees. *IEEE Transactions on Circuits and Systems for Video Technology* 6(3), 243–250 (1996)
14. Tsai, P.: Tree Structure Based Data Hiding for Progressive Transmission Images A Review of Related Works. *Fundamenta Informaticae* 98, 257–275 (2010)
15. Shapiro, J.: Embedded image coding using zerotrees of wavelet coefficients. *IEEE Transactions on Signal Processing* 41, 3445–3462 (1993)
16. Sweldens, W.: The Lifting Scheme: A Custom-Design Construction of Biorthogonal Wavelets. *Applied and Computational Harmonic Analysis* 3, 186–200 (1996)
17. Mallat, S.: *A Wavelet Tour of Signal Processing: The Sparse Way*, 3rd edn. Academic Press, London (2008)
18. Acharya, T., Tsai, P.S.: *JPEG2000 Standard for Image Compression*. John Wiley & Sons, Inc., Hoboken (2004)

Neural Networks to Guide the Selection of Heuristics within Constraint Satisfaction Problems

José Carlos Ortiz-Bayliss, Hugo Terashima-Marín,
and Santiago Enrique Conant-Pablos

Tecnológico de Monterrey, Campus Monterrey
Monterrey, Mexico, 64849

`jcobayliss@gmail.com`, `terashima@itesm.mx`, `sconant@itesm.mx`

Abstract. Hyper-heuristics are methodologies used to choose from a set of heuristics and decide which one to apply given some properties of the current instance. When solving a Constraint Satisfaction Problem, the order in which the variables are selected to be instantiated has implications in the complexity of the search. We propose a neural network hyper-heuristic approach for variable ordering within Constraint Satisfaction Problems. The first step in our approach requires to generate a pattern that maps any given instance, expressed in terms of constraint density and tightness, to one adequate heuristic. That pattern is later used to train various neural networks which represent hyper-heuristics. The results suggest that neural networks generated through this methodology represent a feasible alternative to code hyper-heuristic which exploit the strengths of the heuristics to minimise the cost of finding a solution.

Keywords: Constraint Satisfaction, Neural Networks, Hyper-heuristics.

1 Introduction

A Constraint Satisfaction Problem (CSP) is defined by a set of variables X , where each variable is associated a domain D of values subject to a set of constraints C [31]. The goal is to find a consistent assignment of values to variables in such a way that all constraints are satisfied, or to show that a consistent assignment does not exist. CSPs belong to the NP-Complete class [10] and there is a wide range of theoretical and practical applications like scheduling, timetabling, cutting stock, planning, machine vision, temporal reasoning, among others (see for example [9, 14, 17]).

Several deterministic methods to solve CSPs exist [18,29], and solutions are found by searching systematically through the possible assignments to variables, guided by heuristics. It is a common practice to use Depth First Search (DFS) to solve CSPs [26]. When using DFS to solve CSPs, every variable represents a node in the tree and the deeper we go in the tree, the larger the number of variables that have already been assigned a feasible value. Every time a variable

is instantiated, a consistency check occurs to verify that the current assignment does not conflict with any of the previous assignments given the constraints within the instance. When an assignment produces a conflict with one or more constraints, the instantiation must be undone, and a new value must be assigned to that variable. When the feasible values decrease to zero, the value of a previously instantiated variable must be changed, this is known as backtracking [2]. Backtracking always goes up one single level in the search tree when a backward move is needed. Backjumping is another powerful technique for retracting and modifying the value of a previously instantiated variable and goes up more levels than backtracking in the search tree. Another way to reduce the search space is using constraint propagation, where the idea is to propagate the effect of one instantiation to the rest of the variables due to the constraints among the variables. Thus, every time a variable is instantiated, the values of the other variables that are not allowed due to the current instantiation are removed.

The general idea in this investigation is to combine the strengths of some existing heuristics to generate a method that chooses among them based on the features of the current instance. Hyper-heuristics are methods that choose from a set of heuristics and decide which one to apply given some properties of the instances. Because of this, they seem to be a suitable technique to implement our idea. Different approaches have been used to generate hyper-heuristics (see for example: [1], [4] and [21]) and they have proven to achieve promising results for many optimization problems such as scheduling, transportation, packing and allocation.

This paper is organized as follows. Section 2 presents a brief description of previous studies related to this research. Section 3 describes the methodology used in our solution model. The experiments and main results are presented in Sect. 4. Finally, Sect. 5 presents the conclusions and future work.

2 Background and Related Work

When using neural networks to solve CSPs, it is common to convert the CSP into an optimization problem, where the task of the network is to minimise a non-negative function that vanishes only for solutions [13]. Tsang and Wang [30] described a neural network approach called GENET for solving CSPs with binary constraints through a convergence procedure. Nakano and Nagamatsu [19] proposed a Lagrange neural network for solving CSPs where, in addition to the constraints, each CSP has an objective function. Jönsson and Södenberg developed a neural network approach for solving boolean CSPs and later the same approach was extended to more general CSPs [13].

Even though the term hyper-heuristic was first introduced by Denzinger et al. [7] in 1997, the idea of combining heuristics goes back to 1960s ([8], [6]). Surveys on hyper-heuristic methodologies can be found in [4], [24], and [5]. One of the first attempts to systematically map CSPs to algorithms and heuristics according to the features of the problems was presented in [29]. In that study, the authors presented a survey of algorithms and heuristics for solving CSPs and they

proposed a relation between the formulation of the CSP and the most adequate solving method for that formulation. More recently, Ortiz-Bayliss et al. [20] developed a study about heuristics for variable ordering within CSPs and a way to exploit their different behaviours to construct hyper-heuristics by using a static decision matrix to select the heuristic to apply given the current state of the problem. More studies about hyper-heuristics applied to CSPs include the work done by Terashima-Marín et al. [28], who proposed an evolutionary framework to generate hyper-heuristics for variable ordering in CSPs; and Bittle and Fox [3] who presented a hyper-heuristic approach for variable and value ordering for CSPs based on a symbolic cognitive architecture augmented with case based reasoning as the machine learning mechanism for their hyper-heuristics. The difference between these two approaches lies in the learning method and the set of heuristics used.

3 Solution Model

This section presents the proposed solution model in detail. It describes the problem state representation, the neural network and the way in which the networks are trained and used to code the hyper-heuristics.

3.1 Problem State Representation

For this research we have included only binary CSPs. A binary CSP contains unitary and binary constraints only. Rossi et al. [25] proved that for every general CSP there is an equivalent binary CSP. Thus, all general CSPs can be reduced into a binary CSP. To represent the problem state we propose to use two important binary CSPs properties known as constraint density (p_1) and constraint tightness (p_2). The constraint density is a measure of the proportion of constraints within the instance; the closer the value of p_1 to 1, the larger the number of constraints in the instance. A value of $p_1 = 0.5$ indicates that half of the nodes present a constraint among them. The constraint tightness (p_2) represents a proportion of the conflicts within the constraints. A conflict is a pair of values $\langle x, y \rangle$ that is not allowed for two variables at the same time. The higher the number of conflicts, the more unlikely an instance has a solution. A CSP instance with $p_2 = 1$ is trivially insoluble because all pairs of values in the constraints are not allowed. In contrast, an instance with $p_2 = 0$ does not contain any conflicts and can be solved very easily because all the pairs of values between variables are allowed. We used these two measures to represent the problem state. Our idea is that these two features can be used to describe a CSP instance and to create a relation between instances and heuristics.

The CSP instances used for this research are randomly generated in two stages. In the first stage, a constraint graph G with n nodes is randomly constructed and then, in the second stage, the incompatibility graph C is formed by randomly selecting a set of edges (incompatible pairs of values) for each edge

(constraint) in G . The instance generator receives four parameters: $\langle n, m, p_1, p_2 \rangle$. The number of variables is defined by n and the uniform domain size by m . The parameter p_1 determines how many constraints exist in a CSP instance and it is called constraint density, whereas p_2 determines how restrictive the constraints are and it is called constraint tightness. More details on the framework for problem instance generation can be found in [22] and [27].

Every time a variable is assigned a new value and the infeasible values are removed from domains of the remaining uninstantiated variables, the values of p_1 and p_2 change and a sub-problem with new features appears. This is the reason why we decided to use the constraint density and tightness to represent the problem state and guide the selection of the low-level heuristics. Our idea is that these two features can be used to describe a CSP instance and to create a relation between instances and heuristics.

3.2 Variable Ordering Heuristics

A solution to any given CSP is constructed selecting one variable at the time based on one of the four variable ordering heuristics used in this investigation: Rho, Max-Conflicts (MXC), Minimum Remaining Values (MRV) and Expected Number of Solutions ($E(N)$). Each one of these heuristics orders the variables to be instantiated dynamically at each step during the search process. These heuristics are briefly explained in the following lines.

- The Rho heuristic is based on the approximated calculation of the solution density ρ . This measure considers that, if a constraint C_i prohibits in average a fraction p_c of possible assignments, a fraction $1 - p_c$ of assignments is allowed. Then, the average solution density ρ is the average fraction of allowed assignments through all the constraints. If independence between the constraints is supposed, then ρ is defined as [11]:

$$\rho = \prod_{c \in C} (1 - p_c). \quad (1)$$

The basic idea with the Rho heuristic is to select the variable that enters in the subproblem which contains the largest fraction of solution states. This is, the subproblem with the largest solution density.

- MXC selects the variable that is involved in the larger number of conflicts among the constraints in the instance. The assignment will produce a subproblem that minimises the number of conflicts among the uninstantiated variables.
- MRV is one of the most simple and effective heuristics to determine which variable to instantiate [12,23]. This heuristic selects the variable with the less number of available values in its domain. The idea consists basically in taking the most restricted variable from those which have not been instantiated yet and by doing so, reducing the branching factor of the search.

- $E(N)$ selects the variable in such a way that the subproblem maximizes the expected number of solutions [11]. This heuristic will maximize the size of the subproblem so as the solution density. The value of $E(N)$ is calculated as:

$$E(N) = \prod_{x \in X} |D_x| \times \rho. \quad (2)$$

We have also used Min-Conflicts [16,15] as value ordering heuristic to improve the search. The selection of the value, when using Min-Conflicts, prefers the value involved in the minimum number of conflicts [15]. This heuristic will try to leave the maximum flexibility for subsequent variable assignments. Min-Conflicts is not considered in the selection process of heuristics because it is a value ordering heuristic. In this investigation, Min-Conflicts is used as a complement of the four variable ordering heuristics to improve the overall performance of the model.

3.3 The Training Set

Before applying any neural network approach it is necessary to design the pattern that will be used for training. If the training set is wrong, then we will produce networks which are not useful for the problem. We decide to use a training set that maps every point in the space (p_2, p_1) to one of the four heuristics previously explained. To obtain this set we produced a grid of instances in the range $[0, 1]$ with increments of 0.025 in each dimension. For every point in the grid we generated 30 random instances which were solved using each of the four heuristics. The heuristic with the lower average consistency checks was selected as the best heuristic for those coordinates. Thus, we produced and analysed a grid containing 50430 instances to obtain the training set. The discussed grid represents a ‘rule’ that indicates which heuristic to apply given the properties p_1 and p_2 . The set obtained via this methodology is shown in Fig. 1. In this figure, the best heuristic for each point in the grid (in terms of average consistency checks), is shown. The training set allows us to observe the regions on the space where each heuristic is more suitable than the others. The points in the grid with no mark indicate that there was not a significant difference in the means of the consistency checks of two or more heuristics.

3.4 Neural Networks to Represent Hyper-heuristics

The basic idea behind the proposed hyper-heuristics is that, given a certain instance, a neural network has to decide which variable ordering heuristic to use at each node of the search tree. Every time a variable is instantiated, a new subproblem arises and the properties may differ from the previous instance. The idea is to solve the problem by constructing the answer, deciding which heuristic to apply at each step. The networks used for this research are backpropagation neural networks with a sigmoidal transference function. Also, we have incorporated the momentum to our networks to improve their performance. Each neural

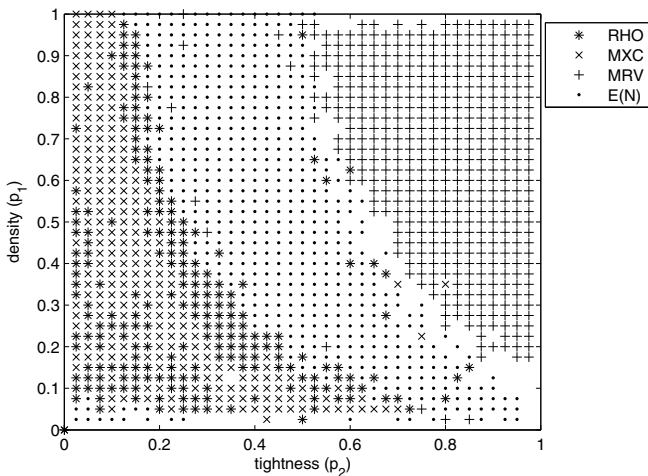


Fig. 1. The training set used for the neural networks

network deals with a simplified problem state described by p_1 and p_2 and uses them as input values; the output of the network is the heuristic to apply at a given time. Once the neural network has been trained using the set from Sect. 3.3 it represents a complete recipe for solving a problem, using a simple algorithm.

Until the problem is solved: (a) determine the current problem state, (b) use the neural network to decide which heuristic to apply, (c) apply the heuristic attached to the state and (d) update the state. This process is shown in Fig. 2.

4 Experiments and Results

The testing set includes 1000 different random instances generated with $n = 20$ and $m = 10$. The instances are uniformly distributed in the space $p_1 \times p_2$, with 10 instances per point. This set forms a grid of instances with increments of 0.1 in each axis, starting from instances with $(p_2 = 0.1, p_1 = 0.1)$ up to instances with $(p_2 = 1, p_1 = 1)$.

We obtained 20 hyper-heuristics as the result of running 20 times the solution model explained in Sect. 3 and those hyper-heuristics were tested with all the instances in the Testing Set. All the networks were generated with two input neurons (p_1 and p_2), two hidden layers and four neurons in the output layer (one for each heuristic). The number of neurons in each hidden layer was randomly decided at the moment the network was created and lied within the range $[5, 15]$. The learning rate and momentum were also randomly selected at generation time. We decided to randomize these parameters to obtain networks with different topologies and observe the differences in the results.

The 20 hyper-heuristics were compared with the average result of the simple heuristics in terms of consistency checks. A consistency check occurs every time

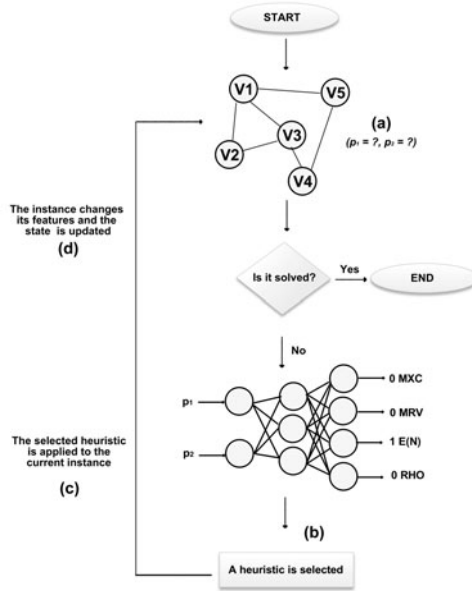


Fig. 2. Process of applying the hyper-heuristic

a constraint must be verified and it is a common measure of the quality of the CSP solving methods. The results of the performance of the hyper-heuristics are shown in Table 1. In this table, $W(mean)$ is a the proportion of instances where the hyper-heuristic performs at least as well as the mean result of the simple heuristics. For the cases where the hyper-heuristic is better than the mean result of the simple heuristics, $W(mean)$ does not provide any information about the percentage of the reduction of consistency checks. This information is presented with $R(mean)$, which indicates the mean reduction in the number of consistency checks with respect to the mean result of the heuristics for every instance in the testing set.

The results suggest that the hyper-heuristics produced with our model represent a feasible solution method for solving CSPs. We have proven that any of these hyper-heuristics will behave at least as well as the mean result of the simple heuristics for a large proportion of instances (no less than 77.2%). In the case of NEHH17, the maximum value of R is achieved with 87.1%. In terms of reduction, NHH08 is the better choice because it is able to reduce the consistency checks of the mean result of the heuristics in 85.3%, in average, for every instance in the testing set.

As an additional result, we tested NHH08 and NHH17 against the best result of the simple heuristics. NHH08 and NHH17 obtained values of $W(Best)$ of 63.0% and 65.8%, respectively. The decrease in the value of W when compared

Table 1. Performance of the hyper-heuristics when compared against the mean result of the simple heuristics

HH	$W(mean)$	$R(mean)$	HH	$W(mean)$	$R(mean)$
NHH01	78.800%	17.777%	NHH11	78.200%	16.086%
NHH02	85.900%	23.739%	NHH12	86.700%	23.522%
NHH03	87.300%	25.195%	NHH13	77.200%	14.331%
NHH04	78.800%	15.148%	NHH14	79.900%	18.234%
NHH05	79.400%	17.627%	NHH15	85.900%	23.534%
NHH06	77.300%	13.246%	NHH16	86.700%	23.371%
NHH07	80.400%	16.471%	NHH17	87.100%	24.540%
NHH08	85.300%	25.257%	NHH18	77.700%	13.851%
NHH09	80.200%	17.939%	NHH19	85.600%	21.188%
NHH10	79.000%	16.547%	NHH20	86.300%	23.729%

with the best result of the heuristics is not necessarily a bad result. In this case we can identify the best result because we are using a small set of heuristics and random instances with suitable generation parameters, but in practice it is not feasible to try various heuristics for each instance and keep the best result. Our hyper-heuristics are not able to overcome the best heuristic for all the cases but they provide acceptable results for a wide range of instances.

5 Conclusions and Future Work

We have presented a methodology to obtain information from a set of instances to produce a pattern that matches CSPs to heuristics. We used that pattern to produce backpropagation neural networks that decide which heuristic to apply given the features of the instances. These neural networks represent variable ordering hyper-heuristics for CSPs and obtained promising results when compared against the mean result of the simple heuristics. Even when these hyper-heuristics need more work to improve their performance, the preliminary results suggest that neural networks hyper-heuristics provide a feasible method for solving CSPs.

As future work we are interested in adding value ordering heuristics to the selection process of the hyper-heuristic and see its contribution to the performance of the model. We also think it is important to test our approach on real instances. Finally, more work is needed to understand the patterns of heuristics and produce new ways to exploit those patterns to guide the heuristic selection.

Acknowledgments

This research was supported in part by ITESM under the Research Chair CAT-144 and the CONACYT Project under grant 99695.

References

1. Bilgin, B., Özcan, E., Korkmaz, E.E.: An experimental study on hyper-heuristics and exam timetabling. In: *Proceedings of the 6th International Conference on Practice and Theory of Automated Timetabling*, pp. 123–140 (2006)
2. Bitner, J.R., Reingold, E.M.: Backtrack programming techniques. *Commun. ACM* 18, 651–656 (1975)
3. Bittle, S.A., Fox, M.S.: Learning and using hyper-heuristics for variable and value ordering in constraint satisfaction problems. In: *Proceedings of the 11th Annual Conference Companion on Genetic and Evolutionary Computation Conference: Late Breaking Papers, GECCO 2009*, pp. 2209–2212. ACM Press, New York (2009)
4. Burke, E., Hart, E., Kendall, G., Newall, J., Ross, P., Shulenburg, S.: Hyper-heuristics: an emerging direction in modern research technology. In: *Handbook of metaheuristics*, pp. 457–474. Kluwer Academic Publishers, Dordrecht (2003)
5. Chakhlevitch, K., Cowling, P.: Hyperheuristics: Recent developments. In: Cotta, C., Sevaux, M., Sörensen, K. (eds.) *Adaptive and Multilevel Metaheuristics, Studies in Computational Intelligence*, vol. 136, pp. 3–29. Springer, Heidelberg (2008)
6. Crowston, W.B., Glover, F., Thompson, G.L., Trawick, J.D.: Probabilistic and parametric learning combinations of local job shop scheduling rules, p. 117 (1963)
7. Denzinger, J., Fuchs, M., Fuchs, M., Informatik, F.F., Munchen, T.: High performance atp systems by combining several ai methods. In: *Proc. Fifteenth International Joint Conference on Artificial Intelligence (IJCAI 1997)*, pp. 102–107. Morgan Kaufmann, San Francisco (1997)
8. Fisher, H., Thompson, G.L.: Probabilistic learning combinations of local job-shop scheduling rules. In: *Factory Scheduling Conference*, Carnegie Institute of Technology (1961)
9. Freuder, E.C., Mackworth, A.K.: *Constraint-Based Reasoning*. MIT/Elsevier, Cambridge (1994)
10. Garey, M.R., Johnson, D.S.: *Computers and Intractability; A Guide to the Theory of NP-Completeness*. W. H. Freeman & Co., New York (1979)
11. Gent, I., MacIntyre, E., Prosser, P., Smith, B.: T.Walsh.: An empirical study of dynamic variable ordering heuristics for the constraint satisfaction problem. In: *Proceedings of CP 1996*, pp. 179–193 (1996)
12. Haralick, R.M., Elliott, G.L.: Increasing tree search efficiency for constraint satisfaction problems. *Artificial Intelligence* 14, 263–313 (1980)
13. Jönsson, H., Söderberg, B.: An information-based neural approach to generic constraint satisfaction. *Artificial Intelligence* 142(1), 1–17 (2002)
14. Mackworth, A.K.: Consistency in networks of relations. *Artificial Intelligence* 8(1), 99–118 (1977)
15. Minton, S., Johnston, M.D., Phillips, A., Laird, P.: Minimizing conflicts: A heuristic repair method for csp and scheduling problems. *Artificial Intelligence* 58, 161–205 (1992)
16. Minton, S., Phillips, A., Laird, P.: Solving large-scale csp and scheduling problems using a heuristic repair method. In: *Proceedings of the 8th AAAI Conference*, pp. 17–24 (1990)
17. Montanari, U.: Networks of constraints: fundamentals properties and applications to picture processing. *Information Sciences* 7, 95–132 (1974)
18. Nadel, B.A.: Algorithms for constraint satisfaction: a survey. *AI Magazine* 13(1), 32–44 (1992)

19. Nakano, T., Nagamatu, M.: Lagrange neural network for solving csp which includes linear inequality constraints. In: Duch, W., Kacprzyk, J., Oja, E., Zadrozny, S. (eds.) ICANN 2005. LNCS, vol. 3697, pp. 943–948. Springer, Heidelberg (2005)
20. Ortiz-Bayliss, J.C., Özcan, E., Parkes, A.J., Terashima-Marín, H.: Mapping the performance of heuristics for constraint satisfaction. In: IEEE Congress on Evolutionary Computation (CEC 2010), pp. 1–8 (July 2010)
21. Özcan, E., Bilgin, B., Korkmaz, E.E.: A comprehensive analysis of hyper-heuristics. *Intelligence Data Analysis* 12(1), 3–23 (2008)
22. Prosser, P.: An empirical study of phase transitions in binary constraint satisfaction problems. Tech. Rep. Report AISL-49-94, University of Strathclyde (1994)
23. Purdom, P.W.: Search rearrangement backtracking and polynomial average time. *Artificial Intelligence* 21, 117–133 (1983)
24. Ross, P., Marfn-Blazquez, J.: Constructive hyper-heuristics in class timetabling, vol. 2 (September 2005)
25. Rossi, F., Petrie, C., Dhar, V.: On the equivalence of constraint satisfaction problems. In: Proceedings of the 9th European Conference on Artificial Intelligence, pp. 550–556 (1990)
26. Russell, S., Norvig, P.: *Artificial Intelligence A Modern Approach*. Prentice-Hall, Englewood Cliffs (1995)
27. Smith, B.M.: Locating the phase transition in binary constraint satisfaction problems. *Artificial Intelligence* 81, 155–181 (1996)
28. Terashima-Marín, H., Ross, P., Farías-Zárate, C., López-Camacho, E., Valenzuela-Rendón, M.: Generalized hyper-heuristics for solving 2d regular and irregular packing problems. *Annals of Operations Research* 179, 369–392 (2010)
29. Tsang, E.: *Foundations of Constraint Satisfaction*. Academic Press Limited, London (1993)
30. Tsang, E.P.K., Wang, C.J.: A generic neural network approach for constraint satisfaction problems. In: *Neural Network Applications*, pp. 12–22. Springer, Heidelberg (1992)
31. Williams, C.P., Hogg, T.: Using deep structure to locate hard problems. In: *Proc. of AAAI 1992, San Jose, CA*, pp. 472–477 (1992)

Microcalcifications Detection Using PFCM and ANN

A. Vega-Corona², J. Quintanilla-Domínguez^{1,3,*}, B. Ojeda-Magaña^{1,2},
M.G. Cortina-Januchs^{1,3}, A. Marciano-Cedeño¹, R. Ruelas², and D. Andina¹

¹ Group for Automation in Signals and Communications GASC

Technical University of Madrid, 28040 Madrid, Spain

{joelq, januchs}@salamanca.ugto.mx, {a.marciano, d.andina}@gc.ssr.upm.es

² Computational Intelligence Laboratory LABINCO-DICIS,

University of Guanajuato. 36885 Salamanca Guanajuato, Mexico

tono@salamanca.ugto.mx

³ Department of Projects Engineering DIP-CUCEI,

University of Guadalajara. 45101 Zapopan Jalisco, Mexico

benojed@hotmail.com, rruelas@newton.dip.udg.mx

Abstract. This work presents a method to detect *Microcalcifications* in *Regions of Interest* from digitized mammograms. The method is based mainly on the combination of *Image Processing*, *Pattern Recognition* and *Artificial Intelligence*. The *Top-Hat transform* is a technique based on mathematical morphology operations that, in this work is used to perform contrast enhancement of microcalcifications in the region of interest. In order to find more or less homogeneous regions in the image, we apply a novel *image sub-segmentation* technique based on *Possibilistic Fuzzy c-Means* clustering algorithm. From the original region of interest we extract two window-based features, *Mean* and *Deviation Standard*, which will be used in a classifier based on a *Artificial Neural Network* in order to identify microcalcifications. Our results show that the proposed method is a good alternative in the stage of microcalcifications detection, because this stage is an important part of the early *Breast Cancer* detection.

Keywords: Microcalcifications detection and classification, Top-Hat transform, Possibilistic Fuzzy c-Means, Artificial Neural Networks.

1 Introduction

Breast cancer is the most common types of cancer among women all over the world. Early detection continues being a key piece to improve the prognosis and

* The authors wish to thank the National Council for Science and Technology (CONA-CyT), the Group of Automation in Signal and Communications (GASC) of the Technical University of Madrid, Computational Intelligence Laboratory LABINCO-DICIS of University of Guanajuato and Department of Projects Engineering DIP-CUCEI of University of Guadalajara.

survival of breast cancer cases [1]. Currently the most reliable and practical method for early detection and screening of breast cancer is mammography. The presence of *Microcalcifications (MCs)* clusters has been considered as a very important indicator of malignant types of breast cancer and its detection and classification are important to prevent and treat the disease. However, the detection and classification of MCs continue being a hard work due to that, in mammograms there is a poor contrast between MCs and the tissue around them. The MCs are tiny deposits of calcium in breast tissue. The MCs appear in the small clusters of a few pixels with relatively high intensity and closed contours compared with their neighboring pixels. Individual MCs are sometimes difficult to detect because of the surrounding breast tissue, their variation in shape, orientation, brightness and diameter size [11].

This work presents a method based on *Image Processing, Pattern Recognition* and *Artificial Intelligence* for identification of MCs in digitized mammogram images. The method consist in: Image selection, image enhancement by morphological *Top-Hat transform*, data clustering and labeling by a partitional clustering algorithm such as *Possibilistic Fuzzy c-Means (PFCM)*, feature extraction based on window-based features such as *mean* and *standard deviation* and finally a classifier based on an *Artificial Neural Network (ANN)*.

2 ROI Selection

The proposed method has been trained and tested using a set of mammograms extracted from a mini-Mammographic database provided by Mammographic Image Analysis Society (MIAS) [10]. Each mammogram from the database is a 1024×1024 pixels and with a spatial resolution of $200 \mu\text{m}/\text{pixel}$. These mammograms have been reviewed by an expert radiologist and all the abnormalities have been identified and classified. To the place where these abnormalities such as, MCs, have been located is known as, *Region of Interest (ROI)*. In this work, the ROI images with a size of 256×256 were used.

3 Detection of Microcalcifications

3.1 Enhancement of Microcalcifications

In the past several years methodologies to detect and / or classify MCs have been developed, but the detection of MCs continues being a difficult task due to some factors, such as, size, shape, low contrast and low visibility from their surrounding as well as the distribution with respect to their fuzzy morphology [4]. The contrast can be defined as the difference in intensity between an image structure and its background. By combining of morphological operations, several image processing tasks can be performed, but in this work we used morphological operations to achieve contrast enhancement.

With the aim to improve the contrast between the MCs and background in the ROI image, we use a *morphological contrast enhancement*. This morphological contrast enhancement, is an image enhancement technique based on Mathematical Morphology operations known as Top-Hat and Bottom-Hat transforms.

But, in this work we will consider only the Top-Hat transform. In previous works such as, [9], [12], this technique was used to obtain satisfactory results in the MCs detection stage.

A Top-Hat is a residual filter which preserves those features in an image that can fit inside the structuring element and removes those that cannot, in other words the Top-Hat transform is used to segment objects that differ in brightness from the surrounding background in images with uneven background intensity. The Top-Hat transform is defined by the following equation:

$$I_T = I - [(I \ominus SE) \oplus SE] \quad (1)$$

where, I is the input ROI image, I_T is the transformed ROI image, SE is the structuring element, $\ominus$ represents morphological erosion operation, $\oplus$ represents morphological dilation operation and $-$ image subtraction operation. $[(I \ominus SE) \oplus SE]$ is also known as the morphological opening operation.

3.2 Segmentation of Microcalcifications by PFCM

Digital image segmentation has been considered the most important intermediate step in image processing to extract a semantic meaning of the pixels. The main objective of clustering process in order to image segmentation is to find groups of pixels with similar features such as, gray level intensity, or more or less homogeneous groups. The similarity is evaluated according to a distance measure between the pixels and the prototypes of groups, and each pixel is assigned to the group with the nearest or most similar prototype [7].

In this work we used a novel method to segment MCs in ROI images, this method was called image sub-segmentation and it is based on *Possibilistic Fuzzy c-Means (PFCM)* clustering algorithm. This method was proposed in our previous work [7].

The PFCM is a more recent partitional clustering algorithm, which has the advantages of the Fuzzy c -Means (FCM) as well as the Possibilistic c -Means (PCM). The FCM has a constraint that makes it very sensitive to outliers. To solve the problem of constraint of the FCM, Krisnapuram and Keller [6] have developed the clustering algorithm PCM, which allows us to identify the degree of typicality which has a data point with respect to the group that belongs. The PCM has a problem, sometimes the prototypes of clusters coincide generating erroneous partitions of the feature space, for this reason it was not so successful. Pal *et al.* [8] to solve the problems of the clustering algorithms as FCM (outlier sensitivity) and PCM (coincident clusters) proposed a hybridized PFCM clustering model. This algorithm has four parameters (m, η, a y b), where the values of a and b represent a relative importance of membership and typicality values in the computation of prototypes respectively. The parameters m and η

represent an absolute weight of membership value and typicality value, respectively. In order to reduce the effect of outliers, it is advised to set $b > a$ and $m > \eta$.

Proposed approach for detection of MCs by sub-segmentation

1. Get the data vector.
2. Assign a value to the parameters (a, b, m, η) .
3. Segment the image taking into account the number of more representative regions, in this case two: Suspicious region with presence of MCs (S_1) and Normal Tissue (S_2), this region we will consider without the presence of MCs.
4. Run the PFCM algorithm to get:
 - The membership matrix U .
 - The typicality matrix T .
5. Get the maximum typicality value for each pixel.

$$T_{max} = \max_i[t_{ik}], \quad i = 1, \dots, c. \quad (2)$$

6. Select a value for the threshold α .
7. With α and the T_{max} matrix, separate all the pixels into two sub-matrices (T_1, T_2), with the first matrix:

$$T_1 = T_{max} \geq \alpha \quad (3)$$

containing the typical pixels of both regions ($Stypical_1$) and ($Stypical_2$), and the second matrix

$$T_2 = T_{max} < \alpha \quad (4)$$

containing the atypical pixels of both regions ($Satypical_1$) and ($Satypical_2$), in this case the atypical pixels are of the most interest, especially the *atypical* pixels of the (S_1)

8. From the labelled pixels z_k of the T_1 sub-matrix the following sub-regions can be generated

$$T_1 = Stypical_1, \dots, Stypical_i, \quad i = 1, \dots, c. \quad (5)$$

and from the T_2 sub-matrix

$$T_2 = Satypical_{1+i}, \dots, Satypical_{2i}, \quad i = 1, \dots, c. \quad (6)$$

such that each region $S_i, i = 1, \dots, c$ is defined by

$$S_i = Stypical_i \cup Satypical_{i+c}. \quad (7)$$

9. Select the sub-matrix T_1 or T_2 of interest for the corresponding analysis. In the case of this work, T_2 is the sub-matrix of interest.
-

4 Classification of Microcalcifications

4.1 Feature Extraction

MCs appear on mammograms as bright spots. These bright spots are small regions with intensity values higher than their surroundings or background. Spatial domain features includes both shape-related features and window-based features. In this work we applied window-based features [4] [5]. These features are *mean* and *standard deviation*, extracted from the original ROI images within a rectangular window size $n \times m$ centered into position (i, j) . The purpose of applying these characteristics is to distinguish the pixels that correspond to possible MCs of the pixels that correspond to background. The mean and standard deviation are defined by:

$$I_{\mu}(i, j) = \frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m f(i, j) \quad (8)$$

$$I_{\sigma}(i, j) = \left[\frac{1}{n \times m} \sum_{i=1}^n \sum_{j=1}^m (f(i, j) - I_{\mu}(i, j))^2 \right]^{1/2} \quad (9)$$

where, I_{μ} , I_{σ} and $f(i, j)$ represent the mean, deviation standard and the gray level value of a pixel located in (i, j) respectively.

4.2 Classification of Microcalcifications by ANN

Artificial Neural Networks (ANNs) are biologically inspired networks based on the neuron organization and decision making process in human brain [2]. In other words, it is the mathematical model of the brain. ANNs are used in a wide variety of data processing applications where real-time data analysis and information extraction are required. One advantage of ANNs approach is that most of the intense computation takes place during the training process. Once ANNs are trained for a particular task, operation is relatively fast and unknown samples can be rapidly identified in the field. Classification is one of the most frequently encountered decision making task of human activity. A classification problem occurs when an object needs to be assigned into a predefined group or class based on a number of observed attributes related to that object. In this work, we proposed a classifier based on ANN in order to classify the patterns such as MCs class or Normal Tissue class. For this purpose, we applied Multilayer Perceptron (MLP). The MLP is the most used in many practical applications, such as in pattern recognition applications. The functionality of the MLP topology is determined by a learning algorithm, the Back Propagation (BP) [3], based on the method of steepest descent. In the process of upgrading the connection weights, it's the most commonly used algorithm by the ANN scientific community.

5 Methodology and Results

In this work we consider applying our method to each ROI individually in order to show the obtained results by means of a segmented image. We selected ten ROI images from mammograms with dense tissue and the presence of MCs.

Morphological Enhancement. With the aim to detect objects that differ in brightness from the surrounding background, the Top-Hat transform is used. In this case the Top-Hat transform is used to increase the contrast between the MCs and background. Then, we apply the same *SE* at different sizes, 3×3 , 5×5 , 7×7 , to perform Top-Hat transform. The *SE* used in this work is a flat *disk-shaped SE*. Fig. 1 shows the original ROI images processed by Top-Hat transform.

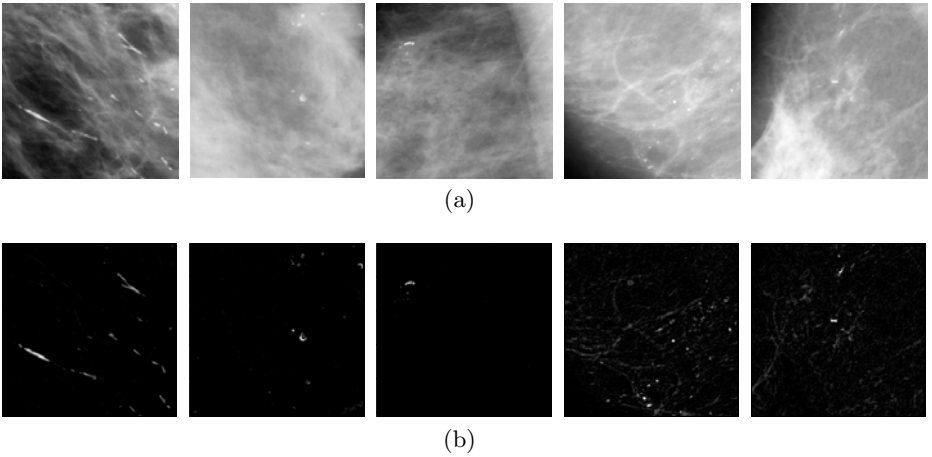


Fig. 1. (a)Original ROIs images (mdb148, mdb170, mdb219, mdb245, mdb249). (b) ROIs images processed by *Top-Hat* transform.

Segmentation of Microcalcifications by PFCM. We consider the ROI images processed by Top-Hat transform to build a *data vector*. In this data vector we apply clustering by PFCM to obtain a label for each pattern belonging to each group of the partition of feature space.

From the point of view of image, with the sub-segmentation we obtain a better identification of possible pixels belonging to MCs, because we found the most atypical pixels of each regions, this is, all those pixels that are away from their prototype, these pixels regularly are very small group with intensity values higher than their surroundings and are very different from the rest of pixels in the ROI image.

Fig. 2 shows the process of image sub-segmentation applying the PFCM, Fig. 2(b) shows suspicious region with presence of MCs and Normal Tissue region, Fig. 2(c) shows each region sub-divided into 4 regions *Stypical*₁, *Satypical*₃ and *Stypical*₂, *Satypical*₄, Fig. 2(d) shows the *atypical*₃ data (pixels) of the

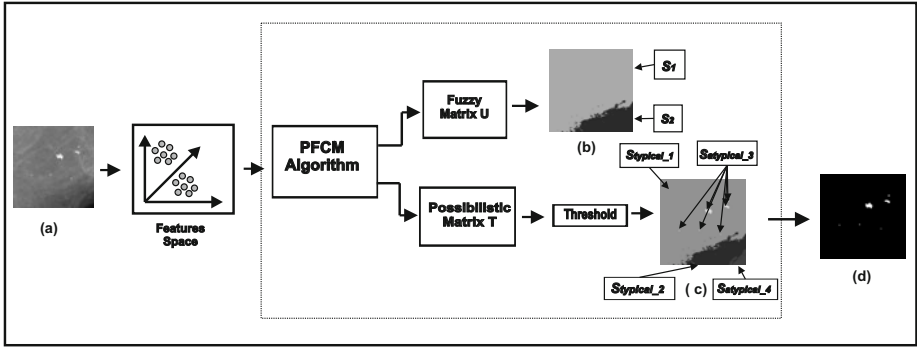


Fig. 2. Process of image sub-segmentation by PFCM

(S_1), which is considered as MCs region, after applying a threshold (α) value. The process of sub-segmentation was performed to 10 ROIs image with the following initial parameters, $a = 1$, $b = 2$, $m = 2$, $\eta = 2$, $\alpha = 0.6$, and we obtained the following values: The number of pixels assigned to the class MCs were 3105 and the number of pixels assigned to the class Normal Tissue were 652255.

Feature Extraction. Two window-based features such as, mean and standard deviation were extracted. These features are extracted from original ROI images within a rectangular windows, in this work we used three different pixel block windows of size 3×3 , 5×5 , 7×7 . In our work, each image obtained by window-based features, are considered as a features to generate a set of patterns that represent MCs and Normal Tissue. This set of patterns we called feature vector (FV). Each of the images used in this work, we know *a priori* that, there are pixels belonging to MCs or Normal Tissue. This FV is considered as a input vector for the classifier. The FV were formed as follow:

$$FV = [I_{\mu_{3 \times 3}}, I_{\sigma_{3 \times 3}}, I_{\mu_{5 \times 5}}, I_{\sigma_{5 \times 5}}, I_{\mu_{7 \times 7}}, I_{\sigma_{7 \times 7}}] \quad (10)$$

The labels of the two classes of the FV were obtained in the previous step by means of sub-segmentation by PFCM. Due to the large amount of patterns of the class that does not belong to MCs respect to the number of patterns that belongs to the class of MCs a balancing was performed, then we obtain two subsets with 3105 patterns to the class MCs and 31050 patterns to the class Normal Tissue.

Classification of Microcalcifications by ANN. In this stage of our method, we used a classifier based on ANN in order to classify the patterns such as Normal Tissue class or MCs class. For this purpose, we applied MLP classifier. We apply different network structures, of which the best obtained results were of the following structure and parameters:

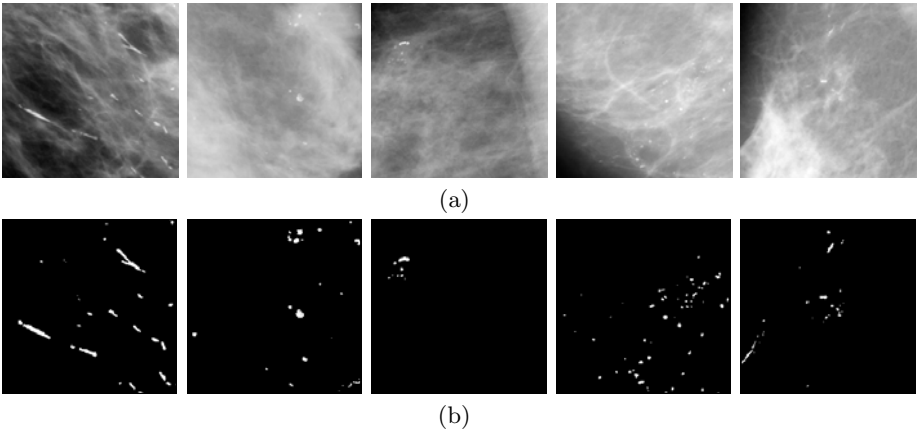


Fig. 3. (a)Original ROIs images. (b)MCs detection using our method.

Table 1. Number of patterns using for training and testing by each classifier

	Numbers of Sample		Total
	Training	Testing	
MCs	2489	616	3105
Normal Tissue	24835	6215	31050

Table 2. Confusion matrix and performance of the classifier

Classifier MLP	Desired Results	Output Results		Sensitivity (%)	Specificity (%)	Total Classification Accuracy (%)
		MCs	Normal Tissue			
<i>Structure</i> ₁ 6 : 24 : 1	MCs	605	11	98.21	99.68	99.54
	Normal Tissue	20	6195			

- 1. Number of input neurons equal to the number of features, for this case 6.
- 2. Number of hidden layers: 1.
- 3. Hidden neurons: 24.
- 4. Output neurons: 1.
- 5. Learning rate η : 1.
- 6. Activation function: sigmoidal with value between [0,1].
- 7. Training conditions, epochs and Mean Squared Error: 250 and 0.001 respectively

We used patterns extracted of the set *FV* in order to train and test our classifier, for this case we used the 80% and 20% of the data randomly selected, respectively. Table 1 shows the distribution of selected data.

Next, we built a confusion matrix to determine the probability of the detection MCs vs. probability of false MCs. Table 2 shows the performance of the classifier presented in this work.

Finally, Fig. 3 shows the obtained results of the MCs detection in the ROIs using the method proposed in this work.

6 Conclusions

The experiment results show that the proposed method can locate MCs in an efficient way, moreover the method promises interesting advances in Medical Field. The analysis of the results served as comparison point with other methods used by other authors mentioned in this work. On the other hand, the use of PFCM was very helpful at the stage of MCs segmentation, because one of the main advantages offered by this algorithm, which is to find typical or atypical data of a region based on the established threshold, thus for case of MCs, the atypical data could be of greater interest than typical data of the same region. Finally, the obtained results show that the implemented method was able to detect MCs satisfactorily, fulfilling the goal of this work.

References

1. World health organization (WHO), <http://www.who.int/en/>
2. Andina, D., Pham, D.: Computational intelligence for engineering and manufacturing, 1st edn. Springer, Heidelberg (2007)
3. Basheer, I.A., Hajmeer, M.: Artificial neural networks: fundamentals, computing, design, and application. *Journal of Microbiological Methods* 43(1), 3–31 (2000)
4. Cheng, H.D., Cai, X., Chen, X., Hu, L., Lou, X.: Computer-aided detection and classification of microcalcifications in mammograms: a survey. *Pattern Recognition* 36(12), 2967–2991 (2003)
5. Fu, J.C., Lee, S.K., Wong, S.T.C., Yeh, J.Y., Wang, A.H., Wu, H.K.: Image segmentation feature selection and pattern classification for mammographic microcalcifications. *Computerized Medical Imaging and Graphics* 29(6), 419 (2005)
6. Krishnapuram, R., Keller, J.M.: A possibilistic approach to clustering. *IEEE Transactions on Fuzzy Systems* 1(2), 98–110 (1993)
7. Ojeda-Magaña, B., Quintanilla-Domínguez, J., Ruelas, R., Andina, D.: Images subsegmentation with the pfc clustering algorithm. 7th IEEE International Conference on Industrial Informatics, 499–503 (2009)
8. Pal, N.R., Pal, S.K., Keller, J.M., Bezdek, J.C.: A possibilistic fuzzy c-means clustering algorithm. *IEEE Transactions on Fuzzy Systems* 13(4), 517–530 (2005)
9. Stojić, T., Reljin, B.: Enhancement of microcalcifications in digitized mammograms: Multifractal and mathematical morphology approach. *FME Transactions* 38, 1–9 (2010)
10. Suckling, J., Parker, J., Dance, D.: The mammographic image analysis society digital mammogram database. *Excerpta Medica International Congress Series* 1069, 375–378 (1994)
11. Wei, L., Yang, Y., Nishikawa, R.M.: Microcalcification classification assisted by content-based image retrieval for breast cancer diagnosis. *Pattern Recognition* 42(6), 1126 (2009)
12. Wirth, M., Fraschini, M., Lyon, J.: Contrast enhancement of microcalcifications in mammograms using morphological enhancement and non-flat structuring elements. 17th IEEE Symposium on Computer-Based Medical System, 134–139 (2004)

Software Development Effort Estimation in Academic Environments Applying a General Regression Neural Network Involving Size and People Factors

Cuauhtémoc López-Martín, Arturo Chavoya, and M.E. Meda-Campaña

Department of Information Systems
University of Guadalajara, México
{cuauhtemoc, achavoya, emeda}@cucea.udg.mx

Abstract. In this research a general regression neural network (GRNN) was applied for estimating the development effort in software projects that have been developed in laboratory learning environments. The independent variables of the GRNN were two size measures as well as a developer measure. This GRNN was trained from a dataset of projects developed from the year 2005 to the year 2008 and then this GRNN was validated by estimating the effort of a new dataset integrated by projects developed from the year 2009 o the first months of the year 2010. Accuracy results from the GRNN model were compared with a statistical regression model. Results suggest that a GRNN could be used for estimating the development effort of software projects when two kinds of lines of code as well as the programming language experience of developers are used as independent variables.

Keywords: Software engineering, software effort estimation, general regression neural network, statistical regression, programming language experience.

1 Introduction

The software process perspectives can be classified as follows [10]: organizations, teams and people. The performance of a software development organization is determined by the performance of its engineering teams, which in turn is determined by the performance of the individual team members, and finally the performance of the latter is, at least in part, determined by the practices these developers follow in doing their work [22]. The levels of software engineering education and training of each developer could be classified in the small and in the large software projects [6]. Software development effort estimation is one of the three main practices used for training developers of small projects at the personal level (the other two are related to software defects and software size [22]). Software development estimation techniques can be classified into two general categories:

1) Expert judgment. This technique implies a lack of analytical argumentation and aims at deriving estimates based on the experience of experts on similar projects; this technique is based on a tacit (intuition-based) quantification step [11].

2) Model-based technique. It is based on a deliberate (mechanical) quantification step [11], and it could be divided into the following two subcategories:

a) Models based on Statistics: Its general form is a linear or nonlinear statistical regression model [3].

b) Models based on computational intelligence: These techniques have emerged in the software engineering field [21], and they include fuzzy logic [15], neural networks [5], genetic programming [4], and other evolutionary algorithms [1].

Based on the assumption of that no single technique is best for all situations and that a careful comparison of the results of several approaches is most likely to produce realistic estimates [2], this study compares the following models with each other: statistical regression and a neural network. Accuracy comparison between neural networks and statistical techniques has been a concern when applying these techniques to several fields such as accounting, finances, health care, medicine, engineering, manufacturing and marketing [18]. The two models were generated from data of small projects developed using practices of the Personal Software Process (PSP) because when this approach was applied to this kind of projects, the PSP has proved useful for delivering quality products on predictable schedules [22]. The models used in this research were generated from a dataset of 156 projects developed by 51 persons from the year 2005 to the year 2008 and then these two models were applied for estimating the development effort of a new dataset consisting of 156 projects developed by 47 persons from the year 2009 to the first months of the year 2010.

1.1 Description of Dependent and Independent Variables

The development effort of the software projects involved in this study was measured in minutes, whereas the size was measured as lines of source code (LOC); in fact, LOC remain in favour in many models [17]. There are two measures of source code size [20]: physical and logical. In this study, physical source lines are considered and all projects were developed based upon a similar coding standard as well as counted by the same counting standard.

The two kinds of physical lines of code that were used for estimating the development effort of this study were New and Changed (N&C) lines of code as well as reused code. Both Added plus Modified code make up the N&C, whereas reused code corresponds to LOC of previously developed projects that are used without modification [9].

Considering that, after software product size, people factors have the strongest influence in determining the amount of effort required in developing a software product [3], the models used in this paper involved an additional independent variable previously considered in [3]: programming language experience, which was measured in months for each developer.

1.2 Criterion for Accuracy Evaluation

The Magnitude of error Relative to the Estimate (MER) is used as criterion for evaluating and comparing the estimation models of this paper. MER was selected

because the Magnitude of Relative Error (or MRE, which has commonly been used as criterion) is not strongly recommended and MER has showed better results than MRE [7]. The MER is defined as follows:

$$MER_i = \frac{|Actual\ Effort_i - Estimated\ Effort_i|}{Estimated\ Effort_i}$$

The MER value is calculated for each observation i whose effort is estimated. The aggregation of MER over multiple observations (N) can be achieved through the Mean MER (MMER) as follows:

$$MMER = (1 / N) \sum_{i=1}^N MER_i$$

The accuracy of an estimation technique is inversely proportional to the MMER. A reference for an acceptable value of Mean of Magnitude of error Relative to the Estimate (MMER) has not been found. In several papers, a Mean of the Magnitude of Relative Error or $MMRE \leq 0.25$ has been considered as acceptable, however, the authors who have proposed this value neither present any reference to studies nor any argumentation providing evidence [12].

1.3 Related Work

Models based upon neural networks and statistical regressions have already been applied for estimating development effort of large software projects [5] [8]. As for neural networks, the feedforward neural network (the most commonly used in the effort estimation field [19]), and a general regression neural network (GRNN) have already been applied for estimating software development effort [13] [16], however, in these two studies the developers' programming language experience was not considered because this independent variable was not found statistically significant in order to use it for comparison with other models.

2 Experimental Design

In this work, collected data were gathered from the same instruments (logs), phases, and standards suggested by the PSP. The experiment was done within a controlled environment having the following characteristics:

- This research involved only graduate students, because previous qualitative analysis have shown that within a PSP course, undergraduate students were more concerned with programming than with the software process issues [14] [23].
- All of the developers were employed and experienced, doing software development in their working environment; however, none of them had taken a course related to personal practices for developing software at the individual level.
- All developers were studying a graduate program related to computer science.
- Each developer wrote seven project assignments (one each day). However, only four of them were selected from each developer. The first three projects were not considered because they had differences in their process phases and logs, whereas in

the last four projects, phases were the same: plan, design, design review, code, code review, compile, testing and post-mortem, and they were based on the same logs.

- Each developer selected his/her own imperative programming language whose coding standard had the following characteristics: each compiler directive, variable declaration, constant definition, delimiter, assign sentence, as well as flow control statement was written in a line of code.

- In all projects the following instruments were used by all developers: project plan summary, defect type standard, time recording log, defect recording log and process improvement proposal. A test report template was introduced from the second to the seventh projects in the testing phase. A code review checklist was introduced from the third to the seventh projects, and design review checklist was used from the fourth to the seventh projects. Thus, from the fourth project on, all the developers used all practices and logs planned for this study. Hence, the first, second and third projects were excluded from this study; otherwise the comparison of the development time results would have been unfair.

- Developers had already received at least one formal course on the object oriented programming language of their choice and they had good programming experience in that language. The sample of this study reduced the bias because it only involved developers whose projects were coded in C++ or Java.

- As this study was an experiment with the aim of reducing bias, the developers were not informed about the experimental goal.

- Developers filled out a spreadsheet for each task and submitted it electronically for examination.

- All of the developers followed the same counting standard.
- Developers were constantly supervised and advised about the process.
- The code written in each project was designed to be reused in subsequent projects.
- The developed projects had complexity similar to that suggested by the original PSP [9] and are described in [15].

3 Description of Estimation Models

The data sample of projects for generating the models had the following characteristics:

1. It considered New and Changed code and Reused code.
2. Programming language experience was considered.
3. Data for all projects were correct, complete, and consistent.

The general regression neural network and the statistical regression were either trained or generated from a dataset of 156 projects developed by 51 persons from the year 2005 to the year 2008.

3.1 Statistical Regression Model

Equation (1) was generated using the actual data from the 156 projects.

$$\text{Effort} = 57.4098 + (1.1 * N\&C) - (0.18 * \text{Reused}) - (0.36 * \text{Programming Language Experience}) \quad (1)$$

The intercept value of 57.40 is the value of the line where independent variables are equal to zero. Signs of each of the three parameters meet with the following assumptions related to the software development:

- 1) The higher the value of new and changed code (N&C), the higher the development effort is.
- 2) The higher the value of reused code, the lower the development effort is.
- 3) The higher the value of programming language experience of developers, the lower the development effort is.

An acceptable value for the coefficient of determination is $r^2 \geq 0.5$ [9]. This equation had an $r^2 = 0.51$. The ANOVA for this equation had a statistically significant relationship between the variables at the 99% confidence level. To determine whether the model could be simplified, a parameter analysis of the multiple regression was done; results from this analysis showed that the highest p-value on the three independent variables was 0.0250, corresponding to reused code. Since this p-value is less than 0.05, reused code is statistically significant at the 95% confidence level (the software tool used was Statgraphics 4.0); consequently, the independent variable of reused code was not removed.

3.2 General Regression Neural Network

A General Regression Neural Network (GRNN) has the following advantages: (a) fast learning and (b) convergence to the optimal regression surface as the number of samples becomes very large. The GRNN has shown that, even with sparse data in a multidimensional measurement space, the algorithm provides smooth transitions from one observed value to another [24].

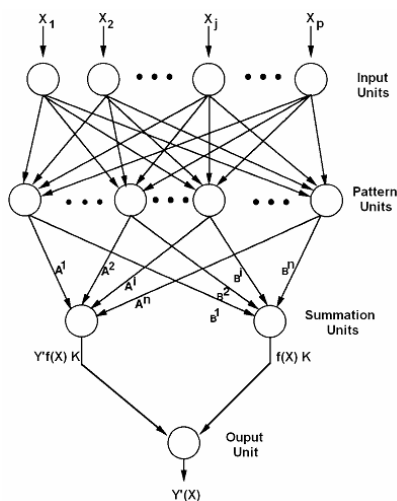


Fig. 1. General Regression Neural Network Diagram

Figure 1 shows the architecture of the implemented GRNN. Input units provide all the scaled measurement independent variables X to all neurons on the second layer,

the pattern units. Each pattern unit can represent one sample exemplar, or, when the number of sample exemplars is large, a cluster center representing a subset of related exemplars in the sample [24]. When a new vector X is fed into the network, it is subtracted from the stored vector in the pattern units. The absolute values of the differences are summed and passed on to an exponential activation function. The pattern units output are fed into the summation units, which perform a dot product between a weight vector and a vector representing the outputs from the pattern units. The summation unit that generates an estimate of $f(X)K$ (where K is a constant determined by the Parzen window used, but that does need computing) sums the outputs of the pattern units weighted by the number of observations represented by each cluster center. The summation unit that estimates $Y'f(X)K$ multiplies each value from a pattern unit by the sum of the samples Y^j associated with cluster center X^i . The output unit merely divides $Y'f(X)K$ by $f(X)K$ to produce the desired estimate of Y [24].

4 Analysis

4.1 Verification of Models

The models presented in Section 3 were applied to the original dataset generated from 2005 to 2008 (the software tool used for training the GRNN was MatLab 6.1 having a spread=10 as best value for fitting) and the MER by project as well as the MMER by model were then calculated. For example, the actual data from a project was developed (design, design review, code, code review, compile and testing phases) in 171 minutes, it contained 95 N&C and 8 reused lines of code, respectively, and it was developed by a person having 14 months of experience in the programming language used. The multiple regression equation generated an effort equal to 155.49 minutes, whereas the GRNN yielded 168.83 minutes. The MER by model is the following:

$$\begin{array}{ll} \text{Statistical regression} & \text{GRNN} \\ MER_i = \frac{|171 - 155.49|}{155.49} = 0.10 & MER_i = \frac{|171 - 168.83|}{168.83} = 0.01 \end{array}$$

The MMER values by model were the following:

- Multiple Linear Regression = 0.27
- General Regression Neural Network = 0.23

The ANOVA for MER for the projects showed that there was a statistically significant difference amongst the accuracy of estimation for the two techniques at the 95.0% confidence level. The following three assumptions of residuals for MER ANOVA were analysed: independent samples, equal standard deviations and normal populations.

4.2 Validation of Models

A new set of 47 programmers developed 156 projects in the year 2009 and the first months of 2010. These experiments had the same characteristics as the projects

presented in Section 2. Once the two models for estimating the development effort were applied to these new data, the MER by project as well as the MMER by model were calculated. The MMER results by model were the following:

- Multiple Linear Regression = 0.24
- General Regression Neural Network = 0.26

Results from an ANOVA for MER showed that there was not a statistically significant difference between the two models at the 95% confidence level. In addition, the assumptions of residuals for MER ANOVA by group were analysed and the equal deviation as well as the normality were met.

5 Conclusions and Future Research

Based on the fact that after product size, people factors have the strongest influence in determining the amount of effort required to develop a software product, the models developed in this research include the developers' programming language experience.

This study compared two models for estimating software development effort of projects developed in a controlled experiment. One of the models developed is one of the most used in the software estimation field: statistical regression, whereas the other model corresponded to a computational intelligence technique: a neural network. These models were generated from a dataset composed of 156 projects developed by 51 persons. These two models were then applied to a new group of 156 projects developed by 47 programmers. All projects were developed in accordance with a similar experiment design within a laboratory learning environment and using a technology specifically designed for that kind of environment: Personal Software Process. In the verification stage, the general regression neural network (GRNN) had a better accuracy than the regression model, whereas in the validation stage, there was not a difference with statistical significance between the two models. These results suggest that a GRNN having as input new and changed code, reused code and programming language experience can be used for estimating the development effort of projects when developed in a laboratory learning environment and based upon a disciplined process as the one suggested by the Personal Software Process.

Based upon complexity increases with size of projects, future research involves the use of a GRNN for estimating the development effort of large scale projects.

Acknowledgment. The authors of this paper would like to thank CUCEA of Guadalajara University, Jalisco, México, Programa de Mejoramiento del Profesorado (PROMEP), as well as Consejo Nacional de Ciencia y Tecnología (Conacyt).

References

1. Aguilar-Ruiz, J.S., Ramos, I., Riquelme, J.C., Toro, M.: An evolutionary approach to estimating software development projects. *Journal of Information and Software Technology* 43, 875–882 (2001)

2. Boehm, B., Abts, C., Chulani, S.: Software development cost estimation approaches: A survey. *Journal of Annals of Software Engineering* 10, 177–205 (2000)
3. Boehm, B., Abts, C., Brown, A.W., Chulani, S., Clarck, B.K., Horowitz, E., Madachy, R., Reifer, D., Steece, B.: *COCOMO II*. Prentice Hall, Englewood Cliffs (2000)
4. Burguess, C.J., Lefley, M.: Can genetic programming improve software effort estimation? A comparative evaluation. *Journal of Information and Software Technology* 43(14), 863–873 (2001)
5. De Barcelos, T.I.F., Simies da Silva, J.D., Sant Anna, N.: An investigation of artificial neural networks based prediction systems in software project management. *Journal of Systems and Software* 81(3), 356–367 (2008)
6. Donald, J.B., Hilburn, T.B., Hislop, G., Lutz, M., McCracken, M., Mengel, S.: *Guidelines for Software Engineering Education*. Carnegie mellon University, CMU/SEI-99-TR-032 (1999)
7. Foss, T., Stensrud, E., Kitchenham, B., Myrtviet, I.: A Simulation Study of the Model Evaluation Criterion MMRE. *IEEE Transactions on Software Engineering* 29(11), 985–995 (2003)
8. Heiat, A.: Comparison of artificial neural network and regression models for estimating software development effort. *Journal of Information and Software Technology* 44(15), 911–922 (2002)
9. Humphrey, W.: *A Discipline for Software Engineering*. Addison-Wesley, Reading (1995)
10. Humphrey, W.: Three Process Perspectives: Organizations, Teams, and People. *Journal of Annals of Software Engineering* 14, 39–72 (2002)
11. Jørgensen, M.: Forecasting of Software Development Work Effort: Evidence on Expert Judgment and Formal Models. *Journal of Forecasting* 23(3), 449–462 (2007)
12. Jørgensen, M.: A Critique of How We Measure and Interpret the Accuracy of Software Development Effort Estimation. In: *1st International Workshop on Software Productivity Analysis and Cost Estimation*, pp. 15–22 (2007)
13. Kalichanin-Balich, I., Lopez-Martín, C.: Applying a Feedforward Neural Network for Predicting Software Development Effort of Short-Scale Projects. In: *International Conference in Software Engineering Research and Applications, SERA*, pp. 269–275 (2010)
14. Lisack, S.K.: The Personal Software Process in the Classroom: Student Reactions (An Experience Report). In: *13th IEEE Conference on Software Engineering Education & Training*, pp. 166–175 (2000)
15. Lopez-Martín, C.: A fuzzy logic model for predicting the development effort of short scale programs based upon two independent variables. *Journal of Applied Soft Computing* 11(1) (2011)
16. López-Martín, C.: Applying a general regression neural network for predicting development effort of short-scale programs. *Journal of Neural Computing and Applications* 20(3), 389–401 (2011)
17. MacDonell, S.G.: Software source code sizing using fuzzy logic modelling. *Journal of Information and Software Technology* 45(7), 389–404 (2003)
18. Paliwal, M., Kumar, U.A.: Neural networks and statistical techniques: A review of applications. *Journal of Expert Systems with Applications* 36, 2–17 (2009)
19. Park, H., Baek, S.: An empirical validation of a neural network model for software effort estimation. *Journal of Expert Systems with Applications* 35, 929–937 (2008)
20. Park, R.E.: *Software Size Measurement: A Framework for Counting Source Statements*. Software Engineering Institute, Carnegie Mellon University, CMU/SEI-92-TR-020 (1992)

21. Pedrycz, W.: Computational Intelligence as an Emerging Paradigm of Software Engineering. In: 14th international conference on Software Engineering and Knowledge Engineering, vol. I, pp. 7–14 (2002)
22. Rombach, D., Münch, J., Ocampo, A., Humphrey, W.S., Burton, D.: Teaching disciplined software development. *Journal Systems and Software* 81(5), 747–763 (2008)
23. Runeson, P.: Experiences from Teaching PSP for Freshmen. 14th IEEE Conference on Software Engineering Education and Training (2001)
24. Specht, D.F.: A General Regression Neural Network. *IEEE Transactions on Neural Networks* 7(3), 568–576 (1991)

An Ensemble of Degraded Neural Networks

Eduardo Vázquez-Santacruz and Debrup Chakraborty

¹ Department of Electrical Engineering and Computer Science, CINVESTAV-IPN,
Unidad Guadalajara, Av. Científica 1145, Colonia El Bajío,
Zapopan, Jalisco 45015, Mexico
evazquez@gdl.cinvestav.mx

² Department of Computer Science, CINVESTAV-IPN, Av. IPN 2508,
Col: San Pedro Zacatenco, Mexico City 07360, Mexico
debrup@cs.cinvestav.mx

Abstract. In this paper we present a new method to create neural network ensembles. In an ensemble method like bagging one needs to train multiple neural networks to create the ensemble. Here we present a scheme to generate different copies of a network from one trained network, and use those copies to create the ensemble. The copies are produced by adding controlled noise to a trained base network. We provide a preliminary theoretical justification for our method and experimentally validate the method on several standard data sets. Our method can improve the accuracy of a base network and give rise to considerable savings in training time compared to bagging.

1 Introduction

Let $\mathcal{L} = \{(\mathbf{x}_i, y_i) : i = 1, \dots, n\}$ be a training set where $\mathbf{x}$ is a feature vector and y is its corresponding numerical response or a class label. There are plenty of procedures available in literature which uses this training set $\mathcal{L}$ to form a predictor function ϕ , which on input $\mathbf{x}$, give y as the output, i.e., the function ϕ approximates the input-output relationship between $\mathbf{x}$ and y . The function generally is of the form $\phi(\mathbf{x}, \mathbf{W})$, where $\mathbf{W}$ is a parameter vector which is decided upon using $\mathcal{L}$. A very popular procedure for obtaining the predictor ϕ is by training a neural network with $\mathcal{L}$. In this case we will call the predictor function $N(\mathbf{x}, \mathbf{W})$, where the parameter vector $\mathbf{W}$ contains the parameters (weights and biases) of the neural network, which are learned with the aid of the training set. The training algorithm finds that $\mathbf{W}$ which minimizes the error committed by the predictor on the training set (the training error). The operational performance measure of a predictor function is the error committed on future data points which are not present in the training set. The error on such points which are not in the training set is known as the generalization error (or test error) of the predictor. Practice has shown that a direct minimization of the training error does not always guarantee a small generalization error. There are plenty of methods available in the literature

which improves the generalization ability of predictor functions learned from data. There are also particular methods in the context of neural networks, that broadly fall into the following categories: (a) early stopping [1] (b) Complexity control of the network (weight pruning strategies, etc.) [16] (c) training with noise [10] (d) ensemble methods [9, 17]. In this paper we are interested with the last paradigm.

It have been noticed that an ensemble of predictors have better generalization abilities than a single predictor function [3, 15]. In the past few years there have been numerous proposals for creating ensemble of predictors. Two of the well known proposals in this regard are Bagging [3] and Boosting [14]. Ample theoretical studies of Bagging and Boosting have been reported in the literature, and these studies clearly point out why and under which scenarios ensembles created by these methods can give better predictions [3, 15]. Also, in the last few years numerous variants of bagging and boosting have been proposed [11, 12, 7].

Right from the early nineties neural network ensembles has also been widely studied [9, 17]. The studies regarding neural network ensembles are mainly for adapting suitably the general ensemble techniques in case of neural networks [5]. The other studies have been focussed on developing heuristics to choose better candidates for an ensemble such that each candidate has good prediction power along with that the selected candidates have better diversity [4, 8], which is known to effect the performance of an ensemble [11, 12, 13].

Bagging involves creating multiple bootstrap samples [6] from $\mathcal{L}$ and training predictors from each of the bootstrap samples. The final output is obtained by a suitable aggregation of the output of each predictor. The type of aggregation depends on the type of the output, i.e., whether it is a numerical response or a class label. Leo Brieman in [3] noted that neural network predictors are unstable, i.e., it is not necessary that for a trained neural network, small changes in the input will produce small changes in the output. In [3] it was also noted that along with neural networks other very popular methods like classification and regression trees, subset selection in linear regression are also unstable. In [3] it has been shown that for an unstable classifier bagging can improve the prediction both in terms of stability and accuracy.

There are theoretical guarantees about good prediction accuracy in case bagging is applied to neural networks, but using bagging to learn multiple neural predictors seems to be suboptimal, as neural network training is computationally expensive. Creating multiple neural networks from the bootstrap samples is costly. In this paper we propose a new method to create neural network ensembles. The method involves adding controlled noise to a base network and thus create numerous clones of the network and use an ensemble of the degraded networks for prediction. Our experiments show that such an ensemble can improve the performance of a base network, but the performance of such ensembles on the average is poorer than conventional bagging. What we gain by our method is a drastic reduction of training time, as in the average training using our method requires almost half training time compared to bagging. Under some assumptions we also provide a theoretical justification of why such an ensemble works.

2 The Strategy

In the discussion that follows by a neural network we shall mean a multilayered perceptron (MLP). As training multiple neural networks from bootstrap samples of the training data is time consuming, here we propose a method for generating multiple copies of neural networks from a single trained network. Let $N(\mathbf{W})$ be a neural network trained using the training set $\mathcal{L}$. We call this network as the *base network*. Here $\mathbf{W}$ is a vector containing all the learnable parameters of the network, thus, if the network contains s weights and r biases, then $\mathbf{W}$ will have $p = s + r$ components. Let $\mathbf{W} = (w_1, w_2, \dots, w_p)$. Now a little perturbation of the weight vector will generate a different network, whose performance would be comparable with the base network. We create an ensemble of these degraded network, which acts as the final predictor network. In the following paragraphs we discuss the steps of our method in details.

Let $N(\mathbf{W})$ be a base network. By a base network we mean an MLP trained with the given training set $\mathcal{L}$. Any standard technique like error back-propagation or some of its variants can be used to train the base network $N(\mathbf{W})$. The size of the parameter vector $\mathbf{W}$ would depend on the architecture of the base network. We assume that the architecture of $N(\mathbf{W})$ is adequate to learn the problem represented by the training set $\mathcal{L}$. Once we have the base network, then we create a degraded version of $N(\mathbf{W})$ by adding a zero mean Gaussian noise to each of its components (weights and bias). Thus if $\mathbf{W} = (w_1, w_2, \dots, w_p)$ be the parameter vector of the base network $N(\mathbf{W})$ and $\mathbf{W}^d = (w_1^d, w_2^d, \dots, w_p^d)$ be the parameter vector of a degraded version of $N(\mathbf{W})$, then

$$w_i^d = w_i + e_i, \forall i = 1, 2, \dots, p \quad (1)$$

where $e_i \sim \mathcal{N}(0, \sigma_i)$, i.e. e_i is a random number drawn from a normal distribution of zero mean and variance σ , and to generate each component of the parameter of the degraded version, e_i is drawn independently of its previous values. Thus we see that the amount of degradation that a component receives is dependent on the value of σ . Further we shall call this σ as the *degradation parameter*. A large σ means a more degraded network, in average. Thus by controlling the parameter σ one can control the degree of degradation of a network. Let ϵ be the training error of the base network on the training set $\mathcal{L}$ and let ϵ_d be the error committed by the degraded network $N(\mathbf{W}^d)$ on $\mathcal{L}$. We call $N(\mathbf{W}^d)$ as a *valid candidate* if $\epsilon_d \leq t\epsilon$. Where t is a user defined threshold, which we call as the *selection threshold*. For our simulations we assume $t = 1.05$, i.e., we accept a degraded copy to be a valid candidate if the error committed by it on the training set is within 5% of the error that the base network commits on the training set [\[1\]](#). Thus by repeated degradation we obtain the desired number of valid candidates and these are used to form the ensemble, with a suitable aggregation function. The overall strategy is summarized in the algorithm shown in Table [\[1\]](#).

¹ Note that, if the error committed by the base network is zero, then this multiplicative threshold does not work, in fact then there is no point in creating an ensemble.

Table 1. The overall strategy to create ensembles from degraded networks

```

Algorithm Make_Ensemble( $N(\mathbf{W}), \sigma, m, t, \epsilon, \mathcal{L}$ )
1.   Let  $\mathbf{W} = (w_1, w_2, \dots, w_p)$ ;
2.    $V \leftarrow \emptyset$ ;  $d \leftarrow 1$ ;
3.   while  $|V| < m$  ,
4.     for  $j = 1$  to  $p$ ,
5.        $e \sim \mathcal{N}(0, \sigma)$ 
6.        $w_j^d \leftarrow w_j + e$ ;
7.     end for
8.      $\mathbf{W}^d \leftarrow (w_1^d, w_2^d, \dots, w_p^d)$ ;
9.     Let  $\epsilon_d$  be the error committed by  $N(\mathbf{W}^d)$  on  $\mathcal{L}$ ;
10.    if  $\epsilon_d \leq t\epsilon$ , then
11.       $V \leftarrow V \cup \{N(\mathbf{W}^d)\}$ ;  $d \leftarrow d + 1$ ;
12.    end if
13.  end while
14.  Create ensemble of the networks in  $\{N(\mathbf{W})\} \cup V$ ;

```

In the algorithm described in Table 1 the inputs are a trained base network $N(\mathbf{W})$, the degradation parameter σ , a positive integer m where $m + 1$ is the number of candidates to be present in the ensemble, the selection threshold t , and ϵ , the error committed by $N(\mathbf{W})$ on $\mathcal{L}$ and the training set $\mathcal{L}$. The algorithm collects the valid candidates in the set V . It creates degraded copies by drawing a random number e from a normal distribution with zero mean and variance σ and adding this noise to the parameters of the base network. Then, it checks whether the degraded copy is a valid candidate and continues creating degraded copies until it gets m copies. At the end an ensemble of $m + 1$ candidate networks is created using the base network and the m valid candidates generated from the base network. By creation of an ensemble we mean using all networks $N(\mathbf{W}), N(\mathbf{W}^1), \dots, N(\mathbf{W}^m)$ together for prediction. For a test point $\mathbf{x}$ we present the point to all networks and obtain $\xi = N(\mathbf{x}, \mathbf{W}), \xi^1 = N(\mathbf{x}, \mathbf{W}^1), \dots, \xi^m = N(\mathbf{x}, \mathbf{W}^m)$. These outputs are aggregated together to get the final output. The most preferred technique of aggregation is to use the simple average or a majority vote.

Note that, m , t and σ are user defined parameters. The choice of m and t is not crucial. The number of candidates in an ensemble can be suitably selected and a guideline for selection of t has already been given. A proper choice of σ is most crucial for the proper functioning of the algorithm. It is possible that the “optimum” value of σ is data dependent. We suggest to use a “small” value for sigma, the following example will illustrate some effects on the choice of σ also will serve as a motivation that our method works.

3 An Example

Here we provide a convincing example showing that the gross methodology works well. We consider the problem of learning a noisy sine curve as shown in the equation below.

$$f(x_i) = 0.4 \sin x_i + 0.5, x \in [-\pi, \pi]. \quad (2)$$

We generate 150 input-output pairs (x_i, y_i) , $i = 1, 2, \dots, 150$, with x_i s generated uniformly on $[-\pi, \pi]$ and $y_i = f(x_i) + r_i$, where $r_i \sim \mathcal{N}(0, 0.001)$ accounts for a zero mean Gaussian noise with a small variance. We use these 150 input-output pairs as a training set. Additionally, we generate 50 more pairs using eq. 2 which we use as the test set. Now we train a base network using the training data and create degraded copies of the base network using different values of σ . Figure 1 shows the variation of the sum of square errors measured on the test data for different values of σ . In this example the base network had 10 hidden nodes in a single hidden layer, and was trained using the conventional back-propagation algorithm. For each run we generated 14 valid candidates from the network and thus created an ensemble of 15 networks according to the algorithm `Make_Ensemble`. In Fig. 1 we show two representative scenarios of variation of the sum of square error (SSE) on the test points of the ensemble against the choice of various values of σ .

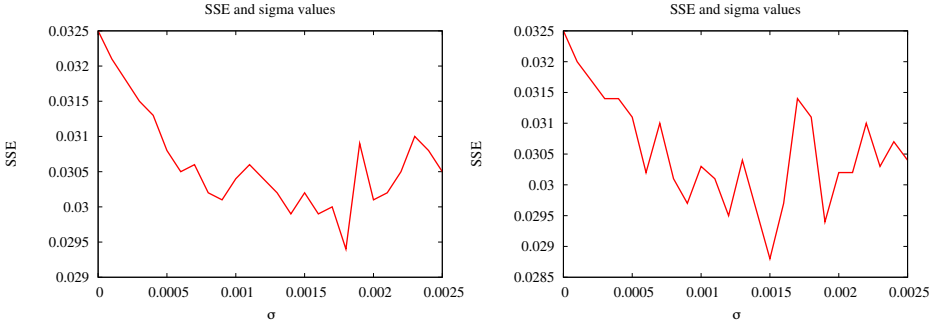


Fig. 1. Two representative runs showing the error of ensemble with variation of the degradation parameter (σ). In the figures the σ is plotted in the x axis and the SSE in the y axis.

Figure 1 clearly shows that as the value of the degradation parameter increases the SSE of the ensemble decreases, and after a certain value of the degradation parameter the SSE starts to increase. In the two scenarios shown in Fig. 1 the optimum value of σ lies in the interval $[0.001, 0.002]$. In all the runs that we made with this data this was true. The explanation of the variation of the SSE with the degradation parameter is probably that for very small values of σ , the valid candidates generated are too similar to the base network, thus there is too little variability among the candidates of the ensemble, thus an improvement over the base network is not possible. For high values of the degradation parameter,

the structure of the base network gets altered and thus the degraded candidates created cannot really sustain the learning capabilities of the base network. Thus, it seems that there would be an optimal value of the degradation parameter which will give rise to a good ensemble. But, in this study we could not give a method to select the parameter σ , but our experience show that a small value of σ can give rise to good ensembles. Based on our experiments we suggest to select σ in the range of $[0.001, 0.002]$, and this seems to give acceptable results across data sets.

4 A Theoretical Justification

In this section we give a theoretical justification of why our method works. We first consider an ideal scenario, where we assume an ensemble of neural networks whose parameters are sampled from a specific distribution. But this specific distribution will always be unknown and sampling from that would not be possible. Later we argue why our scenario closely resembles the ideal scenario.

When a neural network architecture and the internal activation functions are fixed and also the parameters of a learning algorithm (like the learning rate in back-propagation) are fixed then the learning algorithm becomes a deterministic algorithm. A learning algorithm then can be viewed as a function which takes as input a training set and outputs the weights and biases of the network which can be viewed as a parameter vector $\mathbf{W} \in \mathbb{R}^p$. In other words, the learning algorithm on a fixed architecture A can be characterized by the function $\Lambda_A : \mathbb{R}^q \times \mathcal{C} \rightarrow \mathbb{R}^p$. Where the training set $\mathcal{L} \subset \mathbb{R}^q \times \mathcal{C}$, we implicitly assume that the input feature vector is a q dimensional real vector and $\mathcal{C}$ is the set of possible class labels or the numerical responses. Also we assume that the specific architecture A has p learnable parameters. In all machine learning tasks it is assumed that the training data (also the test data) are generated from a fixed (but unknown) time invariant probability distribution. Let the the unknown distribution from which the training data $\mathcal{L}$ has been generated be $\mathcal{P}$. Let $\mathcal{L}_1, \mathcal{L}_2, \dots, \mathcal{L}_r$ be r training sets generated independently from the distribution $\mathcal{P}$. Then for a fixed architecture A , the learning algorithm Λ_A will produce different parameter vectors $\mathbf{W}_1, \mathbf{W}_2, \dots, \mathbf{W}_r$ corresponding to the r different training sets. The distribution $\mathcal{P}$ on the training data will induce a distribution on the parameter vectors $\mathbf{W}_i$ s, let $\mathcal{P}_W$ denote this distribution. Now, if this distribution $\mathcal{P}_W$ is known then we can sample parameter vectors using this distribution, which can be treated as parameters of a neural network trained by the learning algorithm Λ_A using the data generated following the distribution $\mathcal{P}$. We assume that we construct an ensemble of neural networks with parameters drawn from the distribution $\mathcal{P}_W$. The ensemble of such neural networks be denoted by N_E . Let $\mathcal{W}$ be a random variable following the distribution $\mathcal{P}_W$, and $\mathcal{E}$ denote the expectation operator, then we have

$$N_E(\mathbf{x}) = \mathcal{E}_{\mathcal{W}}[N(\mathbf{x}, \mathcal{W})].$$

Assuming $\mathbf{X}$ and Y to be random variables having a joint distribution $\mathcal{P}$ the average prediction error for a single network would thus be

$$e = \mathcal{E}_{\mathcal{W}}[\mathcal{E}_{\mathbf{X},Y}[\{Y - N(\mathbf{X}, \mathcal{W})\}^2]].$$

The error in the ensemble N_E would be

$$e_E = \mathcal{E}_{\mathbf{X},Y}[\{Y - N_E(\mathbf{X})\}^2]$$

Now we have

$$\begin{aligned} e &= \mathcal{E}_{\mathcal{W}}[\mathcal{E}_{\mathbf{X},Y}[\{Y - N(\mathbf{X}, \mathcal{W})\}^2]] \\ &= \mathcal{E}_{\mathcal{W}}[\mathcal{E}_{\mathbf{X},Y}[\{Y^2 - 2YN(\mathbf{X}, \mathcal{W}) + N^2(\mathbf{X}, \mathcal{W})\}]] \\ &= \mathcal{E}_{\mathbf{X},Y}[Y^2] - 2\mathcal{E}_{\mathbf{X},Y}[YN_E(\mathbf{X})] + \mathcal{E}_{\mathbf{X},Y}[\mathcal{E}_{\mathcal{W}}[N^2(\mathbf{X}, \mathcal{W})]] \end{aligned} \quad (3)$$

As for any random variable Z , we have $(\mathcal{E}[Z])^2 \leq \mathcal{E}[Z^2]$, hence

$$\mathcal{E}_{\mathcal{W}}[N^2(\mathbf{X}, \mathcal{W})] \geq [\mathcal{E}_{\mathcal{W}}[N(\mathbf{X}, \mathcal{W})]]^2 = [N_E(\mathbf{X})]^2. \quad (4)$$

Hence using eqs. (3) and (4), we have

$$e \geq \mathcal{E}_{\mathbf{X},Y}[(Y - N_E(\mathbf{X}))^2] = e_E. \quad (5)$$

Thus we see that the error committed by the ensemble is less than that of an individual predictor, if we create an ensemble of neural networks whose parameters have been sampled from $\mathcal{P}_W$. The above explanation that we gave is due to function approximation type problems, and it cannot directly applicable for general classification problems where the outputs are discrete and do not generally bear a metric relationship between them. But, while an multilayered perceptron is trained for a classification task, the class labels are suitably coded as binary vectors and the network learns a function approximation task associated with binary outputs. Due to the choice of the activation functions (typically a sigmoid function is used in case of MLPs), the outputs of the MLP are not binary, they are real numbers in the interval $[0, 1]$. And, those real output vectors are suitably interpreted to get the final solution. So the above analysis is valid for an MLP trained as a classifier also, but may not be valid for a general classifier.

The above analysis represents an idealistic scenario, where we assume that the ensemble is created using parameter vectors $\mathbf{W}_i$ s which follow a certain distribution $\mathcal{P}_W$. $\mathcal{P}_W$ is the distribution induced by the distribution of the input output data through the learning algorithm. Needless to say that the distribution $\mathcal{P}_W$ is unknown, in-fact a knowledge of the input-output data distribution $\mathcal{P}$ does not guarantee a closed form formula for the distribution $\mathcal{P}_W$ as they are related in a highly non-linear manner through the learning algorithm. And, to us $\mathcal{P}$ is also unknown. The best algorithm which would be faithful to the above analysis would be a technique to sample parameter vectors from the distribution $\mathcal{P}_W$, and the first step towards it would be an estimate for the distribution $\mathcal{P}_W$. We claim that our algorithm samples parameter vectors from $\mathcal{P}_W$ under certain assumptions.

To see this, let us observe our algorithm a bit closely. Our algorithm starts with a base network $N(\mathbf{W}^b)$. The parameters of the base network (i.e. $\mathbf{W}^b$) gives us one sample from $\mathcal{P}_W$. If we assume that $\mathcal{P}_W$ follows a multidimensional normal distribution centered around $\mathbf{W}^b$, then our process of degradation do generate parameter vectors following the distribution $\mathcal{P}_W$, with a small variance σ . This naive assumption is unlikely to capture the whole distribution $\mathcal{P}_W$ but probably restricts us to a small area in the whole distribution. But as the results in the following section suggest, this naive approximation also can give us encouraging results.

5 Experimental Results

We report performance of our method on six classification data sets from [2]. For training the neural networks we use the *trainidx* algorithm as implemented in the Neural-network toolbox of MATLAB. All the networks we use in the experiments have a single hidden layer with 10 nodes. As usual, this decision was rather ad-hoc and a change in the number of hidden nodes would not change the conclusions of our experiments. For each run we use the degradation parameter $\sigma = 0.0015$.

Table 2. Performance Comparison

Data set	Base Network Performance (in %)	Degraded Ensemble Performance (in %)	Conventional Bagging Performance (in %)
Iris	91.26 $\pm$ 6.11	91.33 $\pm$ 6.81	96.09 $\pm$ 5.66
Glass	67.87 $\pm$ 3.61	71.11 $\pm$ 9.76	72.96 $\pm$ 8.05
Waveform-40	60.19 $\pm$ 5.99	71.02 $\pm$ 3.62	85.41 $\pm$ 0.94
Waveform-21	62.75 $\pm$ 5.95	72.64 $\pm$ 3.80	84.11 $\pm$ 1.89
Pima-Diabetes	66.35 $\pm$ 2.10	68.41 $\pm$ 1.97	75.11 $\pm$ 4.06
Wine	83.90 $\pm$ 7.99	85.54 $\pm$ 7.63	97.18 $\pm$ 1.89

Table 3. Training Times

Data set	Degraded Ensemble (time in secs)	Conventional Bagging (time in secs)
Iris	18.92 $\pm$ 4.37	134.01 $\pm$ 2.32
Glass	45.41 $\pm$ 3.25	66.11 $\pm$ 1.32
Waveform-40	367.12 $\pm$ 19.61	4077.50 $\pm$ 166.80
Waveform-21	294.79 $\pm$ 18.60	3669.75 $\pm$ 142.14
Pima-Diabetes	41.30 $\pm$ 10.34	797.81 $\pm$ 8.12
Wine	29.99 $\pm$ 8.26	85.54 $\pm$ 7.63

In Table 2, we show the comparative performance of our network. All the results reported are on 10 fold cross validation repeated 10 times. The second column of Table 2 gives us the average performance of a single network. For

each of the trained base networks 10 degraded copies were created and then aggregated with the principle of majority voting. The third column of the table shows the average performance of the ensemble of the degraded networks. The last column give result of conventional bagging, where also 10 candidate networks trained from bootstrap samples of the training data were trained and aggregated using the principle of majority voting.

Table 2 clearly shows that our method of creating ensembles can significantly enhance the performance of the base network. The entries in bold in Table 2 suggests that the improvement in the degraded ensemble was statistically significant². But in all cases the results obtained by our method are poorer than that obtained by conventional bagging. But, what we gain in comparison to bagging is the training time. As discussed earlier, conventional bagging in neural networks amounts to training multiple networks, but in our method the candidates of the ensemble are created by perturbing the parameters of the base network. This gives rise to a huge savings of time compared to bagging. The training times for our method and bagging are depicted in Table 3. Tables 2 and 3 clearly indicates that our method can improve the performance of a single network to a large extent in very less time.

6 Discussions and Conclusion

The results in section 5 show that the method of creating ensembles from degraded networks is able to improve over the base network. The accuracy of these kind of ensembles are not better than bagging, the reason behind this is probably the lack of diversity among the candidate classifiers. The training time is also significantly low. An important feature of our methodology is that the ensemble can be created without access to the training data. It may be possible that a user has a trained network to perform a specific task but does not have access to the training data. In such a scenario, improving the accuracy through other available ensemble methods is not possible, as in all of the reported methods access to the training data is necessary to create ensembles, but our method does not require access to the training data. Additional clones can be generated from a trained network to create ensembles of the clones. This feature may find application in certain scenarios.

Some future work of immediate interest are as follows:

1. The most crucial part of the proposed algorithm is the selection of the degradation parameter σ . Unfortunately we were unable to provide a procedure to obtain an optimal value for σ in this work. But, our experience shows that small values of σ do work well. We are investigating ways to find out an optimal value of σ for a given data. We believe that this will have an immediate impact on the performance of our algorithm.
2. We noted down in Section 4 that there is a theoretical guarantee that an ensemble of networks created from parameters sampled from the distribution $\mathcal{P}_W$ would give rise to less prediction errors. With certain assumptions

² These results are based on a studentized t-test with 95% confidence.

we viewed our scheme for degradation as sampling vectors from the distribution $\mathcal{P}_W$. A possible technique to overcome some assumptions may be to start with multiple base networks and thus try to estimate the distribution $\mathcal{P}_W$ by a better technique (say a kernel density estimate). This would have implication on the training time, but may give rise to better accuracy.

References

1. Amari, S., Murata, N., Muller, K.-R., Finke, M., Yang, H.H.: Asymtotic statistical theory of overtraining and cross-validation. *IEEE Transactions on Neural Networks* 8(5), 985–996 (1997)
2. Asuncion, A., Newman, D.J.: UCI machine learning repository (2007)
3. Breiman, L.: Bagging predictors. *Machine Learning* 24(2), 123–140 (1996)
4. Chen, R., Yu, J.: An improved bagging neural network ensemble algorithm and its application. In: Third International Conference on Natural Computation, vol. 5, pp. 730–734 (2007)
5. Drucker, H., Schapire, R.E., Simard, P.: Improving performance in neural networks using a boosting algorithm. In: Hanson, S.J., Cowan, J.D., Lee Giles, C. (eds.) *NIPS*, pp. 42–49. Morgan Kaufmann, San Francisco (1992)
6. Efron, B., Tibshirani, R.: *An Introduction to the Bootstrap*. CRC Press, Boca Raton (1993)
7. Gao, H., Huang, D., Liu, W., Yang, Y.: Double rule learning in boosting. *International Journal of Innovative Computing, Information and Control* 4(6), 1411–1420 (2008)
8. Georgiou, V.L., Alevizos, P.D., Vrahatis, M.N.: Novel approaches to probabilistic neural networks through bagging and evolutionary estimating of prior probabilities. *Neural Processing Letters* 27(2), 153–162 (2008)
9. Hansen, L.K., Salamon, P.: Neural network ensembles. *IEEE Trans. Pattern Anal. Mach. Intell.* 12(10), 993–1001 (1990)
10. Holmstrom, L., Koistinen, P.: Using additive noise in backpropagation training. *IEEE Transactions on Neural Networks* 3, 24–38 (1992)
11. Kuncheva, L.I.: Diversity in multiple classifier systems. *Information Fusion* 6(1), 3–4 (2005)
12. Kuncheva, L.I., Rodríguez, J.J.: Classifier ensembles with a random linear oracle. *IEEE Trans. Knowl. Data Eng.* 19(4), 500–508 (2007)
13. Kuncheva, L.I., Whitaker, C.J.: Measures of diversity in classifier ensembles and their relationship with the ensemble accuracy. *Machine Learning* 51(2), 181–207 (2003)
14. Schapire, R.E.: A brief introduction to boosting. In: Dean, T. (ed.) *IJCAI*, pp. 1401–1406. Morgan Kaufmann, San Francisco (1999)
15. Schapire, R.E.: Theoretical views of boosting. In: Fischer, P., Simon, H.U. (eds.) *EuroCOLT 1999. LNCS (LNAI)*, vol. 1572, pp. 1–10. Springer, Heidelberg (1999)
16. Setiono, R.: A penalty function approach for pruning feed forward neural networks. *Neural Computation* 9, 185–204 (1997)
17. Zhou, Z.-H., Wu, J., Tang, W.: Ensembling neural networks: Many could be better than all. *Artificial Intelligence* 137(1-2), 239–263 (2002)

Genetic Fuzzy Relational Neural Network for Infant Cry Classification

Alejandro Rosales-Pérez, Carlos A. Reyes-García, and Pilar Gómez-Gil

National Institute of Astrophysics, Optics and Electronics (INAOE)
Computer Science Department
Luis E. Erro No.1 Tonantzintla,
Puebla, México
`{arosales,kargaxxi,pgomez}@ccc.inaoep.mx`

Abstract. In this paper we describe a genetic fuzzy relational neural network (FRNN) designed for classification tasks. The genetic part of the proposed system determines the best configuration for the fuzzy relational neural network. Besides optimizing the parameters for the FRNN, the fuzzy membership functions are adjusted to fit the problem. The system is tested in several infant cry database reaching results up to 97.55%. The design and implementation process as well as some experiments along with their results are shown.

Keywords: Fuzzy relational neural network, genetic algorithm, infant cry classification.

1 Introduction

In this work we present a genetic fuzzy relational neural network. The relational neural network was originally proposed by Pedrycz [6] and it was extended by Reyes, who designed a Fuzzy Relational Neuronal Network (FRNN) for speech recognition [7]. Unlike most of the fuzzy neural network models that only use the elements of fuzzy sets theory for the learning process or in their structure, the FRNN uses fuzzy sets for both input/output data and the structure and functioning of the classifier itself.

The FRNN has been used for pattern recognition. In [8,9,10] the FRNN was implemented for infant cry classification. Also, in [2] a proposal is presented to optimize the parameters for the FRNN using a genetic algorithm. As a complement, in this work we propose to use a genetic algorithm for optimizing the parameters for the FRNN, as well as the parameters of the fuzzy membership functions. The proposed system has been tested on several infant cry databases to classify asphyxia, deaf, normal, hungry and pain cries. All experiments are binary classification. Although the problem of infant cry classification is indeed a multiclass problem, and our team has treated it in that way in several previous works([8,9,10]), but for the present case we present a binary classification because our purpose is to compare our results with a particular similar work

wich precissely had that binary approach. Results are compared with the work of Barajas and Reyes [2], that used the same databases.

The proposed system can be used for automatic infant cry classification. This classifier may be a powerful tool for health professionals and parents during the detection of pathologies in infants.

The rest of the paper is organized as follows. In Section 2 we present the Fuzzy Relational Neural Network Model. In Section 3 we present the designed genetic algorithm used to optimize the parameters of the membership functions as well as the FRNN. Then, in Section 4 we describe the experiment and the results. Finally, in Section 5 the conclusions and future work are presented.

2 Fuzzy Relational Neural Network

The fuzzy relational neural network consists of two layers, the input layer and the output layer. The input layer is formed by a set of $N \times n$ neurons, with each of them corresponding to one of the N linguistic properties assigned to each of the n input features. In the output layer there are l neurons, where each node corresponds to one of the l classes. There is a link from every node in the input layer to every node in the output layer. All the connections, instead of regular weights, are described by fuzzy relations $\mathbf{R} : \mathbf{X} \times \mathbf{Y} \rightarrow [0, 1]$ between the input and output nodes.

The Fuzzy Relational Neural Network (FRNN) operation is divided in two main stages; the first one is for learning and the second one is for processing [7]. Fig. 1 shows the general architecture and stages of this FRNN. Next each phase as well as their modules are described.

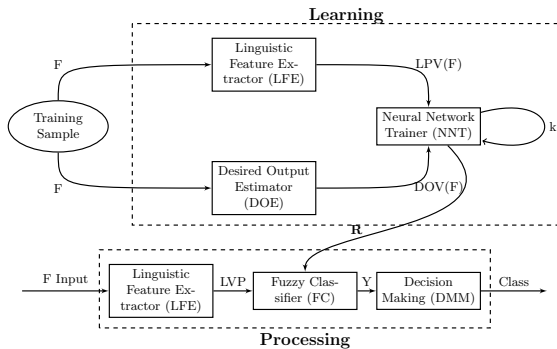


Fig. 1. General architecture and stages of a FRNN for the Automatic Infant Cry Recognition

2.1 Learning Phase

The learning section is divided in three modules: the Linguistic Feature Extractor (LFE), the Desired Output Estimator (DOE) and the Neural Network trainer (NNT).

The first module, LFE, takes the training samples and each input feature (F) is transformed in membership values to each of the linguistic properties assigned. Thus a vector containing n features can be transformed in a $3n$ -dimensional (for describing low, medium, high), $5n$ -dimensional (very low, low, medium, high, very high), or $7n$ -dimensional (very low, low, more or less low, medium, more or less high, high, very high) vectors. The new resulting vector is called Linguistic Properties Vector (LPV).

To calculate the membership values we use four different membership functions: Gaussian, Trapezoidal, Triangular and Bell.

The FRNN is a supervised learning system; in that sense the second module, the DOE, is in charge to compute the membership values of each sample to each class of the problem, which is latter used to calculate the error of the network after each learning iterations. To obtain the desired membership values it is necessary to calculate the weighted distance of the training pattern F_j to the k^{th} class in an l -class problem domain as in equation:

$$z_{ik} = \sqrt{\sum_{j=1}^n \left[\frac{F_{ij} - \mu_{kj}}{\sigma_{kj}} \right]^2} \quad | \quad \text{for } k = 1 \dots l \quad (1)$$

where F_{ij} is the j^{th} feature of the i^{th} pattern vector, μ_{kj} and σ_{kj} denotes, respectively, the mean and the standard deviation of the j^{th} feature for the k^{th} class. The membership value of the i^{th} pattern to class k is defined as follows:

$$\mu_l(\mathbf{F}_i) = \frac{1}{1 + \left(\frac{z_{ik}}{fd} \right)^{fe}} \quad (2)$$

where fe is the exponential fuzzy generator, and fd is the denominational fuzzy generator controlling the amount of fuzziness in this class-membership set. In this case, the higher the distance of the pattern from a class, the lower its membership to that class. Since the training data have fuzzy class boundaries, a pattern point may belong to one or more classes in the input feature space.

Finally, the third module, NNT, takes both the LPV and DOV vectors as the basis for training the network. The LPV is clamped to the input layer and the DOV is clamped to the output layer during training. The outputs of the network are computed to obtain the error at the output layer. The error is represented by the distance between the actual output and the target output. To minimize this error is the objective of the training process. During each learning step, once the error has been computed, the trainer adjusts the relationship values or weights of the corresponding connections, either until a minimum error is obtained or a given number of iterations is completed.

The output of the NNT is a relational matrix¹ containing the knowledge needed to further map the unknown input vectors to their corresponding class during the classification process. The relational neural network, the learning process in a FRNN and parameters updating are explained in detail in [7].

¹ Matrix that contains the fuzzy relations corresponding to each feature.

2.2 Processing Phase

Once the learning phase is completed, the information collected is used to classify unknown patterns; the processing phase is in charge of that task. The modules that form the processing phase are the Linguistic Feature Extractor (LFE), the Fuzzy Classifier (FC), and the Decision Making Module (DMM) (see Fig. 1).

The LFE is similar to the one described in the learning phase. The only difference is that, in the classification phase, the LFE does not calculate new parameters to be applied by the membership function. The LFE in this phase takes the last values of the membership functions, corresponding to each feature in the sample patterns, collected by the LFE in the learning phase, and uses them to calculate the linguistic properties vector (LPV) corresponding to the testing patterns.

The second module, the FC, performs the classification. The fuzzy classification is done by using fuzzy relational products [1]; in our work it can be done in five different ways: by mean of the max-min composition, square, subtriangle and supertriangle relational products, as well as max-geometrical mean. The relational products used are defined, respectively, in equations 3-7.

$$\mathbf{Y}(y_j) = \max_j (\max (\min (\mathbf{X}(x_{ij}), \mathbf{R}(x_{ij}, y_j))), b) \quad (3)$$

$$\mathbf{Y}(y_j) = \max_j (\min (\mathbf{X}(x_{ij}) \leftrightarrow \mathbf{R}(x_{ij}, y_j)), b) \quad (4)$$

$$\mathbf{Y}(y_j) = \max_j (\min (\mathbf{X}(x_{ij}) \rightarrow \mathbf{R}(x_{ij}, y_j)), b) \quad (5)$$

$$\mathbf{Y}(y_j) = \max_j (\min (\mathbf{X}(x_{ij}) \leftarrow \mathbf{R}(x_{ij}, y_j)), b) \quad (6)$$

$$\mathbf{Y}(y_j) = \max_j \left(\max (\mathbf{X}(x_{ij}), \mathbf{R}(x_{ij}, y_j))^{1/2}, b \right) \quad (7)$$

where x_{ij} is the j^{th} component of the i^{th} input pattern, $\mathbf{R}(x_{ij}, y_j)$ is the $(i, j)^{th}$ entry in the relational matrix $\mathbf{R}$ and b is a threshold.

The FC module uses the LPV and the relational matrix –obtained during the learning phase– for classifying new patterns to their corresponding class. The last module, DMM, takes the highest membership value from the class vector and the input sample is assigned to that class.

3 Genetic Algorithms for Optimizing Parameters

Genetic algorithms are part of the fastly emerging evolutionary computation area. Evolutionary computation is inspired in the evolutionary theory and it tries to solve problems using computational models of evolutionary processes, such as selection, reproduction, mutation, survival of the fittest, etc. Generally, a genetic algorithm has five basic components: an encoding that represents the potential solutions for the problem in the form of chromosomes or individuals, a form to create potential initial solutions, a fitness function to measure how close a chromosome is to the desired solution, operators for selection and operators for reproduction [4]. Next we explain how each of these components was designed in this system, in order to optimize the fuzzy membership functions and the parameter of the FRNN.

3.1 Representation of the Chromosome

In this work we use a binary representation of the chromosome because, it fits well to the needs of our problem and because according to Holland [5], this representation allows to have more schemes than decimal representation.

The problem is to optimize the parameter for the FRNN. These parameters are: number of linguistic properties (NPL), type of membership function (TMF), number of training epochs (NTE), learning rate (LR), initial weights for the relational matrix (WRM), the initial threshold (IT), the type relational product to be applied for getting the output classification (TRP), as well as the parameters for the selected membership functions, composing altogether a large search space. The chromosome has $18 + 16 * numLP * n$ bits, where $numLP$ represents the number of linguistic properties and n the number of features in the data set. In that sense, the chromosome size is defined at runtime.

The coding scheme adopted is described in detail next. The type of membership function is represented by two bits: a value of $[0, 0]$ means a triangular function; $[0, 1]$ represents a trapezoidal function, a value of $[1, 0]$ means a gaussian function and $[1, 1]$ represents a general bell function. The number of linguistic properties is represented by two bits: $[0, 0]$ represents three linguistic properties, $[0, 1]$ five linguistic properties and $[1, 0]$ seven linguistic properties. The number of training epochs, learning rate, initial threshold, initial relational weights and the type of output are represented by three bits and their codification is shown in Table 1. The different values for each of these parameters were experimentally established.

Table 1. Bits codification for the genetic algorithm

Parameter	Binary codification							
	[0, 0, 0]	[0, 0, 1]	[0, 1, 0]	[0, 1, 1]	[1, 0, 0]	[1, 0, 1]	[1, 1, 0]	[1, 1, 1]
Epochs train	2	5	10	15	20	25	30	50
Learning rate	0.10	0.15	0.20	0.25	0.30	0.33	0.35	0.40
Bias	0.10	0.15	0.20	0.25	0.30	0.35	0.40	0.50
Weight	0.10	0.20	0.30	0.40	0.50	0.60	0.70	0.80
Output	max-min	square	sub-triangle	super-triangle	max-mean	-	-	-

Finally, each membership function can be represented by four parameters (a, b, c, d) ; every parameter is represented by four bits and its codification is shown in Table 2. For that reason, the updating parameters for the membership functions are represented by $16 * numPL * n$ bits, it means 16 bits for each membership function belonging to each feature. This part of the algorithm is the main extension to the work of Barajas and Reyes [2]. In Table 2 each value of the gene is associated with a value between 0.50 and 1.50, which represents a scaling value (SV) for to increase or decrease appropriately the parameters of fuzzy membership functions.

Table 2. Binary codification for optimizing parameters of the membership function

Gene	SV	Gene	SV	Gene	SV	Gene	SV
[0, 0, 0, 0]	0.50	[0, 1, 0, 0]	0.75	[1, 0, 0, 0]	1.20	[1, 1, 0, 0]	1.40
[0, 0, 0, 1]	0.55	[0, 1, 0, 1]	0.80	[1, 0, 0, 1]	1.25	[1, 1, 0, 1]	1.45
[0, 0, 1, 0]	0.60	[0, 1, 1, 0]	0.90	[1, 0, 1, 0]	1.30	[1, 1, 1, 0]	1.50
[0, 0, 1, 1]	0.70	[0, 1, 1, 1]	1.10	[1, 0, 1, 1]	1.35	[1, 1, 1, 1]	1.00

Initially, the membership functions are uniformly distributed and their parameters take a value into the feature's domain; the dot product with the value obtained by the bits is applied to that value, in that way the parameters for the membership functions are normalized for every specific problem.

3.2 Genetic Operations

The selection operation is done by tournament; more information on this operator is detailed in [4]. We use one point crossover for creating the new population. A one-point crossover operator randomly selects a crossover point in the two previously selected parents, and the bitstrings after that point are swapped between the two of them. A crossover probability of 0.8 was used in all experiments.

In addition, we use a mutation operator with the aim to introduce new genetic material into an existing individual. Similarly to crossover, mutation requires probability to be applied; usually this probability is low and in this case the mutation probability is 0.02.

3.3 Fitness Function

Fitness function is an important part for a genetic algorithm because it is in charge of evaluating the potential solutions. As a fitness function we use the balanced error rate; in that way we avoid to select parameters that have a good performance classifying only one class. The expression for the fitness function is:

$$BER = \frac{e_{(+)} + e_{(-)}}{2} \quad (8)$$

where BER is the balanced error rate, $e_{(+)}$ and $e_{(-)}$ are the missclassifications rates for the positive and negative classes, respectively.

4 Experiments and Results

In this work we experimented with the classification of infant cry databases built by the computer science department of the INAOE. The infant cries were collected by recordings done directly by medical doctors and then, each signal wave was divided in segments of 1 second; each segment represents a sample. For the experiments showed here, we used the acoustic obtained by [2], which

Table 3. Results of classifications using the proposed method

Data base	Accuracy	TPR	TNR	ROC
Asphyxia vs Normal	88.67%	90.00%	87.78%	92.85%
Deaf vs Normal	97.55%	98.75%	95.47%	99.75%
Hungry vs Pain	96.03%	95.59%	96.67%	98.35%

Table 4. Chromosome interpretation

Data base	Chromosome	NPL	TMF	NTE	LR	IT	WRM	TRP
Asphyxia vs Normal	[1, 0, 0, 0, 1, 1, 0, 0, 1, 0, 1, 1, 1, 0, 1, 0, 1, 0, 1]	3	Gaussian	30	0.20	0.50	0.30	Max-mean
Deaf vs Normal	[1, 0, 0, 1, 1, 1, 1, 0, 1, 0, 1, 1, 1, 1, 1, 0, 0, 0, 1, 1]	5	Gaussian	50	0.20	0.50	0.50	Supertriangle
Hungry vs Pain	[1, 0, 0, 1, 0, 0, 1, 1, 1, 0, 0, 1, 1, 1, 0, 1, 1, 1]	5	Gaussian	5	0.35	0.25	0.70	Max-mean

include Mel Frequency Cepstral Coefficients (MFCC). For the feature extraction process every sample of 1 second is divided in frames of 50 milliseconds and from each of which 16 coefficients were extracted, getting vectors with 304 coefficients by sample. The acoustic feature extraction process was done using Praat [3]. Then, we applied Principal Component Analysis to reduce the dimensionality of the vectors, for whose purpose we selected 16 principal components. The corpus has 157 samples of normal infant cry, 340 of asphyxia infant cry, 879 of deaf, 192 samples of pain cry and 350 of hunger cry. Pain and hunger samples were taken from normal babies.

The complete classification system is implemented in Matlab. For our experiments the initial population is formed by 25 individuals and the number of generations is set to 10.

We performed three different sets of experiments. In our first experiment, we classify between two classes: asphyxia and normal cries. The second one, the classification is between deaf and normal cries. Finally, in the third one, we classify between hunger and pain cries. The evaluation was done using 10 fold cross validation and the criteria for evaluation are: percent of correct classification (Accuracy), true positive rate (TPR), true negative rate (TNR) and area under ROC curve. The results of these experiments are showed in Table 3 and Table 4 shows the best chromosome in every case and their corresponding interpretation. Due to the variable dimensionality, the part of the chromosome corresponding to the parameters for fuzzy membership functions is not included in Table 4, but Figure 2 shows some examples of resulting fuzzy membership functions obtained by the algorithm for the best classification results in each case.

4.1 Comparisons with Other Work

Several research results have been published using the same database used by us, or a subset of it. For example, in the work of Suaste-Rivas et al. [10] normal, deaf and asphyxia cries are classified with a FRNN obtaining results up to 88.00%. A later implementation of the FRNN on FPGA was done by Suaste-Rivas et al. [8], also classifying normal, deaf and asphyxia cries and obtaining a generalization

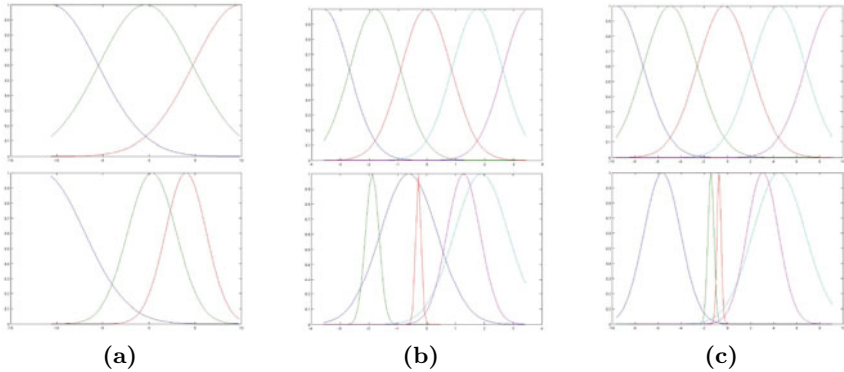


Fig. 2. Fuzzy membership functions (FMF). The images on top represent the initial FMF uniformly distributed, the lower are the FMF resulting after the evolutionary processing. (a) represents the FMF for one feature for asphyxia vs normal database, (b) for hungry vs pain database and (c) for deaf vs normal database.

rate of 94.61%. It is important to point out that these works did not use n -fold cross validation; just randomly selected training and testing sets were used.

In Barajas and Reyes’ work [2] the classification with the same databases was performed. Their reported results are showed in Table 5, which we use to compare our results. The best result reported for each case is in bold.

Table 5. Accuracy of comparison of our method and Barajas and Reyes [2]

Data base	Proposed method	[2]
Asphyxia vs Normal	88.67%	84.00%
Deaf vs Normal	97.55%	98.00%
Hungry vs Pain	96.03%	95.24%

5 Conclusions and Future Work

The proposed method can automatically determine all the optimal parameters for the Fuzzy Relation Neural Network, as well as the parameters for the membership functions. As can be observed in Table 5, the preliminary performance is comparable with similar systems, with the advantage that the designer does not have to deal with establishing neither the FRNN parameters nor the fuzzy membership functions parameters. In addition, by automatically finding the optimal membership functions as well as their parameters for an specific problem, the results can be improved.

For future work we will use another fuzzy relational products for the learning phase, given that currently training is done using max-min composition. We will also test the proposed system with larger databases including more classes to evaluate its performance.

References

1. Bandler, W., Kohout, L.: Fuzzy relational products as a tool for analysis and synthesis of the behaviour of complex natural and artificial systems. In: *Fuzzy sets: theory and applications to policy analysis and information systems*, pp. 341–367 (1980)
2. Barajas, S.E., Reyes, C.A.: Your Fuzzy Relational Neural Network Parameters Optimization with a Genetic Algorithm. In: *The 14th IEEE International Conference on Fuzzy Systems FUZZ 2005*, pp. 684–689. IEEE, Los Alamitos (2005)
3. Boersma, P., Weenink, D.: Praat, a system for doing phonetics by computer. Institute of Phonetic Sciences of the University of Amsterdam, Report 132, 182 (1996)
4. Engelbrecht, A.: *Computational intelligence: an introduction*, 2nd edn. Wiley, Chichester (2007)
5. Holland, J.: *Adaptation in natural and artificial systems*. University of Michigan Press, Ann Arbor (1975)
6. Pedrycz, W.: Neurocomputations in relational systems. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 13(3), 289–297 (1991)
7. Reyes, C.: On the design of a fuzzy relational neural network for automatic speech recognition. Ph.D. thesis, Doctoral Dissertation, The Florida State University, Tallahassee, FL (1994)
8. Suaste-Rivas, I., Díaz-Méndez, A., Reyes-García, C., Reyes-Galaviz, O.: Hybrid Neural Network Design and Implementation on FPGA for Infant Cry Recognition. In: *Text, Speech and Dialogue*, pp. 703–709. Springer, Heidelberg (2006)
9. Suaste-Rivas, I., Reyes-Galaviz, O.F., Diaz-Mendez, A., Reyes-Garcia, C.: A Fuzzy Relational Neural Network for Pattern Classification. In: *Progress in Pattern Recognition, Image Analysis and Applications*, pp. 275–299 (2004)
10. Suaste-Rivas, I., Reyes-Galviz, O.F., Diaz-Mendez, A., Reyes-Garcia, C.: Implementation of a linguistic fuzzy relational neural network for detecting pathologies by infant cry recognition. In: *Advances in Artificial Intelligence—IBERAMIA 2004*, pp. 953–962 (2004)

Speech Compression Based on Frequency Warped Cepstrum and Wavelet Analysis

Francisco J. Ayala and Abel Herrera

Digital speech processing laboratory, Universidad Nacional Autónoma de México,
Facultad de Ingeniería
Ciudad Universitaria, 04510, Mexico City, Mexico
poasf01@cancun.fi-a.unam.mx,
abelhc@hotmail.com

Abstract. In this article it is described the process to extract a set of cepstral coefficients from a warped frequency space (mel and bark) and analyze the perceived differences in the reconstructed signal. We will try to determine if there is any audible improvement between these two most used scales for the purpose of speech analysis by synthesis. We will use the same procedure for parameter extraction and signal reconstruction for both functions, replacing only the warping scale. The proposed system is based on a basic cepstral analysis synthesis model on the mel scale, whose excitation signal generation process has been changed. The inverse MLSA filter was obtained in order to generate the analysis signal, then this signal is fed into a wavelet decomposition block and the resultant coefficients are sent to the decoding system where the excitation signal is reconstructed. Furthermore the mel scale is replaced by bark scale.

Keywords: speech compression, speech encoding, wavelet analysis, warped cepstrum.

1 Introduction

The mel scale was proposed at 1937, following a series of experiments used to establish a perceptual scale based on the perception of tones, the use of mel scale is almost standard for speech recognition application. The Bark scale, proposed by Eberhard Zwicker in 1961, divides the audible spectrum into 24 critical bands that try to mimic the frequency response of the human ear.

Given the characteristics of the human auditory system (nonlinear and time variant), the models needed to represent auditory perception are complex since they involve the use of non uniform frequency scales instead of linear scales such as the Hertz scale. The mel and bark scales are examples of non uniform scales, the use of them is desirable in low rate speech coding systems [1].

Cepstral analysis is performed for coefficients extraction, and then a frequency warping process is applied in order to change the frequency scale of the coefficients. The mel log spectrum approximation filter (mlsa) computes the synthetic

signal [2]. This filter not only works on the mel scale but also on bark scale by just replacing the allpass parameter.

The inverse mlsa filter provides the analysis signal to which other coefficients are extracted to generate the excitation signal of the mlsa filter. This process involves the wavelet analysis. The wavelet coefficients perform well in the excitation signal generation process. Thus, as important is the quality of the cepstral coefficients used to model the mlsa filter, as the performance of the excitation signal.

2 Cepstral Analysis Synthesis

The filter coefficients are obtained through a linear transform from the warped cepstrum defined as the fourier cosine coefficients of the warped log spectrum of speech [1].

The nearest part to the origin of the cepstrum corresponds to the transfer function of the vocal tract and can be used to approximate the spectral envelope of the signal [3]. Hence the linear transform consist of a homomorphic filtering process in order to extract the first M cepstrum elements.

The mel and bark scales can be approximated by the phase characteristics of a first order allpass filter. The transfer function is given by [4]:

$$\tilde{z} = \frac{z^{-1} - \alpha}{1 - \alpha z^{-1}}, |z| < 1. \quad (1)$$

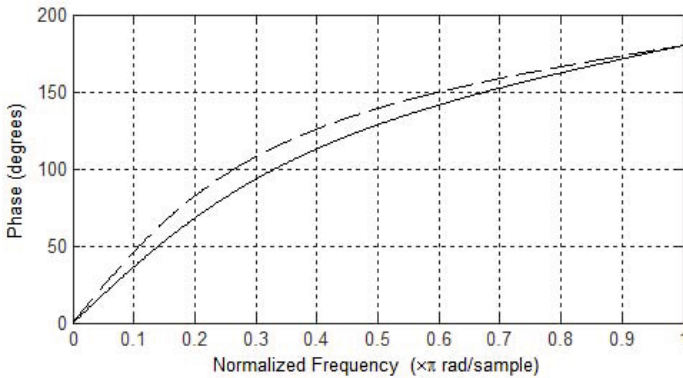


Fig. 1. Phase response of the all pass filter when $\alpha=0.4582$ (dashed line) and $\alpha=0.35$ (solid line)

The cepstrum is fed into an allpass filters chain so that the frequency scale becomes non uniform [1]. For certain values of α , the frequency transformation either resembles the mel scale or the bark scale. It is stated the value $\alpha=0.35$ to approximate the mel scale and $\alpha=0.4582$ for bark scale at 10kHz of sampling frequency [5].

The mlsa filter defined by

$$H(z) = \exp \sum_{m=0}^M \tilde{c}(m) \tilde{z}^{-m} \quad (2)$$

takes the cepstral coefficients and an excitation signal to generate the synthetic speech. Since the coefficients are transformed into a different frequency scale, the mlsa filter is designed over the warped frequency scale, thus each delay element is replaced by the first-order allpass filter. This substitution implements the unwarping while the filter is working [4].

From the inverse mlsa filter

$$\frac{1}{H(Z)} = \exp \sum_{m=0}^M -\tilde{c}(m) \tilde{z}^{-m} \quad (3)$$

it is obtained the analysis signal which is sent to a wavelet analysis process. The purpose of this step is to generate the excitation signal.

3 Wavelet Analysis

The basis of this process is a set of quadrature mirror filters and the discrete wavelet transform [6]. In wavelet analysis, the signal is decomposed into approximations and details. The approximations are the low-frequency components of the signal. The details are the high-frequency components.

The decomposition process is achieved by iterations; one signal is broken down into many lower resolution (by dyadic decimation) components in each iteration [7].

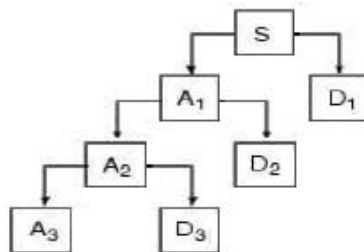


Fig. 2. Decomposition tree. For reconstruction: $S=A_3+D_1+D_2+D_3$.

Given a signal of length N, the discrete wavelet transform DWT consists of four levels at most (for a high quality reconstruction in this approach). The first step produces two sets of coefficients: approximation coefficients A1, and detail coefficients D1. These vectors are obtained by convolving the signal with the low-pass filter for approximation, and with the high-pass filter for detail, followed by decimation.

The second step divides the approximation coefficients A1 in two parts repeating the same procedure, replacing the input signal by A1 and producing A2 and D2, and so on. After these steps a wavelet decomposition four-level tree is obtained.

For the reconstruction process the inverse discrete wavelet transform IDWT is applied to the approximation and detail coefficients. The coefficients vectors are upsampled and filtered. The vectors are zero-padded for the upsampling step and for recovering the original size of the signal at the end of iterations. For filter design the wavelet standardized db2 coefficients are taken.

The filter coefficients are

- (a) Low pass decomposition filter: $h(n)=-0.1294, 0.2241, 0.8365, 0.4830$
- (b) High pass decomposition filter: $h(n)=-0.4830, 0.8365, -0.2241, -0.1294$
- (c) Low pass reconstruction filter: $h(n)=0.4830, 0.8365, 0.2241, -0.1294$
- (d) High pass reconstruction filter: $h(n)=-0.1294, -0.2241, 0.8365, -0.4830$

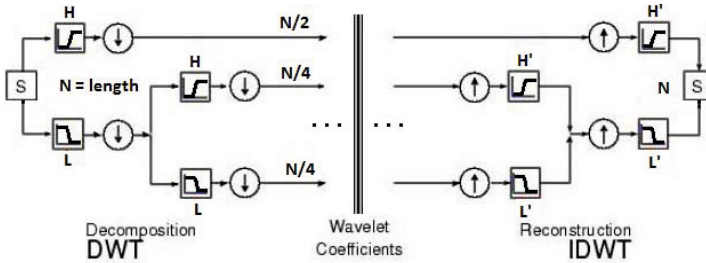


Fig. 3. (QMF) Quadrature mirror filters. Multiple level analysis-synthesis process.

In chapter 5 it is explained how this analysis is applied to the analysis signal.

4 Detection of Fricative Sound Block

Given the loss of details after compression, an improvement of voice naturalness and of intelligibility of fricative consonants can be achieved by adding noise of different band widths and amplitudes to the excitation signal.

For naturalness improvement, a low and constant amplitude noise is added to the entire excitation signal, and for intelligibility improvement of fricative sounds, a higher amplitude noise is only added to the corresponding frame. One bit is sent to indicate the presence of a high amount of zero crossings and a relative low energy of the current frame of the original speech.

In the decoder system, a white noise is divided in four bands of frequency. The band edges are given in Hertz as [500 1000 2500 3500 5000]. In original signals, the average of amplitudes corresponding to the lost information in synthetic speeches was estimated in the four bands in which the original signals were divided. Those amplitudes are the gain of the normalized noise to be added to

the excitation signal. Although the speech characteristics vary frame to frame, the results are actually good.

When the decoder receives a bit indicating the presence of fricative sounds, a white noise is added to the current frame. The noise amplitude is the average of amplitudes in original speeches that contain kind of unvoiced sounds. This block would not be necessary if a high level compression were not desired since the more compressed is the signal the higher the loss of details.

5 Design of the Coding-Decoding System

5.1 Encoding Process

A block diagram of the warped cepstral analysis synthesis system is shown in Fig. 4. First, the M cepstral coefficients are extracted as explained in chapter 2 to each 256-sample sequence. Also each sequence is fed into the inverse mlsa filter to obtain at its output the analysis signal. Then, the cepstral parameters are quantized and transmitted and the analysis signal is decomposed in wavelet coefficients.

After calculating the wavelet transform of the analysis signal it is found that most of the coefficients have small magnitudes, they are close to zero. Consequently, compression involves truncating coefficients below a threshold.

The low-frequency components are the most important part of human voice. When high-frequency components are removed the speech is still intelligible but sounds a little different. For that reason only the detail coefficients are truncated.

Around 90 % of the wavelet coefficients are found to be small and their truncation to zero make a barely perceptible difference to the signal.

In this work, for higher compression all the detail coefficients can be truncated leaving the approximation coefficients of the last level of the decomposition tree. If higher quality of synthetic speech is desired, less compression is needed and some of the details coefficients must be left.

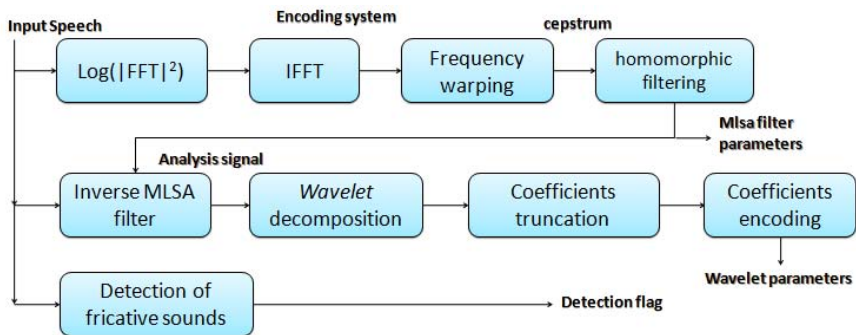


Fig. 4. Encoding system

The non-zero coefficients are stored in one vector; in a second vector is stored the starting position of a string of zeros and the number of zeros. Then the coefficients of the vectors are quantized and transmitted.

5.2 Decoding Process

The decoding system takes the cepstral and wavelet coefficients. Interpolation of the cepstral parameters of two successive frames is performed in order to smooth the transition of synthetic frames.

The interpolated cepstral coefficients are set as the mlsa filter parameters, which are unwrapped in the filter using the negative magnitude of α [4].

The excitation signal is reconstructed from the wavelet coefficients by the wavelet reconstruction process where the vectors are zero padded and convolved with the reconstruction filters in each stage [7].

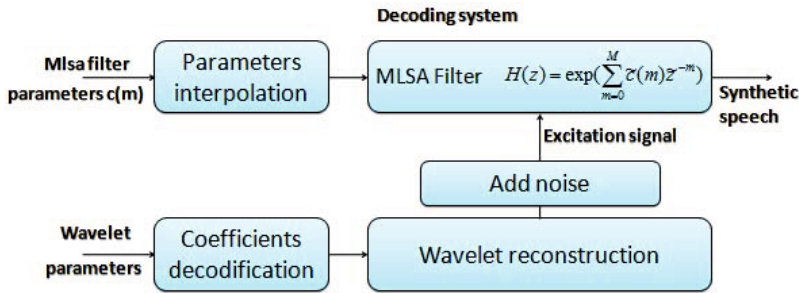


Fig. 5. Decoding system

6 Quantization of the Filter Parameters

MLSA filter parameters are quantized according to their characteristics. It is shown from experimental results that the maximal magnitude of the first coefficient is 8 (for $m=1$) and the maximal magnitude of the rest of the coefficients is 1. The first parameter is truncated to a 3-bit integer. The second set of parameters magnitudes are normalized according to the interval $[0 \ 1]$. This allows the use of only one code book in the decoder. One additional bit is used to represent the sign. The maximal absolute value, of each set of coefficients, is transmitted to restore its original magnitude. No more than 8 bits should be taken for this value.

It is also shown from experimental results that the difference between the maximal and minimal values of each wavelet parameter does not exceed 1. The same normalization procedure is applied to these coefficients which are mapped into the interval $[0 \ 1]$. The same code book is used. The maximal magnitude of each set of parameters is not transmitted. Experiments show that there is no distortion whether it is recovered or not the original magnitude of wavelet parameters.

Table 1. Bit allocation for this coder

Parameter	Resolution (bits/parameter)
Cepstral (sign included)	4
First cepstral parameter	3
Maximal value (cepstral)	8
Wavelet (sign included)	5
Bit of unvoiced frame	1

The selected number of bits for both the wavelet and cepstral coefficients depends on the level of desired compression and quality of synthetic speech. In Table 1 it is shown the proposed bit allocation.

7 Speech Quality and Bit Rate

In order to evaluate the quality of the synthetic speech, short (two seconds) and large (ten seconds) English sentences were recorded to be analyzed and synthesized.¹

Given the parameters:

- (a) F_s : sampling frequency. T : frame duration.
- (b) M : cepstrum order.
- (c) B_c : bits/cepstrum coefficient.
- (d) W : number of wavelet coefficients.
- (e) B_w : bits/wavelet coefficient.
- (f) B_M : bits/Maximal value.
- (g) B_F : bits/First cepstral parameter.
- (h) α : warping parameter.

The overall bit rate B of this coder is calculated by

$$B = \frac{[(M-1) \cdot b_c + W \cdot b_w + b_M + b_F]}{T}. \quad (4)$$

For $F_s=10\text{kHz}$, $T = 25\text{ms}$, $M=26$, $b_c=4$, $W=34$, $b_w=5$, $b_M=7$, $b_F=3$ and $\alpha=0.35$ (mel scale) the data rate is $B=10.9\text{ kbit/s}$. The speech quality is quite good and the intelligibility is very high. The speaker is clearly recognizable and the signal gets naturalness.

For $F_s=10\text{kHz}$, $T = 25\text{ms}$, $M=26$, $b_c=4$, $W=18$, $b_w=5$, $b_M=7$, $b_F=3$ and $\alpha=0.35$ (mel scale) the data rate is $B=7.8\text{ kbit/s}$. The speech quality is good and the intelligibility is high. The speaker is clearly recognizable but the signal loses naturalness.

A MOS test was applied to 10 people. Each person heard (using headphones) and scored the seven recordings separately. The MOS score range is from 1 to 5, 1 being the worst and 5 the best. The average range of each evaluation is shown in Table 2.

¹ Recordings were performed in a quiet laboratory, using a general purpose unidirectional microphone and the sound card of a computer.

Table 2. MOS Test results

Signal	7.8 kbit/s (mel)	10.9 kbit/s (mel)	7.8 kbit/s (bark)	10.9 kbit/s (bark)
1 (male)	3.60	4.4	3.50	4.5
2 (female)	3.9	4.2	3.90	4.2
3 (male)	3.30	4.6	3.44	4.6
4 (male)	3.30	3.9	3.34	3.8
5 (female)	3.37	4	3.35	4
6 (female)	2.90	3.8	2.94	3.9
7 (female)	2.90	3.9	3.50	3.9
Average	3.32	4.11	3.42	4.12

8 Conclusion

The differences between the mel and bark scale is almost imperceptible. As the bit rate increases the differences are completely imperceptible. For a too low bit rate there exist some audible differences between the scales but those are not statistically significant.

The quality of the synthetic speech is actually good; the performance of the coder is fair since the presence of noise does not affect it. The decoder is able to reconstruct any kind of noise but not with the same quality given to speech signals.

The spectral distortion of the parameters given the quantization process is too low. When comparing a synthetic speech generated with quantized parameters and another synthetic speech generated with non-quantized parameters there is not perceptible differences.

The proposed system performs quite good, this system is able to synthesize music, street noise (cars, airplanes, people, etc). No matter what is behind someone's voice, because it is always understandable after being compressed under this coding system.

References

1. Harma, A., Karjalainen, M.: Frequency-Warped Signal Processing for audio Applications. In: 108th AES convention Paris, France (2000)
2. Imai, S.: Cepstral analysis synthesis on the mel frequency scale. In: IEEE International Conference on Acoustics, Speech, and Signal Processing ICASSP 1983, vol. 8, pp. 93–96 (1983)
3. Acero, A., Hon, H.-W.: Spoken language processing: a guide to theory. Algorithm and system development (2001)
4. Smith, J.O., Abel, J.S.: Bark and ERB bilinear transforms. In: IEEE Speech and Audio Processing, vol. 7, pp. 697–708 (1999)
5. Tokuda, K., Kobayashi, T.: Recursive Calculation of Mel-Cepstrum from LP Coefficients (April 1994)
6. Mallat, S.: A wavelet tour of signal processing, pp. 255–263 (1999)
7. Shivraman, G., Nilesh, N.: Speech compression using wavelets. Department of electrical engineering, Veermata Jijabai Technological Institute, University of Mumbai, pp. 29–54 (2002)

Dust Storm Detection Using a Neural Network with Uncertainty and Ambiguity Output Analysis

Mario I. Chacon-Murguía¹, Yearim Quezada-Holguín¹,
Pablo Rivas-Perea², and Sergio Cabrera²

¹ DSP & Vision Laboratory, Chihuahua Institute of Technology, Chihuahua, Mexico
mchacon@itchihuahua.edu.mx

² ECE, University of Texas at El Paso, USA
privas@miners.utep.edu, sergioc@utep.edu

Abstract. Dust storms are meteorological phenomena that may affect human life. Therefore, it is of great interest to work towards the development of a stand-alone dust storm detection system that may help to prevent and/or counteract its negative effects. This work proposes a dust storm detection system based on an Artificial Neural Network, ANN. The ANN is designed to identify not just dust storm areas but also vegetation and soil. The proposed ANN works on information obtained from multispectral images acquired with the Moderate Resolution Imaging Spectroradiometer (MODIS) instrument. Before the multispectral information is fed to the ANN a process to remove cloud regions from images is performed in order to reduce the computational burden. A method to manage undefined and ambiguous ANN outputs is also proposed in the paper which significantly reduces the false positives rate. Results of this research present a suitable performance at detecting the dust storm events.

Keywords: Dust storm detection, image segmentation, neural network output analysis.

1 Introduction

In recent years weather conditions have attracted the attention of the international community. Different countries have suffered the consequences of natural phenomena like earthquakes, tsunamis, floods, hurricane, drought, dust storms, which significantly affect human life as well as the economy of those countries. These climate situations have promoted a new interest on remote sensing research because it offers the potential of better understanding of these phenomena. One of these natural phenomena that have been studied in recent years is dust storm. Dust storms are seasonal meteorological phenomena. Usually, dust storms occur in arid and semiarid regions and can travel long distances. The dust storm may have a negative effect in our planet and human life in different ways like; cloud formation, respiratory illness, aerial and terrestrial transportation. This phenomenon can also damage crops and may cause fertile soil erosion [1]-[3]. Research on dust storm is of great interest because it can help to find methods to prevent and/or counteract its negative effects. This paper

presents the development of an Artificial Neural Network, ANN, classifier intended to detect the presence of a dust storm in satellite multispectral images. The paper is organized in the following sections. Section 2 presents a literature review of this topic. The dust storm data used in this work is described in Section 3. Analysis of the classifier is covered in Section 4. Finally the results and conclusions of the research are commented in Section 5.

2 Literature Review

In order to establish some aspects of the classifier design, a literature review was achieved considering the following points: type of instrument used to acquire the multispectral images, methods used for dust storm detection, performance metrics, and event location. From 43 papers related to the topic of dust storm and satellite multispectral images only 13 were considered for the analysis because only these papers were related specifically to dust storm detection. With respect the type of sensor used, works reported in [4] to [9] use the Moderate-Resolution Imaging Spectroradiometer instrument, commonly named MODIS sensor. The researches in [6] and [7] work with the Aqua MODIS. The sensor AERONET and the AVHRR are used in [10] and [11] respectively. The work reported in [12] employs the CALIPSO and in [13] the MERIS sensor is used to detect dust storm in sand regions. The MFR-7 sensor is mentioned in [14]. Also in [9] it is reported a combination of information obtained from the sensors MODIS and TOMS. Finally, the MISR sensor is used in [15]. With regard to the type of image processing techniques used, the information is the following. All works are based on pixel level feature extraction processing, except in [9] and [10] where a window feature extraction processing is used instead. The method used to evaluate the performance of the dust storm detection systems was only qualitative. The dust storm events reported in the literature were located at; South Korea, China, Mongolia, India, Egypt, Senegal and East of Africa. In some of these works it was necessary to adjust some of the parameters of the method developed in order to obtain correct results.

Considering the information found in the literature review the following points were determined. It was decided to use the MODIS sensor because it has a good spectral resolution, 36 bands from 0.62 μm to 14.382 μm . The temporal resolution is 15 minutes and its spatial resolution 1 Km. The image processing technique selected was pixel level processing because most of the reported works that used this technique achieved better results than the two papers that used window based processing.

3 Data Analysis

3.1 Database

The region of interest for dust storm detection is defined as the north region of Chihuahua State in Mexico and the southwest area of Texas State in the USA. The dust storm events were acquired from the web page <http://ladsweb.nascom.nasa.gov/data/search.html>. These events were acquired by the MODIS sensor. Eight events were downloaded and their information is given in Table 1.

Table 1. Dust Storm Event Information

Date	Hour	Date	Hour
1. April/6/2001	18:30 hrs.	5. April/15/2003	17:10 hrs.
2. April/10/2001	18:05 hrs.	6. April/15/2003	17:15 hrs.
3. July/2/2002	17:55 hrs.	7. April/15/2003	18:50 hrs.
4. December/17/2002	18:45 hrs.	8. November/22/2003	18:20 hrs.

3.2 Band Selection

Not all the information in the bands of the multispectral images is related to dust storm events. In order to determine which bands will be used in the design of the classifier a literature review was achieved. The bands B31 and B32 of the MODIS sensor were related to dust storm information in works reported in [5]-[9]. The bands 4 and 5 of the sensor AVHRR were used for dust storm detection in [1] and [2] but these bands correspond to the bands 31 and 32 of the MODIS sensor. In [6] the band B29 of the MODIS was incorporated because it provides extra information in clear days. Based on these evidences we concluded to use the bands B29, B31 and B32.

3.3 Data Selection

In order to obtain reliable samples for the neural network design, data was statistically selected for each class. The classes considered in this work were dust storm D , vegetation V , and soil S . The samples selected correspond to samples obtained from regions of each class that satisfy the following criterion

$$D_{pc} \leq D_c \quad \text{for } c = \{V, D, S\} . \quad (1)$$

where D_{pc} is the distance of the candidate sample p_{Bc} of the band B of the region c to the mean value of the sample mean, μ_{Bnc} , of the band B in the region c ,

$$D_{pc} = \sqrt{(\mu_{B1c} - p_{B1c})^2 + \dots + (\mu_{Bnc} - p_{Bnc})^2} . \quad (2)$$

D_c is the maximum tolerance distance, one standard deviation σ_{Bnc} , of a sample of a region c to the sample mean of the region.

4 Dust Storm Detection

This section describes the process of the ANN classifier design including a step to detect clouds present in the images and then it follows with the design of the ANN classifier.

4.1 Cloud Detection

Multispectral images in most of the cases represent huge computational burden, therefore it is recommended to reduce the amount of information as much as possible. It was reported in the literature [6], that presence of clouds is an important perturbation that affects the performance of several dust storm detection methods. Thus, cloud detection is

a good alternative to reduce the computation load of dust detection algorithms and at the same time to get rid of possible perturbations. The work in [4] reports that cloud information can be eliminated using bands B_3 and B_7 . The work shows that the maximum energy related to clouds is captured by band B_3 . Meanwhile energy of other elements is minimum in this band. Band B_7 , contrary to band B_3 , capture minimum energy related to clouds and high energy the other elements. Based on the previous information, it is proposed a *difference index* $I_D = B_7 - B_3$ to eliminate cloud information. Using this index a mask image, I_M , can be generated to eliminate the cloud pixels.

$$I_M = \begin{cases} 0 & \text{if } I_D < 0 \\ 1 & \text{if } I_D > 0 \end{cases} \quad (3)$$

Other important result is that the I_D of water pixels will result in zero or near to zero values. Therefore water pixels will be also eliminated. The application of this process is illustrated in Figure 1, event #1 April/6/2001. The original information is shown in Figure 1a, the mask image is shown in Figure 1b and the resulting information of the event is shown in Figure 1c. The cloud elimination process presents some issues. On one hand, small clouds are not completely eliminated. On the other hand, some dark vegetation and soil regions are eliminated. However dust storm information is not significantly affected by this process. An example of this screening is shown in Figures 1d-f.

4.2 Classifiers Design

The classifier was designed to recognize three classes: vegetation, dust storm and soil; unlike other works that only consider a two-class problem dust storm and not dust storm. It was decided to include the class vegetation and soil because they may be of interest in a future work.

The structure of the neural network was determined by selecting the best network in a set of experiments changing the number of layers, 2 and 3 layers, and the number of neurons in each layer. The best neural network was defined with respect the best performance associated with the smaller number of layers and neurons by layer. The

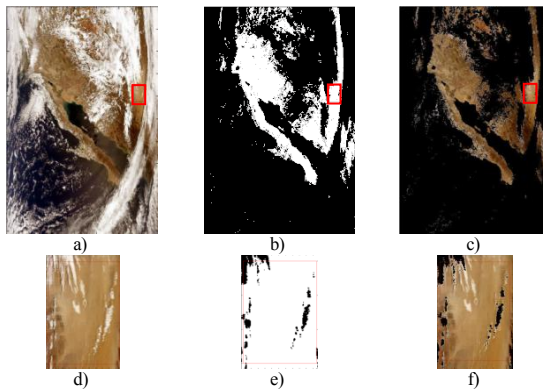


Fig. 1. a) Original image event April/6/2001, b) I_M mask, and c) Image without clouds and water. d), e) and f) zoom-in of the red rectangles in a), b) and c)

ANN model selected for the classifier is a 2-layer Feed-Forward neural network with 15 neurons in the hidden layer, and 3 neuron outputs. The activation functions are sigmoidal. The training was performed with the backpropagation algorithm using the *Scaled Conjugate Gradient Algorithm* since this algorithm was more effective than other gradient methods. Among their advantages we can mention that it does not depend on user parameters like learning rate and momentum. The first classifier, C29, was designed with the information provided by the bands B29, B31 and B32. The number of samples used was 5383; 721 samples of vegetation, 1493 of dust storm and 3169 of soil. The training samples correspond to the 70% of the total number of samples, 15% for validation and 15% for testing. The performances of this ANN were 95.8%, of correct classification for training, validation and testing. Table 2 shows the confusion matrix of the design process. It can be observed that the class dust storm has the best classification performance considering actual and prediction conditions 97.45% followed by Soil, 96.95% and vegetation 89.4%. The total performance is 96%. Figure 2 shows some visual results obtained with this classifier for the events #2, and 8, corresponding to April/10/2001, and November/22/2003 respectively. The result of the event April/10/2001 presents some false positives areas close to the dust storm region. The event April/15/2003 shows a correct detection, but the event of November/22/2003 involves many false positives regions.

Results on non-storm events are not shown because the work is related to the same area; therefore data of these cases is already incorporated and tested on the different dust storm events.

Table 2. Confusion matrix of the classifier C29

	Vegetation	Dust Storm	Soil	Total
Vegetation	578	0	8	98.6%
Dust Storm	12	1450	20	97.8%
Soil	131	43	3141	94.8%
Total	80.2%	97.1%	99.1%	96%

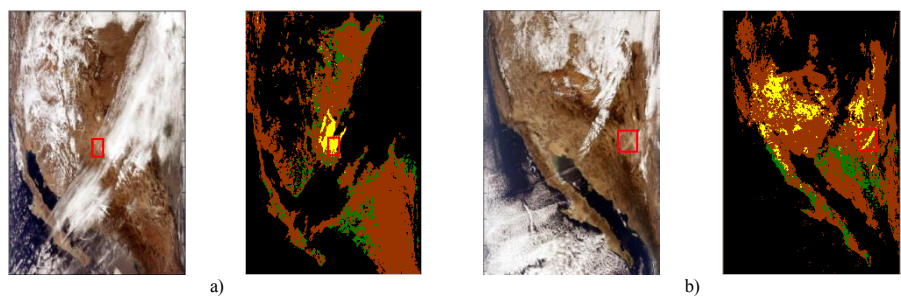


Fig. 2. Results of the classifier C29 for the events: a) April/10/2001, b) November/22/2003. Cloud /water black, vegetation green, storm yellow, soil brown

4.3 C29 Output Analysis

The results shown in the previous section led us to investigate the neuron outputs. The previous results were generated by the following rule *Assign pixel p_B to class i if $O_i > O_j$ for $i, j = \{v, d, s\}$ $i \neq j$* , where v , d , and s stands for the classes vegetation, dust storm and soil respectively. This rule assigns a pixel to the class with the higher value of the neuron outputs. This kind of decision is generally used in some works. However, this is not a guarantee that a good decision is taken. We can analyze two hypotheses: the first one supposes that three or two neuron outputs are high but both with close values; the second hypotheses could be when all outputs are low. In both cases we can determine a winner output but it does not mean the decision will be correct. From this analysis we can propose two cases. An undefined case, when the maximum output of the ANN is less than say 0.6, that is

$$\max(O_i < 0.6) \text{ for } i = \{v, d, s\} . \quad (4)$$

In this situation we can say that the ANN does not have a strong output in any of its outputs, therefore the class is undefined. The 0.6 threshold was determined by considering the output of the neurons as an effectiveness class definition percentage, the value 0.5 represents high vagueness. The second case is when the difference between two neuron outputs used to make a decision is less than 0.3,

$$|O_i - O_j| < 0.3 \text{ for } i = \arg\max O_K \text{ and } j = \arg\max O_K \quad k = \{v, d, s\} \quad i \neq j . \quad (5)$$

In this circumstance, the outputs of the ANN are so ambiguous that a decision is not recommended. Using the previous cases two new images can be generated to analyze the two hypotheses, $I_u(x, y)$ for undefined outputs and $I_A(x, y)$ for ambiguous outputs

$$I_u(x, y) = \{p_B(x, y) : \max(O_i) < 0.6 \text{ for } i = \{v, d, s\} \} . \quad (6)$$

$$I_A(x, y) = \{p_B(x, y) : |O_i - O_j| < 0.3 \text{ for } i = \arg\max O_K \text{ and } j = \arg\max O_K \quad k = \{v, d, s\} \quad i \neq j \} . \quad (7)$$

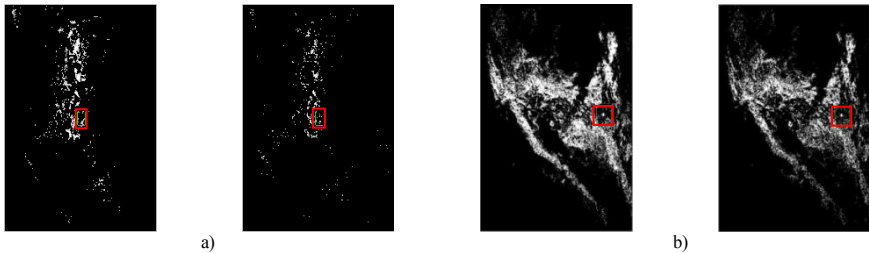


Fig. 3. $p(x, y) \in I_u(x, y)$ and $p(x, y) \in I_A(x, y)$ of a) April/10/2001, b) November/22/2003

Figure 3 illustrate the pixels $p(x,y) \in I_u(x,y)$ and $p(x,y) \in I_A(x,y)$ in white. Information shown in Figures 3 indicates that many pixels fall in the undefined and ambiguous cases. Therefore, assigning a pixel to a class under the highest output criterion is not a recommended criterion. In other analysis it was found that in most cases pixels corresponding to dust storm regions are well defined.

4.4 C29 ANN Output Adjustment

A new criterion to determine a winning neuron was defined based on the analysis described in Section 4.3. A wining neuron is the neuron with values greater than 0.6 and with a difference greater than 0.3 with respect the other outputs. The new rule is, *Assign pixel p_B to class i if $O_i > 0.6$ AND $|O_i - O_j| > 0.3$ for $i, j = \{v, d, s\}$ $i \neq j$.* The color map to describe the new output of the ANN is as indicated in Table 3. Using this new criterion the events were classified again. In most of the cases the results are better because the region of the dust storm is better defined by the new criteria and false positives are reduced as seen in the blue rectangle in Figure 4. The previous process confirm that when undefined outputs and ambiguous outputs are analyzed in the classification processes they positively contribute in the performance of the ANN classifier without negatively effecting the correct detection of the dust storm region. Table 4 shows improvements achieved in the dust storm detection task using the new criteria on the ANN output.

Table 3. Color map of the new ANN outputs

Case	Description	Color
Weak output	Less than 0.6	
Undefined	Outputs with a difference less than 0.3	
Both	The two previous cases	

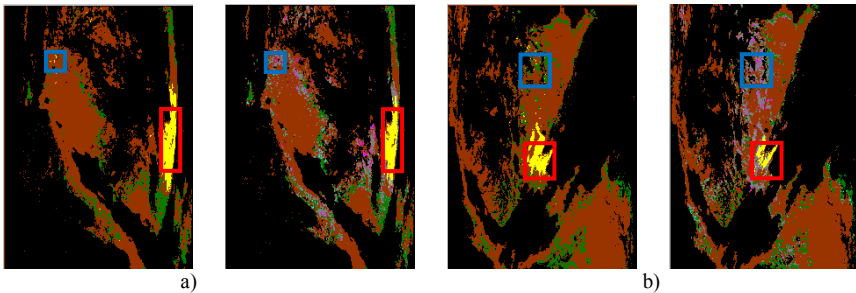


Fig. 4. First result and result with the new criteria for events a) April/6/2001, b) April/10/2001

Table 4. Improvements with the new criteria on the ANN output

Event	Problems using the original criterion	Improvements with the new criterion
April/6/2001	Disperse dust storm false positives.	Less disperse false positives.
	False positives around the dust storm region.	The dust storm region was better defined.
April/10/2001	Disperse dust storm false positives.	Less disperse false positives.
	False positives around the dust storm region.	The dust storm region was better defined.
July/2/2002	The dust storm was not detected. (the event is to weak)	Same problem.
December/17/2002	Disperse dust storm false positives.	Less disperse false positives.
April/15/2003	There are not problems.	There are not problems.
April/15/2003	False positives around the dust storm region.	The dust storm region was better defined.
April/15/2003	Disperse dust storm false positives.	Less disperse false positives.
	The dust storm was not detected.	The dust storm was detected.
November/22/2003	Many disperse dust storm false positives.	Less disperse false positives.

5 Results and Conclusions

Findings in this research indicate that the cloud detection method is a good alternative to reduce the computational burden as well as to get rid of possible perturbations that may cause a negative effect in the performance of the dust storm detector. The ANN classifier presents a suitable performance at detecting the dust storm events. The dust storm events were detected by the method in all events analyzed in this work except in the event #3 where it was not possible to detect the event. This event is also complicated to be perceived by a human observer. In regard to the false detection rate, we can say that it is tolerable because the cost of the event miss is higher than the false detection error cost. The performance of the ANN classifier is also acceptable, at least under a visual evaluation, at detecting the other two classes, vegetation and soil. The proposed method to manage undefined and ambiguous ANN outputs proved to be an important contribution in this research as the false positives were significant reduced without causing negatives hits in the dust storm area.

In conclusion, the proposed dust storm detector system based on ANN can produce preliminary information related to dust storm detection that may be used for posterior analysis. Moreover, the statistical selection of a relevant training dataset, allowed the construction of a low-complexity ANN model. To overcome uncertainty and ambiguity in ANN outputs, adjustments were made and different criteria were established, which produced higher accuracy rates, as well as a decrease in false positives count. However, false positive issues is an important point that needs more work if the system is intended to be used as a stand-alone dust storm system.

Acknowledgments. This work was supported by SEP-DGEST, ITCH, UTEP, and partially supported by CONACYT under grant 193324, SEP-DGRI, and Texas Instruments Foundation endowed scholarship.

References

1. Swapna, J.: Dust Storm Detection Using Classification And Filter Banks. M.S. thesis, University Of Texas At El Paso, El Paso, Texas, USA (2008)
2. Rivera, N.: Detection And Characterization Of Dust Source Area In The Chihuahuan Desert, Southwestern North America. M.S. thesis, University Of Texas At El Paso, El Paso, Texas, USA (2006)
3. Rivas, P., Rosiles, J., Chacon, M.I.: Traditional and Neural Probabilistic Multispectral Image Processing for the Dust Aerosol Detection Problem. In: 2010 IEEE Southwest Symposium on Image Analysis and Interpretation, pp. 169–172 (2010)
4. Qu, J., Xianjun, H., Wang, W., Wang, L., Kafatos, M.: A Study of African Dust Storm and Its Effects on Tropical Cyclones over Atlantic Ocean from Space. In: Geoscience and Remote Sensing Symposium, vol. 4, pp. 2715–2718 (2005)
5. El-Askary, H., Sarkar, S., Kafatos, M.: A Multisensor Approach to Dust Storm Monitoring Over the Nile Delta. *IEEE Transactions on Geosciences and Remote Sensing* 41, 2386–2391 (2003)
6. San-chao, L., Qinhuo, L., Maofang, G., Liangfu, C.: Detection of Dust Storms by Using Daytime and Nighttime Multi-spectral MODIS Images. In: IEEE International Geosciences and Remote Sensing Symposium, pp. 294–296 (2006)
7. Qu, J., Xianjun, H., Kafatos, M., Wang, L.: Asian Dust Storm Monitoring Combining Terra and Aqua MODIS SRB Measurements. *IEEE Geosciences and Remote Sensing Letters*, 484–486 (2006)
8. Tao, H., Yaohui, L., Hui, H., Yongzhong, Z., Yujie, W.: Automatic Detection of Dust Storm in the Northwest of China Using Decision Tree Classifier Based on MODIS Visible Bands Data. In: IEEE Geosciences and Remote Sensing Symposium, pp. 3603–3606 (2005)
9. El-Askary, H., Kafatos, M.: Potential for Dust Storm Detection Through Aerosol Radioactive Forcing Related to Atmospheric Parameters. In: Geosciences and Remote Sensing Symposium, pp. 620–623 (2006)
10. Kaufman, Y.J., Tam, D., Dubovik, O., Karnieli, A., Remer, L.A.: Absorption of Sunlight by Dust as Inferred From Satellite and Ground-based Remote Sensing. *J. Geophysical Research Letters* 28(8), 1479–1482 (2001)
11. Xinping, B., Zhenxin, S.: The Estimation of Dust Aerosol Sources for the Numerical Simulation of Asian Dust Storms Observed in May 2005 in China. In: Geosciences and Remote Sensing Symposium, pp. 828–621 (2006)
12. Huang, J., Minnis, P., Yi, Y., Tang, Q., Wang, X., Hu, Y., Liu, Z., Ayers, K., Trepte, C., Winker, D.: Summer dust aerosols detected from CALIPSO over the Tibetan Plateau. *J. Geophysical Research Letters* 34, 5 (2007)
13. Kaiping, W., Zhang, T., Bin, H.: Detection of Sand and Dust Storms from MERIS Image Using FE-Otsu Algorithm. In: The 2nd International Conference on Bioinformatics and Biomedical Engineering, pp. 3852–3855 (2008)
14. Ogunjobi, K.O., Kim, Y.J., He, Z.: Aerosol optical properties during Asian dust storm episodes in South Korea. In: Theoretical and Applied Climatology, vol. 76(1-2), pp. 65–75. Springer Wien, Heidelberg (2004)
15. El-Askary, H., Abhishek, A., El-Ghazawi, T., Menas, K., Le-Moigne, J.: Enhancing Dust Storm Detection Using PCA based Data Fusion. In: Geoscience and Remote Sensing Symposium, pp. 1424–1427 (2005)

Extraction of Buildings Footprint from LiDAR Altimetry Data with the Hermite Transform

José Luis Silván-Cárdenas¹ and Le Wang^{2,*}

¹ Centro de Investigación en Geografía y Geomática “Ing. Jorge L. Tamayo” A.C.
Contoy 137, Lomas de Padierna, Tlalpan, Mexico D.F. 14240

jlsilvan@centrogeo.org.mx

<http://www.centrogeo.org.mx>

² Department of Geography, The State University of New York
105 Wilkeson Quad, Buffalo, NY 14261

lewang@buffalo.edu

<http://www.buffalo.edu>

Abstract. Building footprint geometry is a basic layer of information required by government institutions for a number of land management operations and research. LiDAR (light detection and ranging) is a laser-based altimetry measurement instrument that is flown over relatively wide land areas in order to produce digital surface models. Although high spatial resolution LiDAR measurements (of around 1 m horizontally) are suitable to detect aboveground features through elevation discrimination, the automatic extraction of buildings in many cases, such as in residential areas with complex terrain forms, has proved a difficult task. In this study, we developed a method for detecting building footprint from LiDAR altimetry data and tested its performance over four sites located in Austin, TX. Compared to another standard method, the proposed method had comparable accuracy and better efficiency.

Keywords: Building footprint, Hermite Transform, Local Orientation.

1 Introduction

Remotely sensed data has become a primary source of information for a number of land management and research activities. The increased spatial resolution of remote sensing data has made it possible to produce detailed inventories of above ground features, such as buildings. Unfortunately, the production of such inventories still relies on much on-screen visual interpretation by human experts. The automatization of such processes is of great value for large-scale applications.

Airborne light detection and ranging (LiDAR) is a technology used routinely for producing high-spatial resolution digital terrain models. LiDAR systems deliver irregularly spaced 3-D points of ground and nonground surfaces. LiDAR measurements have several advantages over traditional aerial photographs and

* This study was partly supported by NSF grants (BCS-0822489 and SEB-0810933) and by CentroGeo.

satellite images because they are not influenced by sun shadow and relief displacement [10]. However, datasets tend to be voluminous and not suitable for automated extraction of building information, mainly because many raster image processing techniques cannot be directly applied to irregularly spaced points. To circumvent such limitation, elevation values are usually rasterized. Once LIDAR measurements are in raster format, the problem consists in discriminating building cells. Such a processing serves as a precursor to form building footprint polygons that can be incorporated in a geographic information system.

Two approaches are often utilized to detect building cells from gridded elevation measurements. One is to apply a classification method to separate the ground, buildings, trees, and other features simultaneously [2]. The more popular way is to separate the ground from nonground LIDAR measurements first and then identify the building points from nonground measurements [8,10]. The proposed method followed the latter approach, building upon a prior work on ground filtering [5]. The rest of the paper presents the theoretical background (Section 2), the method description (Section 3), some results from building detection tests (Section 4) and conclusions (Section 5).

2 Background

This section summarizes basic theoretical results that are relevant for the proposed building detection method, as well as a standard method used for comparison. The reader is referred to the original sources for further details.

2.1 Discrete Hermite Transform

The DHT of a two-dimensional signal $z : \mathcal{G} \mapsto \mathbb{R}$ defined on a grid $\mathcal{G} \subseteq \mathbb{Z}^2$, is comprised by filtered and downsampled versions of the original signal, i.e.,

$$z_{n,m}(p,q) = \sum_{(x,y) \in \mathcal{G}} z(x,y) b_n(x-2p) b_n(y-2q) \quad (1)$$

for $n, m = 0, \dots, N$, where the analysis functions

$$b_n(x) = 2^{-N} \sqrt{C_N^n} \Delta^n \left\{ C_{N-n}^{x+N/2} \right\} \quad (2)$$

for $x = -N/2, \dots, N/2$, correspond to discrete approximations of Gaussian derivatives. The discrete counterpart of the derivative operator corresponds to the forward difference denoted by Δ .

The signal is reconstructed from the DHT representation above as follows:

$$z(x,y) = \sum_{n,m=0}^N \sum_{(p,q) \in \mathcal{G}} z_{n,m}(p,q) \tilde{b}_n(2p-x) \tilde{b}_n(2q-y) \quad (3)$$

where the synthesis function are given by $\tilde{b}_n(x) = 2b_n(-x)$.

The coefficient $z_{n,m}(p, q)$ approximates (up to a normalization factor) the partial derivative of order n with respect to x and order m with respect to y of a Gaussian-smoothed version of the signal z at the location $(p, q) \in \mathcal{G}$. The degree of smoothness is controlled through the standard deviation of the Gaussian function, given by $\sigma = \sqrt{N}/2$. In practice, the parameter N takes the values of 2, 4, 6 or 8, whereas a larger degree of smoothing can be achieved through a multiscale DHT, as described in [4].

Rotated DHT. The rotated DHT [34] is defined in terms of derivatives with respect to a coordinate system (u, v) , that has been rotated by an angle θ with respect to the original coordinate system (x, y) . The rotated coefficients at the sampling location (p, q) , here denoted by $z_{n,m}^{(\theta)}(p, q)$, are expressed as a linear combinations of the original (non-rotated) coefficients $z_{n,m}$. More specifically,

$$\frac{z_{n-m,m}^{(\theta)}(p, q)}{\sqrt{C_n^m}} = \sum_{k=0}^n A_{m,k} \frac{z_{k,n-k}(p, q)}{\sqrt{C_n^k}} \quad (4)$$

where the coefficients $A_{m,k}$ correspond to the generalized binomial filters (GBF), a family of discrete sequences with parameters n and θ , which are given by

$$A_{m,k} = s^k c^{-k} \Delta^m \{C_{n-m}^{k-m} c^{2k-m} s^{n-2k+m}\} \quad (5)$$

for $m, k = 0, \dots, n$, and $c = \cos(\theta)$ and $s = \sin(\theta)$. The first few GBF can be expressed using the matrix notation $\mathbf{A}_n = [A_{m,k}]_{m,k=0,\dots,n}$ as

$$\begin{pmatrix} s & c \\ c & -s \end{pmatrix}, \begin{pmatrix} s^2 & 2sc & c^2 \\ sc & c^2 - s^2 & -sc \\ c^2 & -2sc & s^2 \end{pmatrix}, \begin{pmatrix} s^3 & 3s^2c & 3sc^2 & c^3 \\ s^2c & -s^3 + 2sc^2 & -2s^2c + c^3 & -sc^2 \\ sc^2 & -2s^2c + c^3 & s^3 - 2sc^2 & s^2c \\ c^3 & -3sc^2 & 3s^2c & -s^3 \end{pmatrix}$$

$$\begin{pmatrix} s^4 & 4s^3c & 6s^2c^2 & 4sc^3 & c^4 \\ s^3c & -s^4 + 3s^2c^2 & -3s^3c + 3sc^3 & -3s^2c^2 + c^4 & -sc^3 \\ s^2c^2 & -2s^3c + 2sc^3 & s^4 - 4s^2c^2 + c^4 & 2s^3c - 2sc^3 & s^2c^2 \\ sc^3 & -3s^2c^2 + c^4 & 3s^3c - 3sc^3 & -s^4 + 3s^2c^2 & -s^3c \\ c^4 & -4sc^3 & 6s^2c^4 & -4s^3c & s^4 \end{pmatrix}$$

for $n = 1, 2, 3$ and 4 respectively.

In all the examples presented here, the rotation was set to $\theta = \arctan(z_{0,1}/z_{1,0})$, which makes the rotated coefficients $z_{0,1}^{(\theta)} = 0$ and $z_{1,0}^{(\theta)} = g$, where $g = \sqrt{z_{1,0}^2 + z_{0,1}^2}$ is proportional to the gradient magnitude.

2.2 Region Growing Segmentation

The region-growing segmentation (RGS) method, as applied for building detection in [10], was used as a benchmark for the proposed building detection

method. This RGS is an iterative method that applies a plane-fitting technique to grow regions from seeds.

The RGS algorithm requires a non-ground mask of grided elevation values. Then, for each nonground measurement area, inside and boundary cells are identified. If at least one of the eight neighbors of a cell is a ground measurement, the cell is defined as a boundary cell. Otherwise, the cell is an inside cell. The following residual is calculated for each inside cell $p_0(x_0, y_0, z_0)$ and its eight neighbors:

$$R = \sum_{k \in M} [a(x_k - x_0) + b(y_k - y_0) + c - z_k]^2 \quad (6)$$

where M is a set for the inside cell and its neighbors, and a, b and c are plane parameters estimated through least square. The cell with the minimum residual R is labelled and selected as the first seed cell for region growing. All neighbors of a seed cell are labelled as belonging to the same segment if the deviation between its height and the plane height is under a threshold. A threshold of 0.1 m was used in our implementation. The plane parameters are then updated including the new labelled cells. The neighbors of the grown area are examined further, and the process is continued until no additional cells can be added into the segment. Then, the unlabelled cell with the minimum R is selected as the next seed. The process is repeated until all nonground cells are labelled.

After the RGS algorithm was run (and following [10]), small segments (with less than five pixels) were removed, holes were filled and contiguous segments were merged to form building footprints.

3 Building Detection

This section provides the details of the proposed building detection method, which can be summarized in two steps. In the first step, a height map for above-ground features is produced. This step requires subtracting the terrain component $t(x, y)$ from the elevation surface $z(x, y)$. The second step consists on separating building cells from vegetation cell on the height map. A description of each step is presented bellow.

3.1 Aboveground Features Height

The method used here for generating the terrain component is based on the so-called ground filtering method introduced in [5], which in turn was based on a multiscale implementation of the DHT [4]. Let $\{z_{n,m}^{(k)}\}$ denote the multiscale DHT coefficients of the gridded DSM $z(x, y)$ at a generic lattice point, and let $g^{(k)}$ denote the gradient magnitude at the scale index k , for $k = 0, \dots, K - 1$, where K is the number of pyramid layers of the multiscale DHT. It can be shown that the coefficients

$$\hat{t}_{n,m}^{(k)} = \begin{cases} z_{n,m}^{(k)} & \text{if } g^{(k)} \leq T^{(k)} \\ z_{0,0}^{(k)} - g^{(k)} / \sqrt{2} & \text{if } g^{(k)} > T^{(k)} \text{ and } n, m = 0 \\ 0 & \text{if } g^{(k)} > T^{(k)} \text{ and } n, m > 0 \end{cases} \quad (7)$$

reconstruct a signal $\hat{t}(x, y)$ that approximates the terrain elevation surface, provided that the multiscale gradient thresholds are selected according to

$$T^{(k)} = \frac{2^k m_{max}}{\sqrt{2 + 2\pi(2^k m_{max}/\Delta_{max})^2}} \quad (8)$$

where m_{max} and Δ_{max} are the maximum terrain slope and maximum terrain elevation difference in the site.

In all the tests presented bellow, the number of pyramid layers K of the multiscale DHT decomposition was determined as

$$K = \lceil \log_2(L_{max}/2\delta) \rceil \quad (9)$$

where L_{max} denote the maximum length of aboveground features and δ is the cell size of the gridded elevation values. This number of layers assured that large aboveground features were effectively removed.

In the ground mask, a cell (x, y) is assumed nonground if $z(x, y) - \hat{t}(x, y) > \epsilon$, where ϵ was set to 0.1 in all tests performed here. Once the ground mask is build, elevation values of detected nonground cells can be interpolated from elevation values of surrounding ground cells to produce a more accurate terrain component $t(x, y)$. Finally, the feature height map is computed as:

$$h(x, y) = z(x, y) - t(x, y) \quad (10)$$

3.2 The Planar Roof Model

The rationale here is that most roofs are composed of strongly oriented, mainly planar, surfaces, whereas forested areas are not. Hence, the rotated DHT coefficients along the local gradient is essentially distinct for building roofs and trees.

Let $h_{n,m}^{(\theta)}$ denote the rotated DHT coefficients along the local gradient of the feature height map. Then, building cells can be separated from vegetation cells by thresholding the residual energy term

$$E = \sum_{i=2}^N \sum_{j=1}^i \{h_{i-j,j}^{(\theta)}\}^2 \quad (11)$$

This residual energy measures the degree to which the local pattern does not conform to a one-dimensional signal (such as a planar surface). As it turns out, E is insensitive to planar roofs because the scale-space derivatives are only sensitive to polynomial variations of the derivation order or above.

In all the tests performed in this study an empirical threshold of 0.15 was used. Building masks so-produced were filtered in a similar fashion as the region-growing segmentation masks, so that small segments were eliminated and holes filled.

4 Results

The datasets used here consisted of grided elevation values from the last return of a laser pulse (Fig. 1, top row) and the actual building footprint from visual interpretation of aerial photography (Fig. 1, 2nd row). The study selected four representative sites of Austin City in Texas. Further descriptions of acquisition and preprocessing of the datasets are provided in [6]. The building detection results for each inset are illustrated in Fig. 1 (3rd and 4rd rows). These error maps were built through comparing the detection mask from each method with the actual building footprint layer in raster format. Errors of omission and commission are colored with blue and red for easy identification.

The overall per-pixel accuracy, the kappa statistics, the detection rate, and the commission error [7] were calculated for each method and inset. These results are provided in Table 1. As observed in these table, there was some accuracy variability across insets. Specifically, inset 1 was the most accurately classified into building and non-building by both methods with a per-pixel accuracy around 95%. This was due to the relatively high and large structure of multifamily buildings. On the other hand, insets 2 and 4 represented the most challenging area due to the relatively small size of single-family buildings and the high chance of occlusions by trees. In this case, the largest accuracy was under 90%. On the other hand, no significant differences in accuracies from the DHT and the RGS methods existed. In any case, differences did not always favored one method consistently, so that in the average both methods performed comparably. However, the most significant advantage of DHT over RGS was the saving in computation time, which was in the order of several hundred of times (data not shown). This is because the plane-fitting technique employed by RGS requires multiple matrix inversions, whereas the DHT method mainly involves convolutions operations which are computed efficiently [4].

Table 1. Two level accuracy assessment of building detection. Statistics were based on four insets, each of which had 145, 476, 427 and 490 buildings, respectively. Values were rounded to the nearest integer.

Method	Dataset	% Pixels		% Objects	
		Overall	Acc. Kappa	Detection Rate	Commission
DHT	Inset 1	95.0	80.9	77.9	18.1
	Inset 2	88.5	51.5	80.7	22.4
	Inset 3	90.8	56.3	90.9	26.3
	Inset 4	88.9	61.0	89.8	18.3
	Average	90.8	62.4	84.8	21.3
RGS	Inset 1	94.9	81.3	81.8	29.8
	Inset 2	89.0	51.6	73.0	12.4
	Inset 3	92.0	62.4	90.1	19.6
	Inset 4	88.8	60.5	88.4	14.6
	Average	91.2	63.9	83.3	19.1

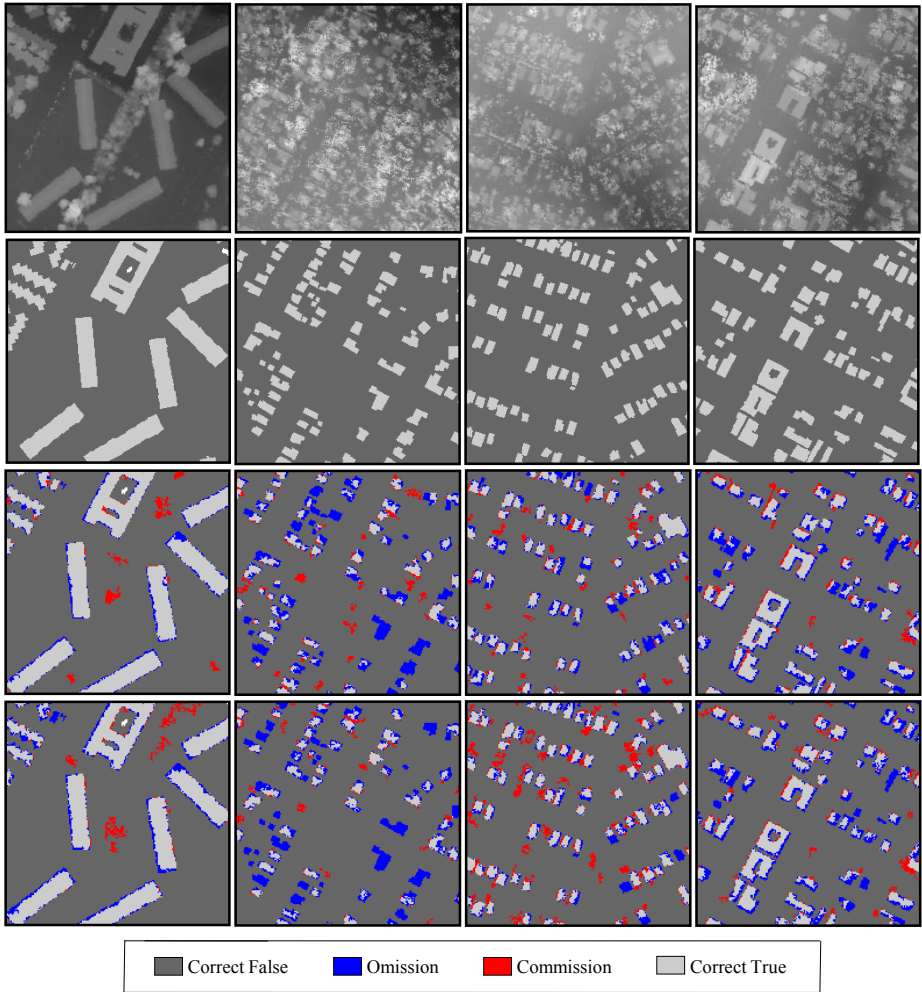


Fig. 1. Building detection results. Rows from top to bottom correspond to original gridded LiDAR data, actual building footprint, error map from RGS, and error maps from DHT, whereas columns from left to right correspond to subsets from inset 1 through inset 4, respectively.

5 Conclusions

This study proposed and tested the DHT as a tool for building extraction from gridded LiDAR data. The proposed building detection method used a multi-resolution ground filtering method based on the multiscale DHT, which is

efficiently computed [5]. The detection of buildings consisted on a simple thresholding of an energy term of the rotated DHT. Results indicated that the DHT building detection method competes with a more traditional method based on plane-fitting region growing segmentation. The appealing advantage of the proposed approach seemed to be its computational efficiency, which is crucial for large scale applications. For instance, this technique can be used for small-area population estimation as in [6]. Further research should explore the optimality of parameter selection for both the ground filtering and the energy thresholding. Also, building occlusions by trees represent a big challenge, which demands alternate approaches. One of such alternatives would be the partial active basis model presented in [9,11], where Gabor wavelet elements could be replaced by Gaussian derivatives.

References

1. Herrera-Domínguez, P., Altamirano-Robles, L.: A Hierarchical Recursive Partial Active Basis Model. *Advances in Pattern Recognition*, 1–10 (2010)
2. Miliaresis, G., Kokkas, N.: Segmentation and object-based classification for the extraction of the building class from LIDAR DEMs. *Computers & Geosciences* 33(8), 1076–1087 (2007)
3. Silván-Cárdenas, J.L., Escalante-Ramírez, B.: Image coding with a directional-oriented discrete hermite transform on a hexagonal sampling lattice. In: Tescher, A. (ed.) *Applications of Digital Image Processing XXIV*, vol. 4472, pp. 528–536. SPIE, San Diego (2001)
4. Silván-Cárdenas, J.L., Escalante-Ramírez, B.: The multiscale Hermite transform for local orientation analysis. *IEEE Transactions on Image Processing* 15(5), 1236–1253 (2006)
5. Silván-Cárdenas, J.L., Wang, L.: A multi-resolution approach for filtering LiDAR altimetry data. *ISPRS Journal of Photogrammetry and Remote Sensing* 61(1), 11–22 (2006)
6. Silván-Cárdenas, J., Wang, L., Rogerson, P., Wu, C., Feng, T., Kamphaus, B.: Assessing fine-spatial-resolution remote sensing for small-area population estimation. *International Journal of Remote Sensing* 31(21), 5605–5634 (2010)
7. Song, W., Haithcoat, T.: Development of comprehensive accuracy assessment indexes for building footprint extraction. *IEEE Transactions on Geoscience and Remote Sensing* 43(2), 402–404 (2005)
8. Weidner, U., Förstner, W.: Towards automatic building extraction from high-resolution digital elevation models. *ISPRS Journal of Photogrammetry and Remote Sensing* 50(4), 38–49 (1995)
9. Wu, Y., Si, Z., Gong, H., Zhu, S.: Learning active basis model for object detection and recognition. *International Journal of Computer Vision*, 1–38 (2009)
10. Zhang, K., Yan, J., Chen, S.: Automatic construction of building footprints from airborne LIDAR data. *IEEE Transactions on Geoscience and Remote Sensing* 44(9), 2523–2533 (2006)

Automatic Acquisition of Synonyms of Verbs from an Explanatory Dictionary Using Hyponym and Hyperonym Relations

Noé Alejandro Castro-Sánchez and Grigori Sidorov

Natural Language and Text Processing Laboratory,
Center for Research in Computer Science (CIC),
Instituto Politécnico Nacional (IPN),
Av. Juan Dios Batiz, s/n, Zacatenco, 07738,
Mexico City, Mexico
noe.acastro@gmail.com, sidorov@cic.ipn.mx

Abstract. In this paper we present an automatic method for extraction of synonyms of verbs from an explanatory dictionary based only on hyponym/hyperonym relations existing between the verbs defined and the genus used in their definitions. The set of pairs *verb-genus* can be considered as a directed graph, so we applied an algorithm to identify cycles in these kind of structures. We found that some cycles represent chains of synonyms. We obtain high precision and low recall.

Keywords: automatic acquisition of synonyms, hyponym and hyperonym relations, directed graph, cycles in explanatory dictionaries.

1 Introduction

Dictionaries are very important linguistic resources that contain the language vocabulary and allow its automatic processing.

There are various kinds of dictionaries and various ways to classify them. In this research we focus on dictionaries aimed at natives of a language (monolingual), without domain restrictions with the registered vocabulary (general) and that present the semantic definition of the lexical entries (explanatory).

Dictionaries present textual sections known as Lexicographic Article (LgA) that consists of an entry named *Lexical Unit* (LU) and the information that defines it or describes it. The information contains the elements that show the constraints and conditions for the use of the LU, and the semantic information (or definition) which represents the basic content of the LgA.

Very well known norms are followed for constructing definitions for the content words (what we primarily are interested in), which are named as *Aristotelic Definition*. It consists in a sentence headed by a generic term or hyperonym (*genus*) followed by characteristics that distinguish the LU from other items grouped within the same genus (*differentia*).

In this work we focus in this kind of lexical relations given between the LU (hyponym) and the genus (hyperonym) used in its definition. We considered all the pairs LU-genus as a directed graph, and then we applied an algorithm to find all the elementary cycles. We found that some of these cycles are made up for verbs that are synonyms.

This approach is similar to other recent works which consider dictionaries as graphs, linking headwords with words appearing in their definitions. In [2] a graph is constructed from a dictionary based on the assumption that synonyms use similar words in their definitions. The vertexes of the graph are words of the dictionary and an edge from vertex *a* to vertex *b* shows that word *b* appears in the definition of *a*. In [7] the graph structure of a dictionary is considered as a Markov chain whose states are the graph nodes and whose transitions are its edges, valued with probabilities. Then the distance between words is used to isolate candidate synonyms for a given word. The work [5] uses multiple resources to extract synonymous English words, like a monolingual dictionary, a parallel bilingual corpus (English-Chinese) and a monolingual corpus. Each resource was processed with a different method to extract synonyms and then an ensemble method was developed to combine the individual extractors. In [11] it is argued that definitions in dictionaries provide a regular syntax and style information (definitions) which provide a better environment to extract synonyms. It is proposed three different methods, two rule-based ones using the original definitions texts and one using the maximum entropy based on POS-tagged definitions.

The paper is organized as follows. In section 2, we explain how we process the dictionary and how we process the genus in the different ways they are used. In section 3, the method of creation of the graph is presented. In section 4, we show the results of our method, explain how we got the synonyms from a dictionary of synonyms for comparison and discuss the results. Finally in section 5, we conclude our studies and propose directions of the future work.

2 Processing of Dictionary

For our experiments the dictionary of Spanish Royal Academy (DRAE, as is known in Spanish) is used. It contains 162,362 definitions (senses) grouped in 89,799 lexical entries. From these, 12,008 lexical entries correspond to verbs, which contain 27,668 definitions (senses).

In this work, we are processing only verbs. We extract them from the dictionary, and then tagged them with the FreeLing parser, an open source text analysis tool for various languages including Spanish [1].

The next step was to identify and separate the grammatical marks, notes on usage, and other elements in the LgA.

2.1 Extraction of Genus from Definitions

Almost all definitions included in the dictionary follow the typical formula represented by *genus + differentia* (see Section 1). The predictable position of these elements allowed us to identify them in an automatic way.

Genus can be found in different ways, as it is shown below (in some cases the language differences between English and Spanish do not allow showing the characteristics in question):

1. As an only verb:
Cotizar. *Pagar una cuota.*
(Pay: *Pay a cuote.*)
2. As a chain of verbs linked by conjunctions or disjunctions:
Armonizar. *Escoger y escribir los acordes correspondientes a una melodía.*
(Harmonize. *Choose and write chords for a melody.*)
Aballar. *Amortiguar, desvanecer o esfumar las líneas y colores de una pintura.*
(Disappear. *Disappear or vanish the lines or colors of a paint*)
3. As a subordinate clause in infinitive carrying out the function of direct complement.
Gallear. *Pretender sobresalir entre otros con presunción o jactancia.*
(Brag. *Pretend to excel boastfully.*)
4. As a verbal periphrasis:
Pervivir. *Seguir viviendo a pesar del tiempo o de las dificultades.*
(Survive. *To remain alive despite the time or difficulties.*)
5. As a combination of the previous points.
Restaurar. *Reparar, renovar o volver a poner algo en el estado que antes tenía.*
(Restore. *To repair, renovate or bring back something to a previous state.*)

The items are shown in ascending order of complexity of processing. The items 1 and 2 are trivial. In 1 we identify the only verb and consider it as genus. In 2 we select all verbs that are heads of the clause as different genus. In items 3 and 4, we consider that the clause had only one genus made up of two verbs. Finally, in 5 we apply the previous considerations to identify the genus.

3 Construction of the Graph

We know that the relation between a LU and its genus is a hyponym-hyperonym relation. So, if we list all the pairs between LU-genus we obtain a directed graph, as is shown in the figure 1.

Each square represents a different verb and each number in circles is a different sense of a verb. So: *S* is a verb with senses 1 and 2. *G1* is the genus (verb) of the sense 1's definition of verb *S*; *G2* is the genus for definition in sense 2 of *S*, and so on.

But, if each verb has different number of senses, we start from a specific sense of the hyponym verb, but we do not know to which sense of the hyperonym we should establish the relation. As there is no explicit information for solving this problem, we assume that the relation can probably be to the first sense of the hyperonym, because dictionaries present the most common used sense in the first sense (see Section 4 for another possibility).

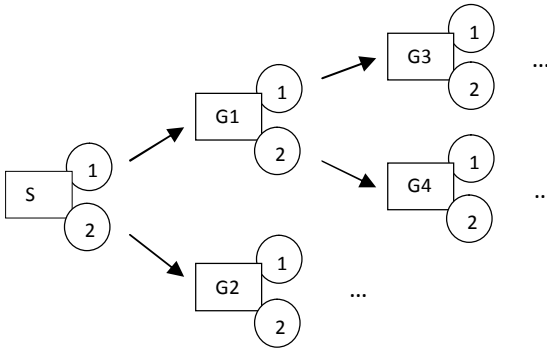


Fig. 1. Graph constructed from hyperonym relations

Now we can formalize these relations as:

$$V_{i=1}^n \rightarrow G_{j=1}(i)$$

Where:

V: Any verb.

i = Number of sense in V that is processed.

n = Total number of senses in V.

G = Genus of sense i in V.

j = First sense of Genus.

All this means that each sense of V, from $i = 1$ to n , is mapped to the first sense of Genus of the processed sense of the verb.

3.1 Extraction of Cycles

Obviously, any dictionary that defines all words it mentions must contain cycles (paths in which the first and the last vertices are identical); thus, cycles are an inevitable feature of a human-oriented dictionary that tries to define all words existing in the given language [4]. But it is assumed that a graph created from hyponym-hyperonym relations cannot contain cycles. However while processing some of the verbs, it is possible to find quite the opposite. For example:

1. **Pasar.** (1) *Llevar, conducir* de un lugar a otro.
(**Pass.** (1) To take, to convey from one place to other).
2. **Llevar.** (1) *Conducir* algo desde un lugar a otro...
(**Take.** (1) Convey something from one place to other...).
3. **Conducir.** (1) *Llevar, transportar* de una parte a otra.
(**Convey.** (1) Take, transport from one place to other).
4. **Transportar.** (1) *Llevar* a alguien o algo de un lugar a otro.
(**Transport.** (1). Take someone or something from one place to other).

Creating the graph, we obtained:

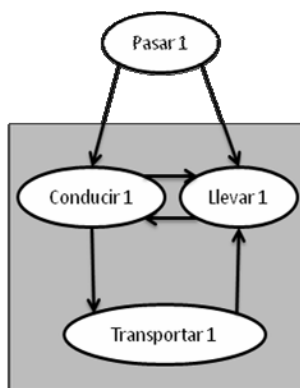


Fig. 2. Graph showing cycles among verbs linked from the genus of their definitions

So, connection between *Conducir* and *Llevar* allows start of the path in some of them and finish in the same starting vertex. There is a longer cycle (understanding *length* as the number of vertices covered to reach the starting vertex), which include the vertexes *Conducir*, *Llevar* and *Transportar*.

If definitions of those verbs are analyzed, the cycle suggests a different semantic relation than hyponym/hyperonym, which is the relation of being a *synonym*. So, what we think is that some (aristotelic) definitions, at least in this dictionary, do not use a genus or hyperonym, but a synonym.

For identification of the cycles, for each verb in the dictionary: it was identified the genus in the first sense, and we created a path to the first sense of the genus. After repeating this process we identified some cycles that correspond to synonymy.

4 Evaluation

The process of obtaining synonyms from the hyponym/hyperonym relations produced the identification of 225 verbs grouped in 84 cycles. This means that exist 84 groups of synonyms. To measure precision and recall we used the Spanish Espasa's Dictionary of Synonyms and Antonyms (2005), which contains more than 200,000 synonyms and antonyms separated for senses and grammatical categories.

The precision of our method was of 0.92. The errors are related with the following: 0.03 of verbs were not found in Espasa's dictionary and 0.05 of verbs that were reviewed by hand represent real synonyms. For example, definitions given by DRAE of verbs "*Sumir*" and "*Hundir*" are:

1. **Sumir.** *Hundir* o meter debajo de la tierra o del agua.
(**Plunge.** To sink or put under the ground or water).
2. **Hundir.** *Sumir*, meter en lo hondo.
(**Sink.** Plunge, put at depth).

In Espasa's dictionary, the only verbs having *sumir* as synonym are *abismar* and *sepultar*, although DRAE's definitions of both verbs show them as synonyms.

On the other hand, most of the cycles are made up for only two verbs, which gives a recall of 0.17 that is rather low. It is necessary to say that Espasa's dictionary does not provide an exhaustive review of the synonyms that represent each sense, i. e. one sense includes various synonyms that in a explanatory dictionary are separated in different senses. For example, for the verb *Poner*, DRAE contains:

1. **Poner.** Colocar en un sitio o lugar a alguien o algo.
(**Put.** To place in a specified position someone or something).

In the synonyms dictionary we found as synonyms of *Poner* verbs like *enchufar* (*plug in*), *adaptar* (*adapt*), *instalar* (*install*), and so on. All of these verbs are related with *Poner* but in a sense that is not the main. We do not know yet how the percentage of this kind of situations affects the recall.

4.1 Selection of the Correct Synonyms

Espasa's Dictionary groups synonyms by senses, so the question is how we can know that we are comparing our group of synonyms with the right synonyms took from the Espasa's Dictionary.

Let us consider the following: the Dictionary was converted into a Database where the synonyms are grouped into two fields: *Headword* (Hw) that is any word and *Synonyms* (Syn) that contains the synonyms of Hw. This relation is not commutative in the dictionary. This is to say, if the word *A* is in Hw and the word *B* is in Syn, it is not guaranteed that exists the interchanged relation (*B* in Hw and *A* in Syn). So, we do the following:

After naming each of our suggested synonyms as *candidates*, we apply the next steps to each candidate in the Espasa's Dictionary:

1. Extract synonyms for candidate *c* (candidate in Hw1).
2. Extract the verbs having the candidate *c* as synonym (candidate in Hw2).
3. Intersect results of step 1 with results of step 2.
4. The group of synonyms (sense) that has a higher number of verbs gotten in step 3 represents the synonyms which we consider to compare with.

4.2 Possible Improvements of the Algorithm

The previous method (see 3.1) only allows finding of a relatively little number of synonyms, and does not guarantee the extraction of all of them.

Here we explain an idea of a future method that works on the different data (all word senses of the genus, as compared to the current implementation of the method that uses only the first word sense of the genus) and in this way can increase the recall.

For example, the next sequence can't be discovered:

1. **Manifestar.** (2) *Descubrir*, poner a la vista.
(**Manifest.** To uncover, to bring to light).
2. **Descubrir.** (1) *Manifestar*, hacer patente.
(**Uncover.** To manifest, to make evident).

Manifestar in sense 2 and *Descubrir* in sense 1, it cannot be found with the previous algorithm (see section 3).

So, the solution is mapping the verb to all the senses of its genus. It can be formalized in the following expression:

$$V_{i=1}^n \rightarrow G_{j=1}^m(i)$$

Where:

V: Any verb.

i = Number of sense in V that is processed.

n = Total number of senses in V.

G = Genus of sense i in V.

j = Each sense of the Genus.

m = Total number of senses in G.

Then, verb V, from $i = 1$ to n , is mapped to all Genus' senses of i.

To get this task, we used the Johnson's algorithm [6], which report a faster processing than the well-known Tarjan's [8], [9] and Tiernan's algorithms [10].

We did some adaptations to the algorithm for our data processing: the inputs are files that are created from a specific verb. Each line of the file is made up from the mapping between verbs in a specific sense to their genus in all senses. For example, let's say that we want to create the file from the verb *Manifestar* in its sense 1, that is:

Manifestar. (1) *Declarar*, dar a conocer.

(**Manifest.** (1) *To declare*, to make known formally).

So, the content of the file is the next:

```
manifestar'1declarar'1
manifestar'1declarar'2
manifestar'1declarar'3
manifestar'1declarar'4
manifestar'1declarar'5
manifestar'1declarar'6
manifestar'1declarar'7
manifestar'1declarar'8
manifestar'1declarar'9...
```

For each file given as input, the algorithm creates another file containing the cycles.

The main problem with this approach is that some cycles generated by the algorithm do not contain correct synonyms. Let us see some lines of the output for the verb *manifestar*:

```
manifestar'2,poner'14,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'25,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'26,representar'3,manifestar'2,
manifestar'2,poner'17,hacer'41,representar'3,manifestar'2,
manifestar'2,poner'43,hacer'25,representar'3,manifestar'2, ...
```

Consulting the definitions of the verb/sense appearing in the first line of the list, we have:

1. **Manifestar.** (2) Descubrir, *poner* a la vista.
(**Manifest.** Uncover, bring to light).
2. **Poner.** (14) *Representar* una obra de teatro o proyectar una película en el cine o en la televisión.
(**Put.** (14) Perform a play or show a movie in the cinema or in the television)
3. **Representar.** (3) *Manifestar* el afecto del que una persona está poseída.
(**Represent.** (3) Manifest the affect that a person has).

It is clear that the senses of the three verbs do not represent the same semantic situation, and the verbs are not synonyms (still, they can be synonyms in other senses).

But even with this kind of troubles, it is possible to see that in the verbs constituting the cycles there are more synonyms than we obtained with the first algorithm. For example, for the verb *Manifestar*, all the verbs that make up the cycles are shown below:

Manifestar (manifest)
Declarar (declare)
 Hacer (make)
 Ejecutar (execute)
 Poner (put)
 Representar (represent)
comunicar (communicate)
 descubrir (discover)
exponer (expose)
presentar (present)
 disponer (arrange)
 mandar (order)
 tener (have)
 colocar (put)
contar (tell)
 arriesgar (risk)

The synonyms of the verb *Manifestar* are shown in boldface. With this method it is possible to get more synonyms and improve the recall. Still, we should verify that the precision will not reduce.

5 Conclusions and Future Work

In this work we propose a method for identifying the synonyms of verbs using an explanatory dictionary. The method is based on hyponym-hyperonym relations between the verbs (headwords) and the genus used in their definitions. This approach allowed us to identify that some aristotelic definitions of verbs do not use a genus or hyperonym, but a synonym. Otherwise we cannot explain why a sequence of verbs constructed from hyperonym relations finish in the starting verb.

The method presents two variants: the former is based on the fact that the first sense defining a headword is the most commonly used, so we think that cycles constructed among the first senses of verbs guarantees that the verbs are synonyms (we did not identify an opposite case at least in the dictionary we use). On the other hand, it has the problem of a low recall. We programmed and evaluated this variant.

We also propose an idea of the second variant thinking in identifying groups of synonyms that cannot be detected using the first method. Our idea is that it will improve the recall. The manual analysis of the cycles obtained using this variant shows promising results, still its exact evaluation is future work. The question is to identify those cycles that the algorithm produces and that are not correct. Some of them include verbs used as *Lexical Functions* (LF), defined as functions that associate a word with a corresponding word such that the latter expresses a given abstract meaning indicated by the name of lexical function. Some method could be used to identify LF (for example [3]) and discard cycles that contain them.

The proposed methods have various lexicographic applications, for example, improvement of definitions of some verbs comparing them with those used in their synonyms, searching a difference between a real hyperonym in a group of synonyms, etc.

Acknowledgements. Work done under partial support of Mexican Government (CONACYT projects 50206-H and 83270, SNI) and National Polytechnic Institute, Mexico (projects SIP 20080787, 20091587, 20090772, 20100773, 20100668; 20111146, 20113295, COFAA, PIFI), Mexico City Government (ICYT-DF project PICCO10-120), and European Commission (project 269180).

References

1. Atserias, J., Casas, B., Comelles, E., Gonz  les, M., Padr  , L., Padr  , M.: FreeLing 1.3: Syntactic and Semantic Services in an Open-Source NLP Library. In: Fifth international conference on Language Resources and Evaluation, Genoa, Italy (2006), <http://www.lsi.upc.edu/nlp/freeling>
2. Blondel, V., Senellart, P.: Automatic extraction of synonyms in a dictionary. In: Proceedings of the SIAM Text Mining Workshop, Arlington, VA (2002)
3. Gelbukh, A., Kolesnikova, O.: Supervised Learning for Semantic Classification of Spanish Collocations. *Advances in Pattern Recognition* 6256, 362–371 (2010)
4. Gelbukh, A., Sidorov, G.: Automatic selection of defining vocabulary in an explanatory dictionary. In: Gelbukh, A. (ed.) *CICLing 2002*. LNCS, vol. 2276, pp. 300–303. Springer, Heidelberg (2002)
5. Hang, W., Ming, Z.: Optimizing synonym extraction using monolingual and bilingual resources. In: *Proc. International Workshop on Paraphrasing* (2003)
6. Johnson, D.: Finding all the Elementary Circuits of a Directed Graph. *SIAM Journal on Computing* 4(1), 77–84 (1975)
7. Muller, P., Hathout, N., Gaume, B.: Synonym Extraction Using a Semantic Distance on a Dictionary. In: *Proceedings of TextGraphs: The Second Workshop on Graph Based Methods for Natural Language Processing*, pp. 65–72 (2006)

8. Tarjan, R.: Depth-first search and linear graph algorithms. *SIAM Journal on Computing*, 146–160 (1972)
9. Tarjan, R.: Enumeration of the elementary circuits of a directed graph. *SIAM Journal on Computing*, 211–216 (1973)
10. Tiernan, C.: An efficient algorithm for finding the simple cycles of a finite directed graph. *Comm. ACM* 13, 722–726 (1970)
11. Wang, T.: Extracting Synonyms from Dictionary Definitions. In: *Recent Advances in Natural Language Processing* (2009)

Using Finite State Models for the Integration of Hierarchical LMs into ASR Systems

Raquel Justo* and M. Inés Torres

University of the Basque Country
Sarriena s/n, 48940 Leioa, Spain

raquel.justo@ehu.es, manes.torres@ehu.es

Abstract. Through out this work we explore different methods to integrate a complex Language Model (a hierarchical Language Model based on classes of phrases) into an Automatic Speech Recognition (ASR) system. The integration is carried out by means of a composition of the different Stochastic Finite State Automata associated to the specific Language Model. This method is based on the same idea employed to integrate the different knowledge sources involved in the recognition process when a classical word-based Language Model is considered. The obtained results show that this integrated architecture provides better ASR system performance than a two-pass decoder where the complex LM is employed to reorder the N-best list.

Keywords: stochastic finite state models, speech recognition, hierarchical language models.

1 Introduction

Statistical decision theory is applied in a wide variety of problems within pattern recognition framework that aim at minimising the probability of erroneous classifications. The maximization of the posterior probability $P(\bar{w}|\bar{x})$ allows to get the most likely sequence of symbols $\bar{w}$, that matches a given sequence of input observations $\bar{x}$, as shown in eq. (1).

$$\hat{\bar{w}} = \arg \max_{\bar{w}} P(\bar{w}|\bar{x}) \quad (1)$$

Using the Bayes' decision rule eq. (1) can be rewritten as eq. (2). If we focus on the problem of Automatic Speech Recognition (ASR) the term $P(\bar{w})$ corresponds to the prior probability of a word sequence and it is commonly estimated by a Language Model (LM), whereas $P(\bar{x}|\bar{w})$ is estimated by an Acoustic Model (AM), typically a Hidden Markov Model (HMM).

$$\hat{\bar{w}} = \arg \max_{\bar{w}} P(\bar{w}|\bar{x}) = \arg \max_{\bar{w}} P(\bar{x}|\bar{w})P(\bar{w}) \quad (2)$$

* This work has been partially supported by the Government of the Basque Country under grant IT375-10, by the Spanish CICYT under grant TIN2008-06856-C05-01 and by the Spanish program Consolider-Ingenio 2010 under grant CSD2007-00018.

Nowadays Automatic Speech Recognition (ASR) systems use, mainly, Statistical Language Models (SLMs) in order to represent the way in which the combination of words is carried out in a specific language. Other approaches such as syntactic LMs, including a stochastic component, could also be employed in this kind of applications, i.e. stochastic context free grammars (SCFG) [5,2] or stochastic finite state models [10,11]. Although syntactic models can better model the structure of the language they still present problems regarding automatic inference and integration in ASR systems. In this work we use a syntactic approach, specifically k -Testable in the Strict Sense (k -TSS) LMs. k -TSS languages are a subclass of regular languages and, unlike SCFGs, they can be easily inferred from a set of positive samples by an inference algorithm [4]. Moreover, k -TSS LMs can be represented by Stochastic Finite State Automata (SFSA) allowing an efficient composition of them with other models, i.e. HMMs (in ASR applications).

AT&T laboratories presented an approach that simplifies the integration of different knowledge sources into the ASR system by using finite state models, specifically Stochastic Finite State Transducers (SFST) [10]. The underlying idea is to use a SFST to model each knowledge source, then SFSTs are compounded to obtain an only one SFST where the search of the best word sequence is carried out. Although optimization algorithms [8] can be applied the resulting SFST could still result too memory demanding. One way to solve this problem is the “on-the-fly” composition of SFSTs [3]. In the same way, since k -TSS LMs that can be represented by SFSA are considered in this work, the automaton associated to the LM is compounded with the HMMs representing AMs. Moreover, the idea of “on-the-fly” composition has also been used to obtain less memory demanding approaches.

One of the problems to be faced within the ASR framework is the selection of an appropriate LM. Among SLMs, word n -gram LMs are the most widely used approach, because of their effectiveness when it comes to minimizing the Word Error Rate. Large amounts of training data are required to get a robust estimation of the parameters defining aforementioned models. However there are numerous ASR applications for which the amount of training material available is rather limited. Different approaches can be found in the literature in order to solve this problem [9,12]. In this work, we employ hierarchical LMs based on classes of phrases [7] that has demonstrated to be efficient when dealing with data sparseness problems.

This kind of complex LMs, integrating different knowledge sources, entail an additional problem regarding the integration of them into the ASR system. One of the ways employed to solve this problem is to use a two-pass decoder, that is, first, a list of the N -best hypothesis is obtained from a classical decoder that considers a word-based LM. Then, the complex LM of choice is employed to reorder the list and to obtain the best word sequence. This decoupled architecture allows the recognition process to be carried out without any change in the decoder. However, it does not permit to take advantage of all the potential of the model because the recognition process is not guided by the LM of choice.

Alternatively, an integrated architecture which employs a one-pass decoder could be considered. This kind of integration is based on the use of SFSA associated to the LM. In this work in order to integrate hierarchical LMs into the ASR system we propose to use the same idea employed to integrate different knowledge sources in an ASR system. That is, the integration is carried out by doing an “on-the-fly” composition of the different SFSA associated to the different knowledge sources in the hierarchical LM.

2 A Hierarchical Language Model Based on Classes of Phrases

In this section we present the LMs employed in this work: a word-based LM M_w , a hierarchical LM based on classes of phrases M_{sw} and an interpolated LM, M_{hsw} , fully described and formulated in [7]. These models are defined within the Stochastic Finite State framework, specifically we use k -TSS LMs.

Thus, under the k -TSS formalism the probability of a sequence of N words ($\bar{w} = w_1, \dots, w_N = w_1^N$) is obtained considering the history of previous $k_w - 1$ words as shown in eq. (3), when considering a classical word based model (M_w).

$$P(\bar{w}) \simeq P_{M_w}(\bar{w}) = \prod_{i=1}^N P(w_i | w_{i-k_w+1}^{i-1}) \quad (3)$$

On the other hand, the probability of a word sequence ($\bar{w}$) using the M_{sw} model is given in the equation below.

$$P(\bar{w}) = \sum_{\forall \bar{c} \in \mathcal{C}^*} \sum_{\forall s \in \mathcal{S}(\bar{w})} P(\bar{w} | s, \bar{c}) P(s | \bar{c}) P(\bar{c}) \quad (4)$$

where $\mathcal{C}^*$ is a set of all the possible class sequences ($\bar{c}$) given a priori defined set of classes made up of phrases. s is a segmentation of a word sequence $w_1, \dots, w_N$ in M phrases and can be understood as a vector of M indexes. The set of all possible segmentations of a word sequence $\bar{w}$ is denoted by $\mathcal{S}(\bar{w})$.

The third term involved in eq (4) can be calculated as a product of conditional probabilities and it is approached by a class k -TSS model. The SFSA associated to the model can be inferred from a classified corpus and provides the probability for each class sequence as eq (5) shows.

$$P(\bar{c}) = \prod_{i=1}^T P(c_i | c_1^{i-1}) \simeq P(c_i | c_{i-k_c-1}^{i-1}) \quad (5)$$

where $k_c - 1$ stands for the maximum length of the considered class history.

To estimate the probability of the second term in eq (4) we assume that the segmentation probability is constant, that is, $P(s | \bar{c}) \simeq \alpha$.

Finally, $P(\bar{w}|s, \bar{c})$ is estimated considering that given a sequence of classes $\bar{c}$ and a segmentation s , the probability of a phrase given a class c_i depends exclusively on this c_i class and not on the previous ones

$$P(\bar{w}|s, \bar{c}) \simeq \prod_{i=1}^T P(w_{a_{i-1}+1}^{a_i} | c_i) \quad (6)$$

The term $P(w_{a_{i-1}+1}^{a_i} | c_i)$ represents the probability of a sequence of words, which is the phrase corresponding to the segmentation indexes $(a_{i-1} + 1, a_i)$, given the class of this phrase. To estimate this probability, a k -TSS model, represented by an SFSA, can be used for each class, as shown in eq (7).

$$P(w_{a_{i-1}+1}^{a_i} | c_i) \simeq \prod_{j=a_{i-1}+1}^{a_i} P(w_j | w_{j-k_{cw}+1}^{j-1}, c_i) \quad (7)$$

where $k_{cw} - 1$ stands for the maximum length of the word history that is considered in each class c_i .

Summing up, $N_c + 1$ (where N_c is the considered number of classes) SFSA are needed to represent the M_{sw} model: one for each class considering the relations among words inside the classes and an additional one that considers the relations among classes.

Finally, an interpolated model (M_{hsw}) is defined, here, as a linear combination of a word-based LM, M_w , and a hierarchical LM based on classes of phrases, M_{sw} . Using such a model the probability of a word sequence is given by eq. (8).

$$P_{M_{hsw}} = \lambda P_{M_w}(\bar{w}) + (1 - \lambda) P_{M_{sw}}(\bar{w}) \quad (8)$$

3 Integration of Complex LMs into an ASR System

The goal of an ASR system is to obtain the most likely word sequence given the acoustic signal uttered by the speaker. In this work, all the models involved in the decoding process (acoustic models AM, language model LM and lexical model) were integrated into the SFSA framework. Thus, the problem of finding the most likely word sequence would be solve by finding the most likely path in the search network obtained by doing the composition of all the automata representing the models. However, a static composition of all the automata can cause computation problems regarding memory allocation when large vocabularies are employed. Instead of carrying out such a composition where different parts of the network are replicated, the composition of different models could be done on demand at decoding time. Fig. 1 illustrates the search network built to carry out this kind of integration when a classical M_w model is employed.

A vocabulary of two words w_1 = “no” and w_2 = “nada” has been employed. In order to obtain the transition probabilities among different nodes s_i of the network, the SFSA associated to each model has to be consulted when required. Specifically, the transition probabilities among words (red arrows in Fig. 1) are calculated turning to the SFSA associated to the word k -TSS LM (M_w).

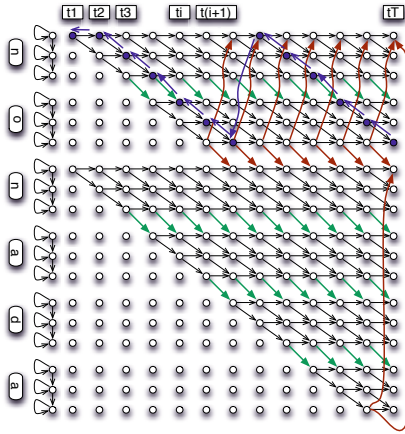


Fig. 1. Search network for a word 1-TSS model

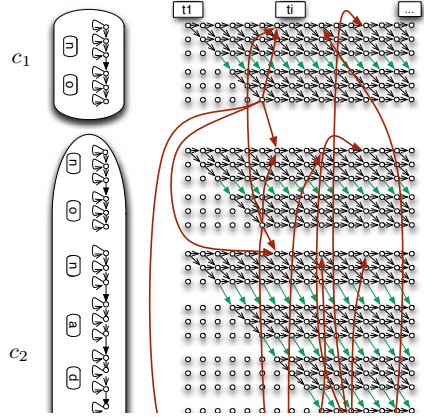


Fig. 2. Search network for M_{sw} , with $k_c = 1$ and $k_{cw} = 1$

However, in this work, we want to integrate in the ASR system a LM considering different knowledge sources, the M_{sw} model. In order to do this we used different architectures for comparison purposes. In the first one, decoupled architecture shown in Fig. 3, the recognition process is carried out using a two-pass decoder that considers a standard word-based LM (M_w). The output of the ASR system is a word-graph from which the N-best list is commonly extracted. However, the obtention of the word graph entails prohibitive computational costs or coarse approaches due to very restrictive assumptions [6]. Thus we do not obtain the real N-best list but an approach of it. Then, the M_{sw} model is employed to provide a new score to the obtained hypothesis and to reorder them in terms of this new score. Thus, we finally obtain a new best hypothesis which is considered the output of the system when the M_{sw} model is used. Although, this architecture tries to simulate the integration of the model into the ASR system

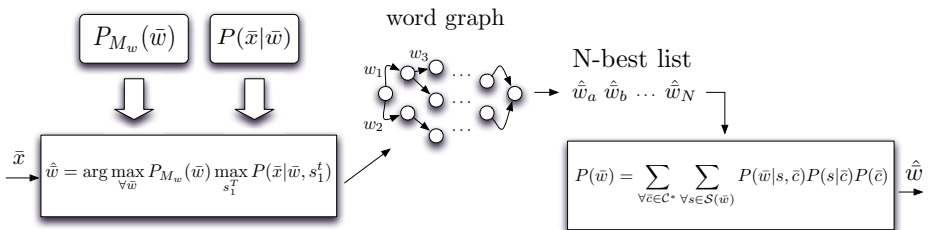


Fig. 3. Decoupled architecture for an ASR system considering a M_{sw} model

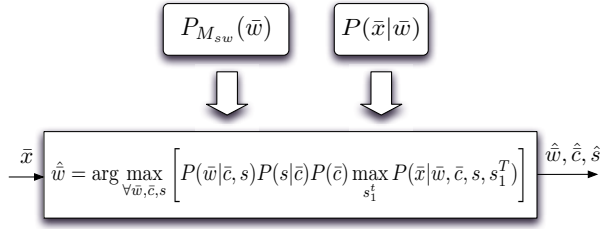


Fig. 4. Integrated architecture for an ASR that considers a M_{sw} model

the recognition process is not guided by the LM of choice, so the obtained result is limited by the best result a M_w model could provide using a word graph.

On the other hand and taking advantage of the use of stochastic finite state models, we propose in this work to integrate complex LMs into a one step decoder as shown in Fig. 4. In this architecture the decoder was modified to be able to integrate the M_{sw} model in the recognition process.

The M_{sw} model can be represented by different SFSA, a SFSA that captures the relations among the classes and N_c (where N_c is the size of the class vocabulary) additional SFSA to consider the relations among the words inside each class. Under the approach proposed in this work an “on-the-fly” composition of the automata could be done at decoding time in the same way the composition of the automata associated to lexical and word-based language model (M_w) is carried out in the standard decoder.

Let us show an example to illustrate this method. We assume that Fig. 5 and 6 represent the automata considering the relations among classes and the specific automaton associated to the class c_2 respectively. Fig. 2 shows the search network for this example when M_{sw} is considered.

When the probability of a transition is needed the corresponding automata have to be considered. Specifically the probabilities of the transitions among words (red arrows) are obtained in the way described below:

Let us focus on the word sequence “\$ no nada más gracias”. There are different paths in the search network associated to the different segmentations and classifications for this word sequence.

If we consider one of those paths, the class sequence “ $c_{\$}c_2c_2$ ” and the segmentation “no nada más - gracias”, the associated probability is obtained according to eq. (5), (6) and (7) as follows:

$$P(\$ \text{ no nada más gracias}) = P(c_{\$})P(\$|c_{\$})P(c_2|c_{\$})P(\text{no}|c_2) \cdot P(\text{nada}|\text{no}, c_2)P(\text{más}|\text{nada}, c_2)P(c_2|c_{\$}c_2)P(\text{gracias}|c_2) \quad (9)$$

where $P(\$|c_{\$}) = 1$ y $P(c_{\$}) = 1$. $P(c_2|c_{\$})$ is the probability of the transition labeled with c_2 in the SFSA of Fig. 5 (red transition). $P(\text{no}|c_2)$ is obtained from the transition labeled with the word “no” in the automaton associated to c_2 class of Fig. 6 (blue transition), $P(\text{nada}|\text{no}, c_2)$ is obtained from the green transition

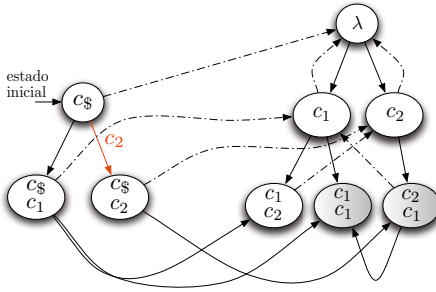


Fig. 5. Class k -TSS model with a value $k_c = 3$

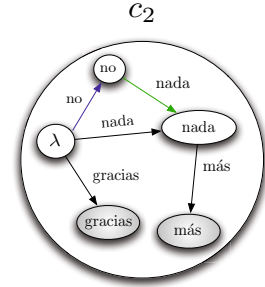


Fig. 6. k -TSS model associated to c_2 class with a value $k_{cw} = 2$

in Fig. 6 and so on. However, to obtain the probability $P(c_2|c_§c_2)$ it is necessary to consider again the automaton of Fig. 5. That is, the probabilities associated to the automaton of Fig. 5 have to be considered when a final state is reached in the specific automaton of a class and transitions among classes are needed.

Moreover, the use of stochastic finite state framework allows to integrate into a one step decoder the hybrid model M_{hsw} defined like the linear combination of a M_w and a M_{sw} model.

4 Experimental Results

The experiments described in this section were carried out over DIHANA corpus. This corpus consists of 900 human-machine dialogues in Spanish. 225 speakers ask by telephone for information about long-distance train timetables, fares, destinations and services. It comprises 5,590 different sentences to train the LM with a vocabulary of 865 words. The test set (a subset of the whole test set) includes 400 spoken utterances. This task has intrinsically a high level of difficulty due to the spontaneity of the speech and the problematic derived from the acquisition of large amount of training data. Thus, data sparsity is a problem that need to be faced.

Different LMs and methods of integration were evaluated in terms of *Word Error Rate* (WER). First of all we used a word k -TSS LM M_w (with a value $k_w = 3$). This model was integrated into the ASR system and evaluated in terms of WER. Then M_{sw} ($k_c = 2$ and $k_{cw} = 2$) and M_{hsw} models were considered and also integrated into the ASR system using the one-pass decoder. On the other hand, M_{sw} with the same features was considered but in this case the recognition process was carried out by using the decoupled architecture (two-pass decoder). The obtained results are given in Table 1.

As Table 1 shows the integration carried out by means of one-pass decoder provides better results than the integration using a two-step decoder. In fact, M_{sw} model significantly outperforms M_w (improvement of 8.7%), when using

Table 1. WER results for M_w , M_{sw} and M_{hsw} models using different architectures

	M_w	M_{sw} (one-pass)	M_{hsw} (one-pass)	M_{sw} (two-pass)
WER	16.81	15.18	14.23	15.74

the first one, whereas the two pass decoder provides an improvement of 6.4% when rescoring with this M_{sw} model. Moreover, in the two-pass decoder there are two LMs involved (M_w in the first step and M_{sw} in the second step), thus it should be compared with the results obtained with the one pass-decoder and the interpolation of both models (M_w and M_{sw}), that is M_{hsw} model, which provides an improvement of 14.2%. These differences in the system performance could occur due to the fact that the more complex LM (M_{sw}) guides the recognition process when the integrated architecture is considered, while the recognition is guided by the M_w model in the decoupled architecture. Thus, the obtained results proof that a way of integrating the LMs into the ASR system is needed in order to evaluate the system performance for different LMs.

5 Conclusions

In this work we explore different methods to integrate a hierarchical LM based on classes of phrases into an ASR system. The LM is defined within the Stochastic Finite State framework, thus it can be represented by means of different SFSA. The integration is carried out employing an “on-the-fly” composition of the different SFSA associated to the model. WER results are obtained for this integrated architecture (one-pass decoder) and for a decoupled one (two-pass decoder). The obtained results show that the integrated architecture provides significantly better results than those obtained with the decoupled architecture.

References

1. Benedí, J., Lleida, E., Varona, A., Castro, M., Galiano, I., Justo, R., López, I., Miguel, A.: Design and acquisition of a telephone spontaneous speech dialogue corpus in Spanish: DIHANA. In: Proceedings of LREC 2006, Genoa, Italy (May 2006)
2. Benedí, J.M., Sánchez, J.A.: Estimation of stochastic context-free grammars and their use as language models. *Computer Speech & Language* 19(3), 249–274 (2005)
3. Caseiro, D., Trancoso, I.: A specialized on-the-fly algorithm for lexicon and language model composition. *IEEE Transactions on Audio, Speech & Language Processing* 14(4), 1281–1291 (2006)
4. García, P., Vidal, E.: Inference of k-testable languages in the strict sense and application to syntactic pattern recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 12(9), 920–925 (1990)
5. Jurafsky, D., Wooters, C., Segal, J., Stolcke, A., Fosler, E., Tajchman, G., Morgan, N.: Using a stochastic context-free grammar as a language model for speech recognition. In: Proceedings of ICASSP 1995, pp. 189–192. IEEE Computer Society Press, Detroit (1995)

6. Justo, R., Pérez, A., Torres, M.I.: Impact of the approaches involved on word-graph derivation from the asr system. In: Proceedings of the IbPRIA 2011, Las Palmas de Gran Canaria, Spain, June 8-10 (2011) (to be published in LNCS)
7. Justo, R., Torres, M.I.: Phrase classes in two-level language models for asr. *Pattern Analysis & Applications* 12(4), 427–437 (2009)
8. Mohri, M., Riley, M.: A weight pushing algorithm for large vocabulary speech recognition. In: Proceedings of INTERSPEECH 2001, Aalborg, Denmark, September 2001, pp. 1603–1606 (2001)
9. Niesler, T., Whittaker, E., Woodland, P.: Comparison of part-of-speech and automatically derived category-based language models for speech recognition. In: ICASSP 1998, Seattle, pp. 177–180 (1998)
10. Pereira, F., Riley, M.D.: Speech recognition by composition of weighted finite automata. In: *Finite-State Language Processing*, pp. 431–453. MIT Press, Cambridge (1996)
11. Torres, M.I., Varona, A.: k-TSS language models in speech recognition systems. *Computer Speech and Language* 15(2), 127–149 (2001)
12. Zitouni, I.: Backoff hierarchical class n-gram language models: effectiveness to model unseen events in speech recognition. *Computer Speech and Language* 21(1), 99–104 (2007)

Use of Elliptic Curves in Term Discrimination*

Darnes Vilariño, David Pinto, Carlos Balderas,
Mireya Tovar, Beatriz Beltrán, and Sofia Paniagua

Faculty of Computer Science
Benemérita Universidad Autónoma de Puebla, Mexico
{darnes,dpinto,mtovar,bbeltran,sofia}@cs.buap.mx

Abstract. Detection of discriminant terms allow us to improve the performance of natural language processing systems. The goal is to be able to find the possible term contribution in a given corpus and, thereafter, to use the terms of high contribution for representing the corpus. In this paper we present various experiments that use elliptic curves with the purpose of discovering discriminant terms of a given textual corpus. Different experiments led us to use the mean and variance of the corpus terms for determining the parameters of a Weierstrass reduced equation (elliptic curve). We use the elliptic curves in order to graphically visualize the behavior of the corpus vocabulary. Thereafter, we use the elliptic curve parameters in order to cluster those terms that share characteristics. These clusters are then used as discriminant terms in order to represent the original document collection. Finally, we evaluated all these corpus representations in order to determine those terms that best discriminate each document.

1 Introduction

Term discrimination is a way to rank keywords of a given textual corpus [1]. The final aim of term discrimination is to support Natural Language Processing (NLP) tasks in order to improve the performance of their computational systems. Information retrieval, text classification, word sense disambiguation, summarization, are some examples of NLP tasks that may get benefit of a good term discrimination method [2].

We use the discriminant terms in order to represent the document with the hope of removing those terms that may introduce noise. Therefore, we may obtain a double benefit: on the one hand, we reduce the number of computational operations because of the corpus size reduction; on the other hand, we are expecting to increase the performance of the NLP system used in the task because we only consider to use the terms really involved in the characterization of the document [3].

* This work has been partially supported by the projects: CONACYT #106625, VIEP #VIAD-ING11-I, #PIAD-ING11-I, #BEMB-ING11-I, as well as by the PROMEP/103.5/09/4213 grant.

Up to now, different methods for automatic term discrimination have been proposed. Perhaps one of the most successful approach is the well-known tf-idf term weighting schema which was proposed by Salton in the 1970's [4]. This model proposes a simple manner for representing documents of a collection by means of weighted vectors. Each document is represented as a vector whose entries are weights of the vocabulary terms obtained from a text collection. The problem associated with this approach is that in huge collections of documents, the dimension of the vector space can be of tens of thousands, leading to a number of computational calculations that may be prohibitive in practice.

Some other approaches for term discrimination exist in literature. For instance, in [5] it is presented a statistical analysis of some set of words without knowledge of the grammatical structure of the documents using the concept of entropy. The theory of testors is another approach that may be used for term discrimination [6]. A testor is a set of features which may be used to represent a dataset. Although this theory may be adequate for selecting terms in a collection, it lacks of algorithms for efficient calculation of the testor set. In fact, in [7] it was presented the fastest algorithm, which is not polynomial in complexity.

Even if there exist various approaches for finding discriminant terms in document collections, we consider that the problem of determining those terms that better represent the documents (with a maximum tradeoff of precision and recall) still an open problem. Therefore, we are encouraged to explore new mechanisms in the term discrimination field.

In this paper, we present diverse experiments with the purpose of investigating the usefulness of elliptic curves, a topic highly investigated in the cryptography field, in the term discrimination and document representation task.

The remaining of this paper is organized as follows. In Section 2 we present a brief description of theoretical issues of elliptic curves. Thereafter, we propose different models for document representation by stating the parameters of a reduced Weierstrass equation that from now and forwards we will generally call as "elliptic curve". The evaluation of the different representations is given in Section 3. We use a corpus borrowed from the information retrieval field in order to perform those evaluations. Finally, in Section 4 the conclusions and findings are given.

2 Use of Elliptic Curves in Term Discrimination

An elliptic curve is an equation $y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_5$, where x and y are variables, and $a_1, \dots, a_5$ are constant elements of a field. Even if elliptic curves are important in mathematical areas such as number theory, they constitute a major area of current research and they find applications in some other areas such as cryptography [8].

The formal definition of an elliptic curve is fairly technical and requires some background in algebraic geometry. However, it is possible to describe some features of elliptic curves over the real numbers using only some concepts of algebra and geometry.

In this context, an elliptic curve is a smooth plane curve defined by an equation of the form:

$$y^2 = x^3 + ax + b, \quad (1)$$

where a and b are real numbers. The equation (1) is called a Weierstrass equation, and its discriminant must be different of zero in order to be non-singular; that is, its graph has no cusps or self-intersections. In Figure 1 we may see an example of an elliptic curve with parameters $a = 0.75$ and $b = 1.09$, that correspond to the mean and standard deviation of one term of the one of the eight corpus evaluated in this paper.

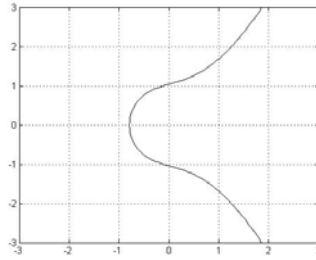


Fig. 1. An example of an elliptic curve with $a = 0.75$ and $b = 1.09$

An interesting feature of the elliptic curves is that these are parabolic curves centered on the x axis, when the parameters a and b are positive. Therefore, we may establish a distance measure between any pair of elliptic curves. In the context of NLP, we consider factible the use of elliptic curves for representing the documents. By having an appropriate set of parameters for elliptic curves would lead to have distance measures among the documents and, therefore, a similarity measure between any pair of documents. Thus, we consider important to investigate the adequate values for a and b in order to obtain an accurate representation of documents.

In this paper we propose three different approaches of values for the parameters of the elliptic curves, which we have named DR_1 , DR_2 and DR_3 . In the case of approaches DR_1 and DR_2 , we have defined the function $\text{ascii}(c_j)$, which is the ASCII code of the character c_j of term t ($t = \{c_1c_2c_3...c_{|t|}\}$):

DR_1 :

a is equal to $\sum_{j=1}^{|t|} \text{ascii}(c_j)$, where t is the most frequent term;

b is equal to $\sum_{j=1}^{|t|} \text{ascii}(c_j)$, where t is the less frequent term;

DR_2 :

a is equal to $\sum_{i=1}^{10} \sum_{j=1}^{|t_i|} \text{ascii}(c_{ij})$, where t_i is one of the 10 most frequent terms;

b is equal to $\sum_{i=1}^{10} \sum_{j=1}^{|t_i|} \text{ascii}(c_{ij})$, where t_i is one of the 10 less frequent terms;

DR_3 :

a_j is equal to the frequency mean of the corpus term t_j . In other words, given a corpus with n documents,

$$a_j = \bar{x}_j = \frac{1}{n} \sum_{i=1}^n freq(t_j, d_i), \quad (2)$$

whereas $freq(t_j, d_i)$ is the frequency of the term t_j in the document d_i .

b_j is equal to the frequency standard deviation of the corpus term t_j . In other words, given a corpus with n documents,

$$b_j = \sigma_j = \sqrt{\frac{1}{n} \sum_{i=1}^n (freq(t_j, d_i) - \bar{x}_j)^2}, \quad (3)$$

where $freq(t_j, d_i)$ is the frequency of the term t_j in the document d_i .

In the following section we show the obtained results after evaluating the above presented approaches in a document collection gathered for information retrieval purposes.

3 Experiments

The aim of the aforementioned document representation schemata is to detect discriminant terms. In order to visualize the appropriate representation of the documents, we present in this paper the elliptic curves of one document collection (see corpus C1 in Section 3.1). Each figure correspond to one approach proposed. The rest of the curves are also available, but due to space limitations these were not included in this paper.

In Figure 2 we may observe the DR_1 approach. As we may see, having considered only two terms for representing the documents lead us to have a very ambiguous representation schema. In this Figure is quite difficult to distinguish a clear division among the elliptic curves. The stepforward is to verify whether or not, adding more terms would improve the document representation. Figure 3 show a set of elliptic curves in which we have considered the 10 most and less frequent terms in order to represent each document. Again, we observe that the parameters do not assist for the correct representation of the document. We consider that, in particular, the second parameter (b =less frequent terms) is not helpful due to the high number of terms with frequency one in the vocabulary of the corpus.

In order to analyze the degree of discrimination that each term of the corpus of evaluation has in the representation of documents, in Figure 4 we have plotted the approach DR_3 . As it may be observed in this figure, this representation schema offers (at least from the visual point of view) a set of curves that allow to study the behavior of each corpus term, in order to determine the discrimination degree of each one of them.

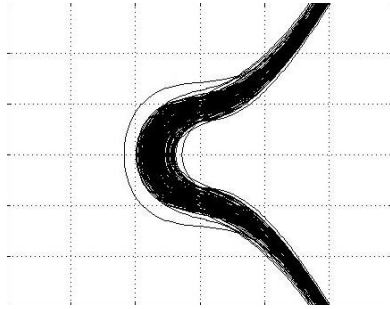


Fig. 2. Elliptic curves with approach DR_1 for corpus C1

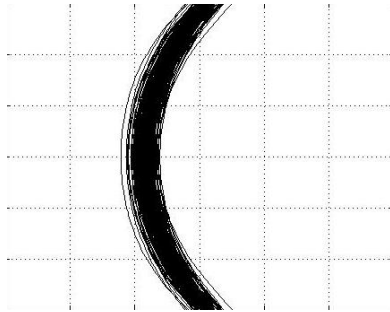


Fig. 3. Elliptic curves with approach DR_2 for corpus C1

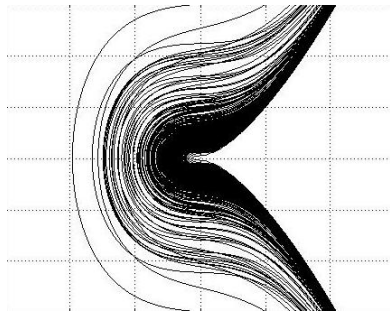


Fig. 4. Elliptic curves with approach DR_3 for corpus C1

Table 1. Corpora used in the experiments

Corpus name	Num. of docs	Vocabulary size	Maximum frequency	Max frequent term	Terms with frequency one
C1	210	15631	1299	México	7786
C3	164	12156	646	México	6160
C4	97	13533	352	México	8878
C5	256	21083	796	México	13179
C10	206	13851	686	México	6976
C11	105	8836	371	México	4676
C14	280	15751	1709	PEMEX	7630
C15	7	1357	28	México	1006

Having analyzed the three different schemata, we decided to evaluate the DR_3 approach with a greater number of documents (eight corpus). In the following subsection we describe the dataset used in these experiments. In subsection 3.2 we present the evaluation of the different document collections. Finally, we conclude this section discussing the findings of this investigation.

3.1 Dataset

In order to observe the degree of discrimination of each term, we consider a group of documents that hold some kind of similarity among them. In this case, we have selected a collection of Mexican newspaper text in the Spanish language that was used in one competition of information retrieval¹. Each group corresponds to a set of relevant documents for a given topic/query. For instance, the first corpus is made up of documents relevant to the query “Mexican Opposition to the North American Free Trade Agreement (Oposición Mexicana al TLC)”. The name we gave to each corpus, together with other features such as the vocabulary size, the total number of terms, the maximum frequency (with the associated term) and the number of terms with frequency one are shown in Table 1. We attempted to provide various features in the evaluated corpus in order to be able to obtain some conclusions of the implemented document representations.

3.2 Evaluation

The final aim of our investigation is to find an appropriate representation of each document by means of an elliptic curve. If we are able to find this curve, then we would easily determine a simple similarity measure between any pair elliptic curves and, therefore between the two corresponding documents.

In order to do so, we first need to determine the most representative terms. That is the reason because we have split the whole corpus vocabulary in various groups of terms. The different thresholds used in the partitions, together with an ID we have assigned to each partition, are given in Table 2.

¹<http://trec.nist.gov/>

Table 2. Thresholds used for the DR_3 representation approach

ID	Parameter thresholds
HIGH	$\bar{x}_j \in [1.0, \infty) \wedge \sigma_j \in [1.0, \infty)$
MEDIUM	$\bar{x}_j \in [0.1, 1.0) \wedge \sigma_j \in [0.1, 1.0)$
MEDIUM-LOW	$\bar{x}_j \in (0, 1.0) \wedge \sigma_j \in (0, 1.0)$
LOW	$\bar{x}_j \in (0, 0.1) \wedge \sigma_j \in (0, 0.1)$

The rationale of the aforementioned thresholds follows. HIGH was proposed with the aim of capture those terms that appear, in average, one time in each document. The standard deviation (high) in this case permits to obtain those terms whose distribution along the document collection is not uniform. We hypothesize that these thresholds would allow to obtain the best discriminant terms. MEDIUM get those terms with a lower frequency than HIGH, but the occurrence of these terms is more or less uniform through the corpus. The LOW set of thresholds bring together the terms that uniformly and nearly not appear in the corpus. Finally, MEDIUM-LOW is proposed with the goal of observing the behavior of these terms in the document representation.

In Figure 5 we may observe the behavior of each group of terms when calculating the similarity among all the documents of each corpus evaluated. Each square represents the similarity of the documents when we use only those terms that fulfill the thresholds defined in Table 2. From left to right, each square uses the HIGH, MEDIUM, MEDIUM-LOW and LOW parameters, respectively.

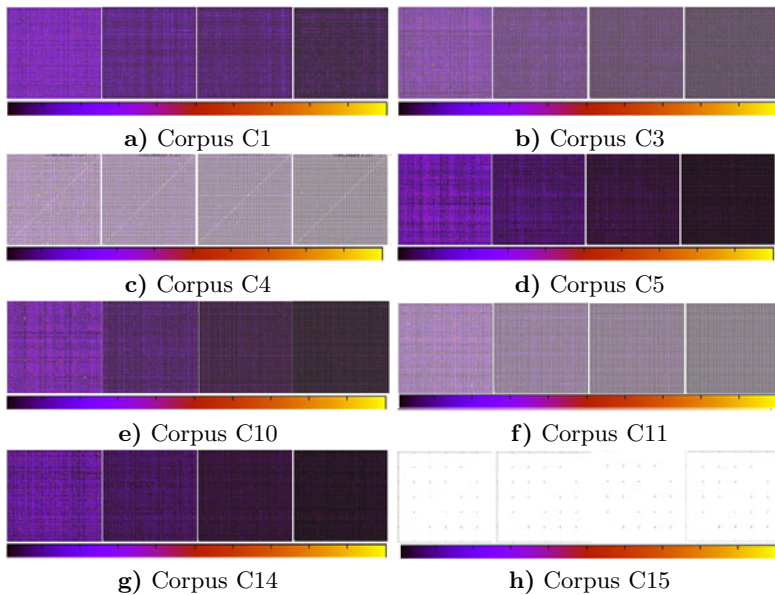


Fig. 5. Profile of similarity for all corpora

The lighter is one point in the square, the higher is the similarity between the two documents associated. We may observe that in all cases the HIGH representation obtains the best degree of similarity among the documents. We consider that this result is obtained due to the nature of the corpora used in the experiments. All the corpus belong to the information retrieval field and, therefore, the documents were grouped based on the frequency of their terms.

Figure 5 shows the expected behavior on document representation: the more frequent a term is, the better its degree of discrimination. Therefore, the DR_3 schema has shown to be a good representation of the corpus features.

These experiments are a first step towards the definition of proper document representation based on elliptic curves. As future work, we are considering to merge all the means and standard deviations in a vectorial representation which should be used as parameter for the elliptic curves.

4 Conclusions and Further Work

In this paper we have presented an study of the use of elliptic curves for term discrimination with the final purpose of finding an appropriate document representation. The aim is to have a simple and fast method for classifying and retrieving information from huge amount of documents.

We have evaluated three different approaches that consider the frequency of the terms in the corpus. Both, the most and less frequent terms were evaluated in order to observe their behavior in the document representation task.

In general, we have found that the most discriminant terms in the corpora used in the experiments carried out are those that appear, in average, one time in each document ($\bar{x}_j \geq 1$) with high standard deviation ($\sigma_j > 1$), i.e., those terms whose distribution along the document collection is not uniform.

However, there exist some cases in which other term frequencies allow to improve the precision of the task implemented. Therefore, it is important to further analyze a robust representation that permits to include such characteristics in a simple elliptic curve. We still need to determine a mechanism in order to integrate the characteristics of each term of a given document in a simple parameter of the elliptic curve. Further experiments will be carried out following this research line.

In conclusion, based on these preliminar results, we consider that it is possible to use the theory of elliptic curves as representation schema in order to succesfully characterize documents.

References

1. Can, F., Ozkarahan, E.A.: Computation of term/document discrimination values by use of the cover coefficient concept. *Journal of the American Society for Information Science* 38(3), 171–183 (1987)
2. Manning, D.C., Schütze, H.: *Foundations of statistical natural language processing*. MIT Press, Cambridge (1999)

3. Pinto, D.: On Clustering and Evaluation of Narrow Domain Short-Text Corpora. Phd thesis, Department of Information Systems and Computation, UPV (2008)
4. Salton, G., Wong, A., Yang, C.: A vector space model for automatic indexing. *Communications of the ACM* 18(11), 613–620 (1975)
5. Montemurro, M.A., Zanette, D.H.: Entropic analysis of the role of words in literary texts. *Advances in Complex Systems (ACS)* 05(01), 7–17 (2002)
6. Pons-Porrata, A., Berlanga-Llavori, R., Ruiz-Shulchloper, J.: Topic discovery based on text mining techniques. *Information Processing and Management* 43(3), 752–768 (2007)
7. Santiesteban, Y., Pons-Porrata, A.: LEX: a new algorithm for the calculus of typical testors. *Mathematics Sciences Journal* 21(1), 85–95 (2003)
8. Hankerson, D., Menezes, A.J., Vanstone, S.: *Guide to Elliptic Curve Cryptography*. Springer, New York (2003)

Author Index

- Abdullah, Siti N.H.S. 230
 Acevedo, Antonio 103
 Acevedo, Elena 103
 Aguilar-González, Pablo M. 194
 Alarcon-Aquino, Vicente 146, 240
 Almanza, Victor 95
 Altamirano, Luis C. 50
 Andina, D. 260
 Ascencio-Lopez, J.I. 155
 Ayala, Francisco J. 297
 Ayala-Ramirez, Victor 220

 Balderas, Carlos 341
 Barrientos, Mario 211
 Bataineh, Bilal 230
 Batyrshin, Ildar 85, 95
 Bautista, César 50
 Bautista-Villavicencio, David 34
 Beltrán, Beatriz 341
 Bonet, Isis 67
 Boyer, Kim L. 1
 Buhmann, Joachim M. 12

 Cabrera, Sergio 305
 Castro-Sánchez, Noé Alejandro 322
 Chacon-Murguía, Mario I. 118, 305
 Chakraborty, Debrup 278
 Chavez, Edgar 75
 Chavoya, Arturo 269
 Coello Coello, Carlos A. 22
 Conant-Pablos, Santiago Enrique 250
 Cortina-Januchs, M.G. 260
 Cosultchi, Ana 85
 Cruz-Barbosa, Raúl 34
 Cruz-Santiago, Rene 184
 Cruz-Techica, Sonia 202

 De Ita, Guillermo 50
 Díaz-Pérez, A. 164

 Escalante-Ramirez, Boris 202

 Faidzul, M. 230
 Felipe, Federico 103
 Figueroa Mora, Karina 42

 Galan-Hernandez, J.C. 240
 Gallegos-Funes, Francisco J. 184
 García, María M. 67
 Gómez-Flores, W. 164
 Gómez-Gil, Pilar 288
 Gonzalez-Fraga, J.A. 155
 Graff, Mario 75
 Grau, Ricardo 67

 Hamouda, Atef 136
 Hentati, Jihen 136
 Herrera, Abel 297

 Justo, Raquel 332

 Kober, Vitaly 194

 Lizarraga-Morales, Rocio A. 220
 López-Martín, Cuauhtémoc 269

 Madrid, Humberto 211
 Marcano-Cedeño, A. 260
 Meda-Campaña, M.E. 269
 Mihai, Cosmin 174
 Minaei-Bidgoli, Behrouz 60
 Mújica-Vargas, Dante 184

 Naouai, Mohamed 136
 Nava-Ortiz, M. 164

 Ojeda-Magaña, B. 260
 Omar, K. 230
 Orozco-Monteagudo, M. 174
 Ortiz-Bayliss, José Carlos 250

 Paniagua, Sofia 341
 Paredes, Rodrigo 42
 Parvin, Hamid 60
 Perez-Vargas, Francisco J. 118
 Pinto, David 341

 Quezada-Holguín, Yearim 305
 Quintanilla-Domínguez, J. 260

 Ramirez-Cortes, J.M. 240
 Rangel, Roberto 42
 Reyes-García, Carlos A. 288

- Rivas-Perea, Pablo 305
Rodríguez, Abdel 67
Rodríguez-Asomoza, Jorge 146
Rosales-Pérez, Alejandro 288
Rosas-Romero, Roberto 146
Ruelas, R. 260
- Sahli, Hichem 174
Salas, Joaquín 127
Sanchez-Yanez, Raul E. 220
Santiago-Ramirez, Everardo 155
Shahpar, Hamideh 60
Sheremetov, Leonid 85
Sidorov, Grigori 322
Silván-Cárdenas, José Luis 314
Starostenko, Oleg 146, 240
- Taboada-Crispi, A. 174
Tellez, Eric Sadit 75
Terashima-Marín, Hugo 250
Tomasi, Carlo 127
Torres, M. Inés 332
Toscano-Pulido, G. 164
Tovar, Mireya 341
- Vázquez-Santacruz, Eduardo 278
Vega-Corona, A. 260
Velasco-Hernandez, Jorge 85
Vellido, Alfredo 34
Vilarino, Darnes 341
- Wang, Le 314
Weber, Christiane 136
Wu, Dijia 1