Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
Joaquín Salas Rodríguez
Gabriella Sanniti di Baja (Eds.)

LNCS 7914

Pattern Recognition

5th Mexican Conference, MCPR 2013
Querétaro, Mexico, June 2013
Proceedings

Springer

Lecture Notes in Computer Science 7914

Commenced Publication in 1973
Founding and Former Series Editors:
Gerhard Goos, Juris Hartmanis, and Jan van Leeuwen

Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
Joaquín Salas Rodríguez
Gabriella Sanniti di Baja (Eds.)

Pattern Recognition

5th Mexican Conference, MCPR 2013
Querétaro, Mexico, June 26-29, 2013
Proceedings

Springer

Volume Editors

Jesús Ariel Carrasco-Ochoa
Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)
72840 Sta. Maria Tonantzintla, Puebla, Mexico
E-mail: ariel@inaoep.mx

José Francisco Martínez-Trinidad
Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)
72840 Sta. Maria Tonantzintla, Puebla, Mexico
E-mail: fmartine@inaoep.mx

Joaquín Salas Rodríguez
Instituto Politécnico Nacional (IPN)
76090 Colinas del Cimatario, Queretaro, Mexico
E-mail: jsalasr@ipn.mx

Gabriella Sanniti di Baja
Istituto di Cibernetica "E. Caianiello", CNR
80078 Pozzuoli, Naples, Italy
E-mail: g.sannitidibaja@cib.na.cnr.it

ISSN 0302-9743 e-ISSN 1611-3349
ISBN 978-3-642-38988-7 e-ISBN 978-3-642-38989-4
DOI 10.1007/978-3-642-38989-4
Springer Heidelberg Dordrecht London New York

Library of Congress Control Number: 2013940329

CR Subject Classification (1998): I.2, I.4, I.5, H.3, F.1, H.4

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

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 2013 Mexican Conference on Pattern Recognition (MCPR 2013) was the fifth event in the series. The conference was jointly organized between the National Polytechnic Institute (IPN) and the Computer Science Department of the National Institute for Astrophysics Optics and Electronics (INAOE) of Mexico, under the auspices of the Mexican Association for Computer Vision, Neurocomputing and Robotics (MACVNR), which is a member society of the International Association for Pattern Recognition (IAPR). The MCPR series of conferences provides a forum for the exchange of scientific results, practice, and recently adquired knowledge, and it also promotes cooperation among research groups in pattern recognition and related areas in Mexico and the rest of the world.

MCPR 2013, held in Queretaro, Mexico, attracted as in prior years worldwide participation. Contributions were received from 18 countries. Out of the 81 papers received, 36 were accepted. All submissions were strictly peer-reviewed by the Technical Program Committee, which consisted of 83 outstanding researchers, all of whom are specialists in pattern recognition.

The Technical Program of the conference included lectures by the following distinguished keynote speakers: Roberto Manduchi, Departament of Computer Engineering, University of California at Santa Cruz, USA; Raul Rojas, Department of Mathematics and Computer Science, Free University of Berlin, Germany; and Sai Ravela, Department of Earth, Atmospheric and Planetary Sciences, Massachusetts Institute of Technology, USA. They, along with Robert Pless, Department of Computer Science and Engineering, Washington University in St. Louis, and Edgar F. Roman-Rangel Computer Vision and Multimedia Lab, CUI, University of Geneva, presented enlightening tutorials during the the conference. To all of them, we express our sincere gratitude for these presentations.

This conference would not have been possible without the efforts of many people. In particular, we are grateful to the members of the Technical Program Committee, to all of the authors who submitted papers to the conference, and last but not least, to the members of the local Organizing Committee.

For this edition of MCPR, the authors of accepted papers were invited to submit expanded versions of their papers for possible publication in a thematic special issue titled "Pattern Recognition Applications in Computer Vision and Image Analysis" to be published in Elsevier's *Pattern Recognition*, the journal of the Pattern Recognition Society.

We hope that MCPR will continue to provide a fruitful forum to enrich collaborations between Mexican researchers and the broader international community in the area of pattern recognition.

June 2012

Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
Joaquín Salas
Gabriella Sanniti di Baja

Organization

MCPR 2013 was sponsored by the Computer Science Department of the National Institute of Astrophysics, Optics and Electronics (INAOE) and the National Polytechnic Institute (IPN) of Mexico.

General Conference Co-chairs

Gabriella Sanniti di Baja	Istituto di Cibernetica Eduardo Caianiello, Italy
Jesús Ariel Carrasco-Ochoa	National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
José Francisco Martínez-Trinidad	National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
Joaquín Salas	Research Center on Applied Science and Advanced Technology (CICATA) of National Polytechnic Institute (IPN) of Mexico, Mexico

Local Arrangements Committee

Alvarado Alvarado Marisol
Álvarez Herrera Arturo
Azarcoya Cabiedes Willy
Cerón Benítez Gorgonio
Cervantes Cuahuey Brenda Alicia
Flores Ramírez Alma Lucero
González Veronica Aidee
López Lucio Gabriela
Meza Tlalpan Carmen
Muñoz Olín Edith
Ortiz Dania Carolina
Ramírez Pedraza Pedro Alfonso
Reséndiz Hernández Gabriela
Sandoval Iván
Terven Salinas Juan Ramón
Vera Alfaro Pablo
Zenteno Daniel

Scientific Committee

Asano, A.	Kansai University, Japan
Barajas, J.	University of California Santa Cruz, USA
Batyrshin, I.	Mexican Petroleum Institute, Mexico
Benedi, J.M.	Universidad Politécnica de Valencia, Spain

Bigun, J.	Halmstad University, Sweden
Borges, D.L.	Universidade de Brasília, Brazil
Caballero, K.L.	University of California Santa Cruz, USA
Castelan, M.	CINVESTAV-Saltillo, Mexico
Chollet, G.	ENST, France
Cuervo-Lafayye, E.	HP Labs, USA
Dickinson, S.	University of Toronto, Canada
Dornaika, F.	University of the Basque Country, Spain
Ebner, M.	Ernst Moritz Arndt Universität Greifswald, Germany
Facon, J.	Pontifícia Universidade Católica do Paraná, Brazil
Ferri, F.J.	Universitat de València, Spain
Fred, A.	Instituto Superior Técnico, Portugal
Gainza, P.	Duke University, USA
Gatica, D.	Idiap Research Institute, Switzerland
Gelbukh, A.	CIC-IPN, Mexico
Goldfarb, L.	University of New Brunswick, Canada
Gomes, H.	Universidade Federal de Campina Grande, Brazil
González, J.	Universitat Autònoma de Barcelona, Spain
González-Barbosa, J.J.	CICATA-IPN, Mexico
Graña, M.	University of the Basque Country, Spain
Grau, A.	Universitat Politécnica de Catalunya, Spain
Heutte, L.	Université de Rouen, France
Hurtado-Ramos, J.B.	CICATA-IPN, Mexico
Igual, L.	University of Barcelona, Spain
Jiang, X.	University of Münster, Germany
Kampel, M.	Vienna Univerity of Technology, Austria
Klette, R.	University of Auckland, New Zealand
Kober, V.	CICESE, Mexico
Koster, W.	Universiteit Leiden, The Netherlands
Lacroix, S.	LAAS-CNRS, France
Laurendeau, D.	Université Laval, Canada
Lopez-de-Ipiña-Peña, M.K.	Universidad del País Vasco, Spain
Lorenzo-Ginori, J.V.	Universidad Central de Las Villas, Cuba
Manduchi, R.	University of California Santa Cruz, USA
Mascarenhas, N.D.	University of São Paulo, Brazil
Mayol-Cuevas, W.	University of Bristol, UK
Menezes, P.	University of Coimbra-Polo II, Brazil
Mihailidis, A.	University of Toronto, Canada
Mora, M.	Catholic University of Maule, Chile
Morales, E.	INAOE, Mexico

Nolazco, J.A.	ITESM-Monterrey, Mexico
Pardo, A.	Universidad Católica del Uruguay, Uruguay
Peynot, T.	University of Sydney, Australia
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
Radeva, P.	University of Barcelona, Spain
Raducanu, B.	Universitat Autònoma de Barcelona, Spain
Raposo-Sanchez, J.M.	Instituto Superior Técnico, Portugal
Real, P.	University of Seville, Spain
Rios-Cabrera, R.	KU Leuven Nederlands, Belgium
Rodriguez, R.	ICIMAF, Cuba
Rojas, R.	Free University of Berlin, Germany
Roman-Rangel, E.F.	University of Geneva, Switzerland
Ross, A.	West Virginia University, USA
Rueda, L.	University of Windsor, Canada
Ruiz-Shulcloper, J.	CENATAV, Cuba
Sahbani, A.	Universitè Pierre et Marie Cuerie, France
Sanchez-Cortes, D.	Idiap Research Institute, Switzerland
Sang-Woon, K.	Myongji University, South Korea
Sansone, C.	Università di Napoli, Italy
Santana, R.	Universidad Politécnica de Madrid, Spain
Sappa, A.	Universitat Autònoma de Barcelona, Spain
Scinzas, C.	University of Cyprus, Cyprus
Shi-Hua, Z.	Nanjing University, China
Sossa Azuela, J.H.	CIC-IPN, Mexico
Sousa-Santos, B.	Universidade de Aveiro, Portugal
Spyridonos, P.	University of Loannina, Greece
Stathaki, T.	Imperial College London, UK
Sucar, L.E.	INAOE, Mexico
Suleiman, W.	University of Sherbrooke, Canada
Taati, B.	University of Toronto, Canada
Tomasi, C.	Duke University, USA
Valev, V.	Institute of Mathematics and Informatics, Bulgaria
Vitria, J.	University of Barcelona, Spain
Wang, S.	University of Sherbrooke, Canada
Yoshiaki, S.	Ritsumeikan University, Japan
Zagoruiko, N.G.	Russian Academy of Sciences, Russia
Zhi-Hua, Z.	Nanjing University, China

Additional Reviewers

Aguena M.
Assoum, A.
Bailly, K.
Buch, A.
Clady, X.
Dias, P.
Durandau, G.
Gómez-Gil, M.P.
Hermann, S.
Jarrassé, N.
Lazo-Cortés, M.
Ledeneva, Y.
Mustafa, W.
Nicolle, J.
Olvera-López, J.A.
Orihela-Espina, F.
Rezaei M.
Salvadeo D.
Tao, J.
Torres-Méndez, L.A.
Urcid-Serrano, G.
Viveros-Jiménez, F.
Xu, M.
Zhila, A.

Sponsoring Institutions

National Institute of Astrophysics, Optics and Electronics (INAOE)
Mexican Association for Computer Vision, Neurocomputing and Robotics (MACVNR)
International Association for Pattern Recognition (IAPR)
National Council for Science and Technology of Mexico (CONACyT)
Secretariat of Public Education of Mexico (SEP)
Council for Science and Technology of Queretaro (CONCYTEQ)
Operating Commission for Promoting Academic Activities of the National Polytechnic Institute of Mexico (COFAA-IPN)

Table of Contents

Kenote Addresses

Computer Vision

Image Processing

Pattern Recognition and Artificial Intelligence

Neural Networks

Document Processing

Empirical Characterization of Camera Noise

Jeremy Baumgartner[1], Markus Hinsche[2], and Roberto Manduchi[1]

[1] University of California, Santa Cruz, CA
{jbaumgar,manduchi}@ucsc.edu
[2] Hasso Plattner Institute, Potsdam, Germany
markus.hinsche@student.hpi.uni-potsdam.de

Abstract. Noise characterization is important for several image processing operations such as denoising, thresholding, and HDR. This contribution describes a simple procedure to estimate the noise at an image for a particular camera as a function of exposure parameters (shutter time, gain) and of the irradiance at the pixel. Results are presented for a Pointgrey Firefly camera and are compared with a standard theoretical model of noise variance. Although the general characteristic of the noise reflects what predicted by the theoretical model, a number of discrepancies are found that deserve further investigation.

1 Introduction

The quantitative estimation of image noise is critical for basic operations such as denoising [5], thresholding [6], and HDR [2]. The simplifying assumption of "uniformly distributed Gaussian noise" in images is well known to be incorrect: for the same camera, the statistical characteristics of noise depend on the exposure parameters as well as on the irradiance received by the pixel under consideration. Theoretical noise models and procedures for noise parameter estimation have been described by several authors [3,7,4,8]. In general, previous work either assumes access to raw data from the sensor, or tries to "reverse engineer" the image signal processor (ISP) that performs operations such as gamma correction, gamut mapping, and white point calibration, in order to estimate the "true" irradiance at a pixel and the noise characteristics of the acquisition process. Published work ranges from methods that assume a well-controlled illumination and reflection surface [3], to approaches that attempt to estimate relevant parameters from a single image [5,1]. In this work we take an intermediate stance: we assume that a number of pictures of a stationary backdrop are taken with a number of different exposure settings, but make no particular assumption about the illumination and reflectance characteristics of the scene, except that they should remain constant during data acquisition. This can be easily achieved in a standard lab environment. For each exposure setting, pixels with similar mean value of brightness are pooled together for noise variance estimation. This procedure produces a characterization of camera noise as a function of the mean brightness value and of exposure parameters of interest (shutter time and gain). The results can be used to validate theoretical models and compute relevant

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 1–9, 2013.

parameters, or could be stored as a look-up table to be queried when needed. This may be especially useful when using image data that has been already processed by the ISP with unknown non-linear characteristics.

This contribution is organized as follows. We first summarize in Sec. 2 the standard noise model for photodiode-based sensors. We then describe our simple experimental setup in Sec. 3 and our procedure for estimation of noise variance in Sec. 4. This procedure, which pools together pixels with similar mean value of brightness, is an original contribution of this work. We then present an analysis of the results vis-a-vis the theoretical model in Sec. 5. Sec. 6 has the conclusions.

2 Noise Model

In this section we summarize the standard camera noise model [3]. Let P be the power of light impinging on a pixel (equal to the irradiance at the pixel multiplied by the pixel area times the fill factor). Light generates a photocurrent I, equal to P times the quantum efficiency of the photodiode. Due to photon noise, I should be regarded as a Poisson process with rate parameter $\bar{I}$. Dark current I_{DC} also contributes to the charge accumulated at the pixel. This can also be modeled a Poisson process with rate parameter $\bar{I}_{DC}$ We will assume the I and I_{DC} are independent.

The charge accumulated at a pixel is equal to $Q = (I + I_{DC}) \cdot T$, where T is the shutter time. Since Q is the sum of two independent Poisson variables, it is itself distributed as a Poisson variable with mean $(\bar{I} + \bar{I}_{DC}) \cdot T$ and variance equal to its mean. This charge is then transferred to the amplifier. The main purpose of the amplifier is to increase the range of the signal that is passed to the quantizer, thus increasing the quantization SNR. Read-out (thermal) noise n_{RO} is generated by the amplifier; this noise is usually modeled as a zero-mean Gaussian variable with variance σ^2_{RO}. If G is the amplifier gain, the amplified signal is thus equal to $((I + I_{DC}) \cdot T + n_{RO}) \cdot G$. The amplified signal is then quantized with N bits. The quantization process can, in first approximation, be modeled by the addition of a uniform random variable with zero mean and variance equal to $\sigma^2_{qt} = \Delta^2/12$, where Δ is the quantization interval. Note that, referring to the measured values represented as integers, the quantization interval is equal to 1, and thus the quantization noise variance of the measured values ("brightness", denoted by B) is equal to $1/12$. Summarizing, the brightness value B measured at a pixel is a random variable with mean $\bar{B}$ equal to:

$$\bar{B} = (\bar{I} + \bar{I}_{DC}) \cdot TG \tag{1}$$

and variance σ^2_B equal to:

$$\sigma^2_B = (\bar{I} + \bar{I}_{DC}) \cdot TG^2 + \sigma^2_{RO} \cdot G^2 + \frac{1}{12} = \bar{B} \cdot G + \sigma^2_{RO} \cdot G^2 + \frac{1}{12} \tag{2}$$

where we have assumed that the sources of noise are uncorrelated with each other. Note that the read-out noise is expressed in units of measured values. It is noteworthy that the noise on the measured values is independent of the shutter time.

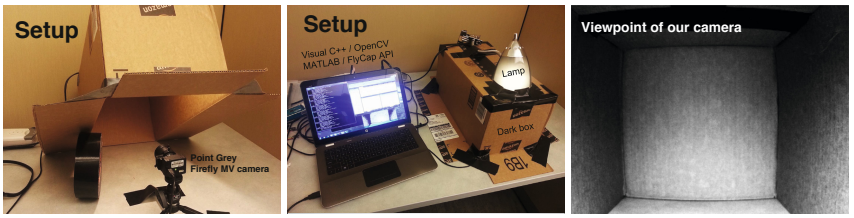


Fig. 1. Our experimental setup

3 Experimental Setup

We devised a data collection procedure that can be easily replicated in standard lab settings, without the need for expensive equipment or for equipment that requires accurate calibration. The goal is to collect a large number of images of a stationary background with different values of shutter time and gain. We do not attempt to measure the irradiance at each pixel directly, nor do we expect that, for a given picture, the irradiance is uniform across the image. In fact, due to non-uniform illumination, shading, off-axis attenuation and other factors, a highly non-uniform distribution of irradiance values should be expected for a given picture. The main requirement for our data collection is that the illumination of the (stationary) background being imaged is constant with time. In practice, this means using an artificial light source, and avoiding other sources of possibly changing light as well as shadows due to moving people in the room. For our experiments, we placed the camera inside a cardboard box as shown in Fig. 1. A lamp illuminated the inside of the box through a hole in one side of the box. The lamp was left on for a certain amount of time before data collection to ensure stable illumination.

We used a Firefly MV 0.3 MP Color USB 2.0 camera with a Fujinon 2.8mm lens. This camera uses an Aptina MT9V022 CMOS sensor. Data from the sensor is first adjusted for gain and brightness (bias) and then quantized at 10 bits, after which gamma correction is applied followed by pixel correction, demosaicing and white balance, before quantization to 8 bits. We should note that the Firefly camera give the option of accessing raw data; we decided to use the 8-bit, demosaiced output as this is the format most commonly used in practical applications. However, we disabled the gamma correction and white balance in our tests in order to obtain a linear and scene-independent response. Color data was transformed to greyscale (luminosity) using the standard conversion formula.

We collected 50 images of the scene for several combinations of shutter time T and gain G. We considered values of T from 3 ms to 13 ms in steps of 2 ms, and values of G from 0 dB to 12 dB in steps of 2 dB. Overall, 2100 pictures were collected.

4 Variance Estimation Procedure

The goal of this work is to estimate the variance of the measured value B as a function of its mean value $\bar{B}$, shutter time T, and gain G. Inspired by [8], we use a "pooling" approach, whereby pixels that are assumed to receive the same irradiance are grouped together to compute useful statistics. Unlike [8], though, we don't pool together pixels in an image with similar values of B. Instead, for each choice of T and G, we first estimate the mean value $\bar{B}$ at each pixel by averaging over the 50 images taken for that (T,G) pair. The mean value $\bar{B}$ is again quantized to 8 bits. For each one of the 256 possible values of $\bar{B}$, we compute the location of pixels that take on that particular value of $\bar{B}$. This set of pixels are assumed to receive the same irradiance, and thus to be samples of the same random variable. The sample variance computed from this pool of pixels over the 50 images represents our estimation of the variance σ_B^2. In order to reduce the effect of estimation variance, only values of $\bar{B}$ for which the pool contains at least 10 pixels (500 values overall) are considered for this computation.

5 Analysis of Results

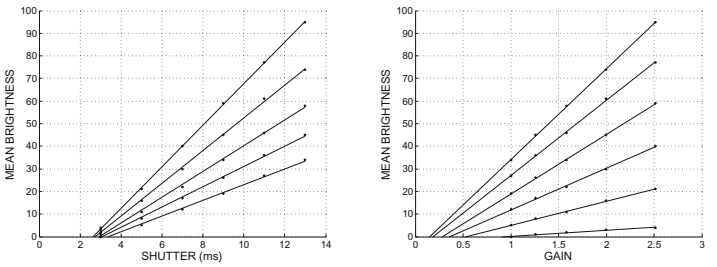


Fig. 2. The average value $\bar{B}$ at a pixel as a function of shutter time (left) and gain (right). Note that the gain is expressed on a linear scale. The dots represent measured values, while each line is a least square linear fit. Each line in the left (right) plot corresponds to one specific value of gain (shutter time).

5.1 Validation of Linearity Assumption

According to Eq. (1), the mean measured value $\bar{B}$ at a pixel should be proportional to both shutter time T and gain G. In order to validate this assumption, we plotted in Fig. 2 the mean value $\bar{B}$ of one pixel in the image (averaged over all 50 frames taken for each exposure setting) as a function of T and G. (Note that, for the pixel selected for these plots, the value $\bar{B}$ at the maximum gain and shutter time was equal to 95.) For each plot (representing a fixed value of G or T), we also computed and plotted the least squares linear fit.

From Fig. 2 it is clear that, for fixed G, the function $\bar{B}(T, G)$ is linear with non-null intercept. Least-squares fitting with a common intercept for all values of G results in an intercept of $T_0 = 2.95$ ms with reasonably small residual. Thus, the linear form in Eq. (1) represents an appropriate model if one substitutes T with $T - T_0$. The characteristic $\bar{B}(T, G)$ for fixed T is also linear with non-null intercept, but least-squares estimation of a common intercept produces fairly large residuals.

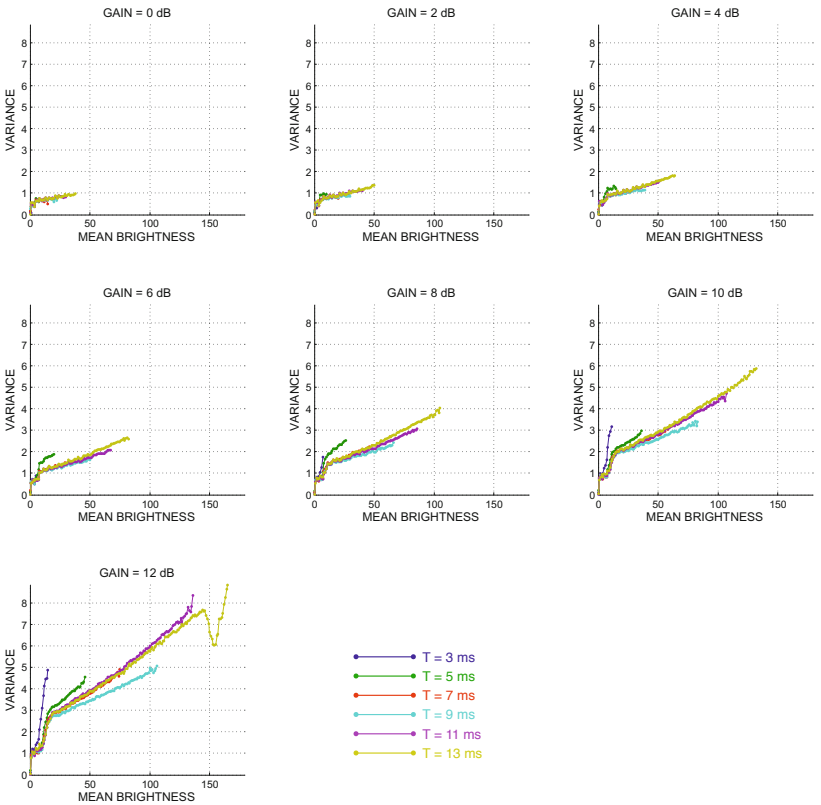


Fig. 3. The variance of B as a function of the mean brightness $\bar{B}$. Each subfigure represents a different value of the gain G; each curve corresponds to a different value of the shutter time T.

5.2 Noise Variance as a Function of $\bar{B}$

According to Eq. (2), for a given value of the gain G, the variance of the measured values B should be a linear function of $\bar{B}$ with intercept on the Y axis at $\sigma_{RO}^2 G^2 + 1/12$. In Fig. 3 we plotted the measured values of σ_B^2 as a function of $\bar{B}$ for all

values of G and T. (Note that, when the gain and/or shutter time are small, then $\bar{B}$ can only take small values.) The plots show that, for each value of the gain G, the characteristic of σ_B^2 as a function of $\bar{B}$ is indeed approximately linear with intercept for $\bar{B}$ larger than a certain value. However, when $\bar{B}$ is smaller than this value, the variance drops. This phenomenon could be explained in part by the "floor bias": since B cannot take on negative values, the noise distribution is skewed for small $\bar{B}$. In addition, the linear slope seems to have an unexpected dependency on the shutter time. In particular, for small values of T (3-5 ms), the slope is quite large; for $T = 7$ ms, 11 ms and 13 ms the slope is constant, while for $T = 9$ ms the slope is noticeably smaller. Another strange inconsistency can be noticed for values of $\bar{B}$ around 150 with T=13 ms and G=12 dB, where the curve of σ_B^2 has an unexpected dip.

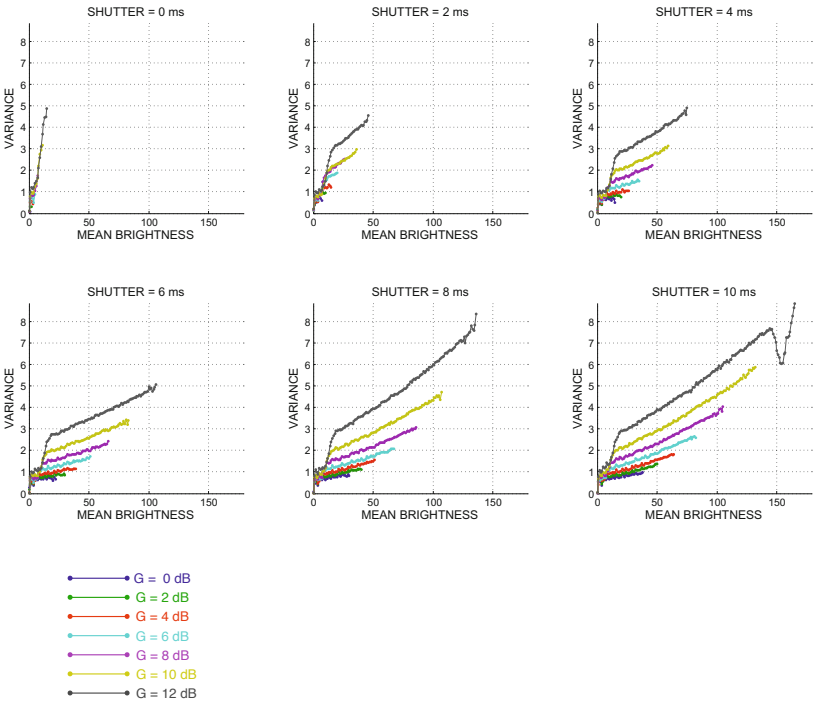


Fig. 4. The variance of B as a function of the mean brightness $\bar{B}$. Each subfigure represents a different value of the shutter time T; each curve corresponds to a different value of the gain G.

Fig. 4 plots the same data but grouping together curves with the same shutter time. As expected, the slope and intercept of the linear part of each plot depend on the gain G.

5.3 Noise Variance as a Function of T

As noted earlier, the shutter time T does not appear in the expression of the variance of B (2). Hence, for fixed $\bar{B}$ and G, the characteristic of σ_B^2 as a function of T is expected to be flat. Fig. 5 plots σ_B^2 against T for different values of $\bar{B}$ and G. It can be seen that the characteristic is only approximately flat, with noticeable variations around the mean value.

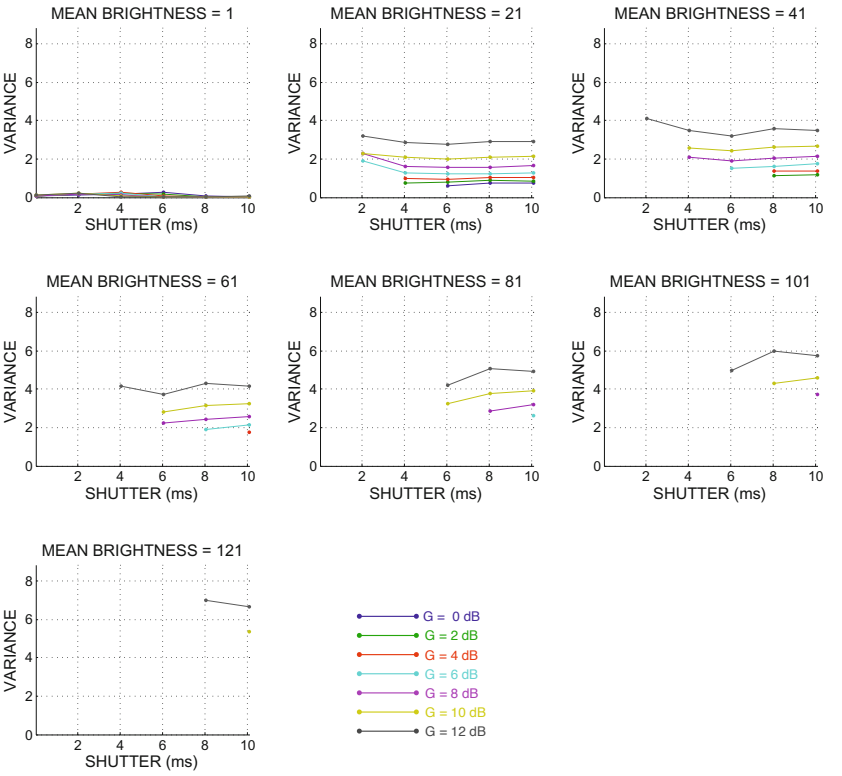


Fig. 5. The variance B as a function of the shutter time T. Each subfigure represents a different value of the mean brightness $\bar{B}$; each curve corresponds to a different value of the gain G.

5.4 Noise Variance as a Function of G

Each subfigure in Fig. 6 shows the variance of B as a function of the gain G for different values T and for a fixed value of $\bar{B}$. According to Eq. (2), these plots should overlap with each other (since they are independent of T) and should have a parabolic characteristic. It is seen that the plots do tend to overlap, except for T=5 ms and T=9 ms. A light concavity can be observed, consistent with the theory, although the linear term $\bar{B}G$ in Eq. (2) dominates in this interval

of values. Note that in both Fig. 5 and 6, the variance σ_B^2 takes on very small values for small $\bar{B}$; this phenomenon was already observed and discussed earlier in relation to Figs. 3 and 4.

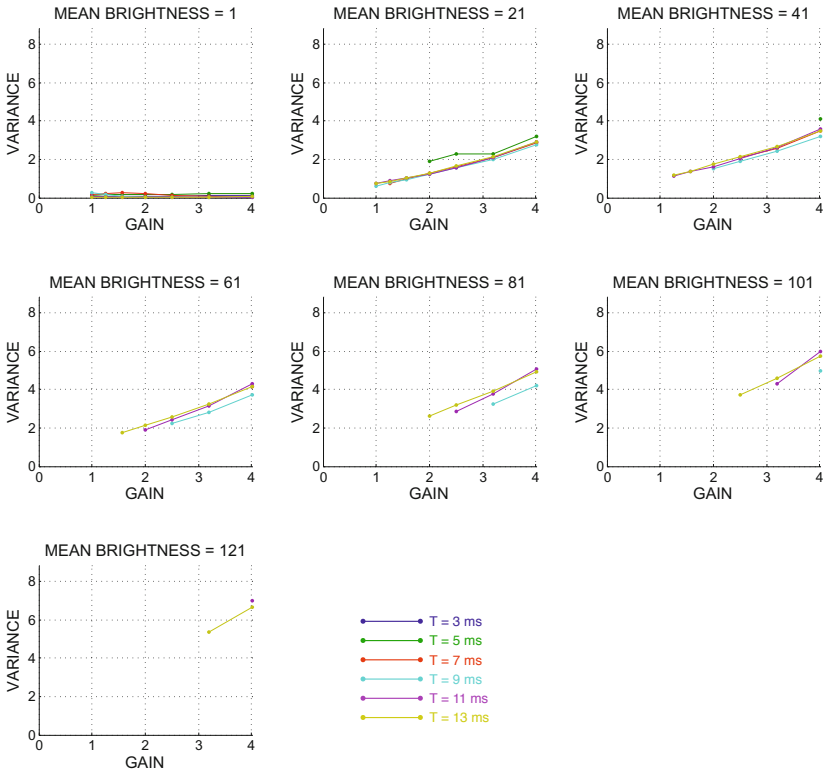


Fig. 6. The variance B as a function of the the gain G (shown in linear scale). Each subfigure represents a different value of the mean brightness $\bar{B}$; each curve corresponds to a different value of the shutter time T.

6 Conclusions

We have described a simple approach to computing the noise characteristics at a pixel as a function of the mean brightness value at the pixel and of the exposure parameters. This procedure can be easily reproduced without the need for calibrated illuminators or uniformly reflective material, and thus can be used to quickly characterize different camera models. We compared the characteristics of the mean brightness value and of the noise variance as a function of exposure parameters against the theoretical model. Our analysis brought to light a number of discrepancies that deserve further study, thus confirming the importance of direct empirical measurements for correct camera modeling.

Acknowledgment. This material is based upon work supported in part by the National Science Foundation under Grant No. IIS-083564. The authors would like to thank Dr. Orazio Gallo for useful advice.

References

1. Foi, A., Trimeche, M., Katkovnik, V., Egiazarian, K.: Practical Poissonian-Gaussian noise modeling and fitting for single-image raw-data. IEEE Transactions on Image Processing 17(10), 1737–1754 (2008)
2. Granados, M., Ajdin, B., Wand, M., Theobalt, C., Seidel, H.-P., Lensch, H.: Optimal HDR reconstruction with linear digital cameras. In: Proc. CVPR (2010)
3. Healey, G., Kondepudy, R.: Radiometric CCD camera calibration and noise estimation. IEEE Transactions on Pattern Analysis and Machine Intelligence 16(3), 267–276 (1994)
4. Janesick, J.: Scientific charge-coupled devices. SPIE Press (2001)
5. Liu, C., Szeliski, R., Kang, S., Zitnick, C., Freeman, W.: Automatic estimation and removal of noise from a single image. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(2), 299–314 (2008)
6. Rosin, P.: Thresholding for change detection. In: International Conference on Computer Vision, pp. 274–279 (1998)
7. Tian, H.: Noise Analysis in CMOS Image Sensors. PhD thesis, Stanford University (2000)
8. Tsin, Y., Ramesh, V., Kanade, T.: Statistical calibration of CCD imaging process. In: IEEE International Conference on Computer Vision (ICCV 2001) (July 2001)

Vision-Based Biometrics for Conservation*

Sai Ravela, James Duyck, and Chelsea Finn

Earth Signals and Systems Group Earth,
Atmospheric and Planetary Sciences
Massachusetts Institute of Technology

Abstract. Identifying individuals in photographs of animals collected over time is a non-invasive approach that enables ecological studies and conservation planning. Here we propose SLOOP, the first image retrieval system incorporating interactive image processing and matching tools with relevance feedback from crowdsourcing to solve large-scale individual identification for multiple species. One outcome is an advance in matching and image retrieval methodology; another is the creation of a community-based individual identification system that enables conservation planning.

1 Introduction

The development of effective conservation strategies for rare, threatened, and endangered species requires unbiased and precise information on their life history requirements and population ecology. Capture-Mark-Recapture (CMR) studies enable researchers to identify individual animals over time, which is particularly useful for questions related to individual growth and survival, dispersal and movement ecology, and reproductive strategies. CMR studies typically use techniques in which animals are physically marked or tagged. These methods are intrusive to varying degrees and some animals may be difficult to tag efficiently in large numbers. Alternative identification techniques that overcome some of these limitations are needed.

Numerous efforts have been made to identify individual animals using photo-identification methods but often manually and using ad hoc strategies. Manual matching is only feasible for small collections; at $10s$ per comparison, a $1,000$ photograph catalog will take 60 days, and a $10,000$-sized catalog will take about 15 years of nonstop work to analyze. Ad hoc strategies lead to imprecise quantitative analyses and they do not easily port across multiple species. A reusable automated or semi-automated identification system would better advance the application of individual identification in conservation biology.

We contend that image retrieval with crowdsourced relevance feedback offers the basis for large-scale, high performance biometrics for conservation. In this approach, an image retrieval system ranks the images in a database by *visual*

* This material is based upon work supported by AFOSR(FA9550-12-1-0313) and NSF DBI-1146747. Any opinion, findings, and conclusions or recommendations expressed in this material are those of the authors(s).

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 10–19, 2013.

similarity [1]. Instead of browsing all pairs, the user views a bounded number of retrievals to identify matches. Although not as automated as object recognition, high recall at substantially lower effort than manual matching can be achieved. The throughput can be improved by dynamically incorporating relevance feedback from user-identified matches. A 10,000-collection will require about three months to analyze. Crowdsourcing can accelerate this process further; the same task is completed in a few hours and at marginal financial cost. To be sure, a *big data* challenge in the form of a large number of visual features and relevance judgments must now be addressed.

The MIT SLOOP[1] system [2–4] is, to the best of our knowledge, the first such image retrieval system for animal biometrics that addresses these challenges. SLOOP has expanded to multiple species. It has an operational implementation at the Department of Conservation, Dunedin, New Zealand and finds mention in popular press [5]. SLOOP is interactive and scalable to distributed implementations and it also easily incorporates new methods in a collaborative framework. In this paper, we describe the SLOOP system architecture and methods, with application to several species.

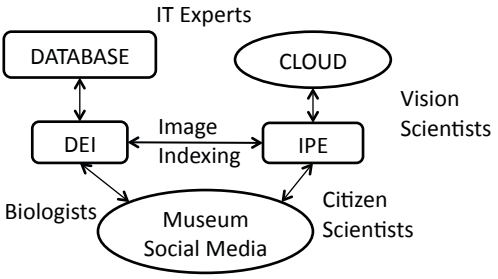


Fig. 1. SLOOP system architecture consists of interactions between a Data Exchange and Interaction (DEI) server and an Image Processing Engine, blending multiple areas of expertise into a single vision application

2 Related Work

There have been many efforts in the ecological community to use pattern recognition methods to aid individual identification. For example, species identification for plants is available [6], but SLOOP predates this system and is geared to individual identification of animals. Several efforts were made to identify individual animals using photo-identification methods [7–11] prior to SLOOP. They are advanced by incorporating new reusable matching techniques with rank aggregation and relevance feedback. The SLOOP system's underlying methods utilize several published techniques [1–4, 12–15].

[1] The name stands for a sloop observed Chris Yang and Sai Ravela saw over the Charles river.

3 The SLOOP System Architecture

The structure of the SLOOP retrieval system is shown in Figure 1. SLOOP is comprised of a Data Exchange & Interaction Server (DEI) and an Image Processing Engine (IPE). The DEI implements the user interface/database (see Figure 5) as a web application running on a GlassFish Server with Postgres binding [16]. The IPE contains segmentation, illumination correction, rectification, matching, relevance feedback, and crowdsourcing tools. It is run as a native MATLAB/Octave server, and can therefore incorporate research codes with relative ease. The separation of IT and Vision components allows independent contributions to be easily absorbed. IPE and DEI interact through a database and, together, they implement a workflow that each image undergoes in its journey from being a photograph to becoming an identity. The end-result is an annotated table is made available to the biologist for subsequent analysis.

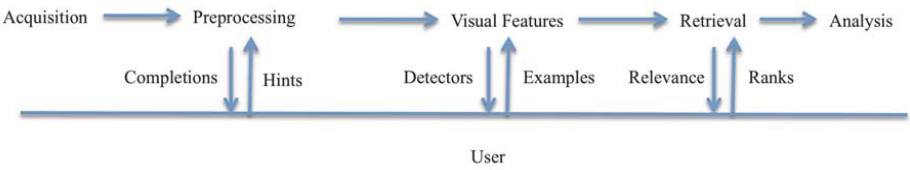


Fig. 2. A SLOOP workflow is an interaction between the users and the system and includes preprocessing, feature extraction, relevance judgement and relevance feedback

The steps involved in a workflow (see Figure 2) typically involve uploading images and metadata, preprocessing images to correct for illumination and geometry, extracting features, producing ranked retrievals, incorporating user judgements, and iterating using relevance feedback. The output of the system is a table with identities associated with each image.

SLOOP enables multiple users to simultaneously work on different aspects of the workflow. IPE and DEI typically operate asynchronously in batch or interactive modes. Some users might be aiding preprocessing, others verification of matches while the IPE might be running rectification and matching jobs on a cloud computing system.

Naturally, synchronization of the system state (tables) becomes necessary for certain tasks. For example, as multiple users verify images, the closure operation linking all matching pairs into a clique through multiple, possibly contradictory, relevance judgements is necessary. SLOOP synchronizes the collection's state through a "Nightly Build" process that locks users out before performing closure operations (among others), thus preventing deadlocks or conflicts from developing. At the end of each Nightly Build, the SLOOP system is completely consistent, all cohorts share the same identity, new identities have been formed where needed and identities are merged where required. At any time after the latest Nightly Build, the user can lock SLOOP to control the quality of the identity tables and unlock it with updated information. In the remainder of this section, we describe SLOOP's methods along a typical workflow.

3.1 Preprocessing Methods

Images are typically preprocessed to prepare them for feature extraction and matching. Although the preprocessing steps vary between species, segmentation, illumination correction, image rectification and feature extraction are often reused tasks. To facilitate preprocessing, SLOOP contains a Mean-Shift based segmentation algorithm. An additional algorithm is being deployed that uses a combination of SVM and graph-cuts on color-texture features. Rectification includes a spline-based algorithm. For illumination correction, SLOOP includes a variety of global correction techniques, as well as an example-based specularity removal method [12]. Once images are preprocessed, features are extracted. Multiscale patch features, local features including invariant feature histograms [1, 2] SIFT features [13], and point feature extraction methods are most commonly used.

A distinguishing aspect of preprocessing in SLOOP is that it is interactive. In contrast to the traditional view that image processing algorithms need to be fully automated, SLOOP takes an interactive approach where users seed or correct first guess solutions of the algorithms. In this way, SLOOP is designed to maximally reduce human effort but maintain performance. As an example, we show in Figure 3 the specularity removal algorithm within SLOOP [12]. In this approach, the user marks specular and normal regions. Based on this input, SLOOP replaces hypothesized specular spots with information from normal regions to seamlessly in-fill —a task that is otherwise exceedingly difficult [12].

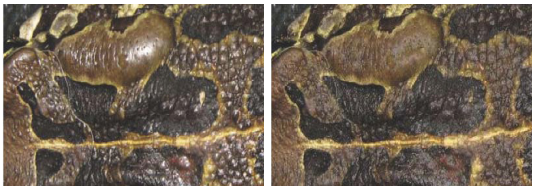


Fig. 3. Interactive example-based specularity removal

3.2 Matching Techniques

SLOOP has appearance-based and geometry-based matching methods, including additions that combine the two. There are four appearance-based methods within SLOOP: multiscale-PCA (MSPCA) [3] on patches and their regularized spatial derivatives, local-feature methods including local-feature histogram [1, 2], SIFT [13], and a new promising method called nonparametric scale-cascaded alignment [4] for deformation invariant matching. In geometry based methods, SLOOP implements an iterative correspondence and affine alignment on point features. These techniques are discussed in more detail in the context of their applications. A common element of all of these methods is that they benefit from interaction in the form of relevance feedback.

3.3 Relevance Feedback

Once matching is complete and ranked hypotheses are produced for photographs in a catalog, the user must verify them. Although the number of images a user must review changes from species to species, typically about $10 - 20$ images at a time are shown. The user selects the matching images (non-matching ones are marked implicitly) and this information is incorporated.

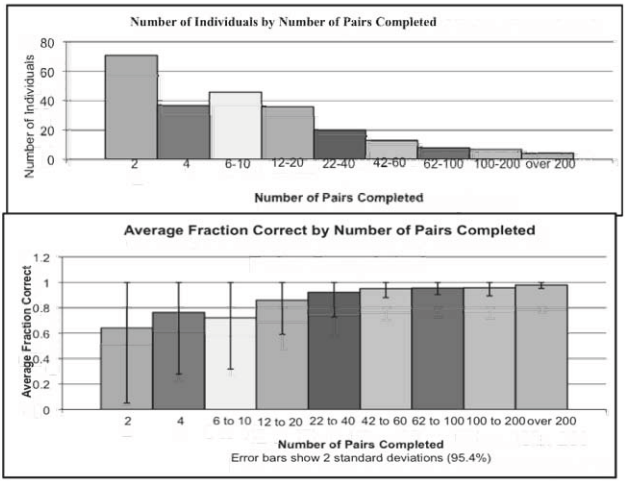


Fig. 4. Crowdsourcing provides sufficient quality for large-scale relevance feedback

In addition to logging the verification and naively presenting the next set of ranked retrievals, SLOOP uses the cohorts identified by the user to iteratively improve similarity judgements in two ways: a) the best score from the cohort group is used to rerank images and, b) the population-based prior is replaced by a cohort-based covariance in MSPCA and local feature histogram methods [1]. Relevance feedback turns out to be very useful. It isdemonstrated in the next section.

Crowdsourcing and Social Media: The efficient indexing of large image sets demands a number of rapidly available relevance judgements. Because verification entails ordinary matching skills, crowdsourcing is one way to gather this information. In one experiment, three pairs of images are presented to a crowd-user in random order: a positive control pair of a known match, a negative control pair of a known non-match, and the experimental pair of unknown status. We accept the user's judgement on the *unknown* pair when, akin to ReCAPTCHA, the control performance is perfect on the *known* pair, which is also the condition for payment. The responses are used to update the ranking in the manner described previously.

In Figure 4, we show the results of a crowdsourcing experiment with Mechanical Turk where a HIT is the task as defined above for the marbled salamander.

In a matter of days, we were able to gather hundreds of workers. The same effort took on the order of a year in grant research, and then there were only a handful. We paid the users five to six cents for each verification task. What we found was a natural selection mechanism; the combination of financial incentive and testing produced a situation where users who were not good at matching tried it only a few times before moving on to a different task, and those who were good kept working. One worker in particular, produced 99.96% recognition rate and answered over 1,000 tests in one day! By the time someone performs 40 known-pairs (twenty tests) of comparisons, statistically their recall is likely to be 95% on average (see Figure 4). The number of people passing this barrier is about a third of the total population, which is very promising.

4 Application to Individual Identification

The Earth Signals and Systems Group's SLOOP Research Program is currently engaged in developing individual identification algorithms for several species. These include *Oligosoma otagense* (Otago Skink, 8,900 photos/900 individuals), *Oligosoma grande* (Grand Skink, 21,700/2,500), *Naultinus gemmeus* (Jewelled Gecko, 10,000/1,600), *Ambystoma opacum* (Marbled Salamander, 10,000/2,000), and *Rhincodon typus* (Whale Shark, 35,000/3,000). Additional work has been done on Fowler's Toad and is planned for Archey's Frog, Humpback Whales, Giant Manta Ray, Hector's Dolphins, Southern Right Whales and Scree Skinks. Here, we discuss the application of SLOOP to three species. Each uses a different matching technique with the potential for advances to influence methods for all species.

4.1 Grand and Otago Skinks

The SLOOP Skink systems (SLOOP-SK) incorporate photographs of the left and of right sides of the animals. These are grouped into capture events which may include photographs of one or both sides of an individual animal. In the Otago Skink dataset, there are currently approximately 4,000 captures with both left and right views and approximately 900 captures with only one view. Of all captures with images, approximately 900 individuals have been identified. In the Grand Skink dataset, there are currently approximately 10,100 captures with both left and right views, approximately 1,500 captures with only one view, and approximately 2,500 individual animals.

For each photograph, a worker verifies the image quality. A worker then marks four key points on the image (see Figure 5). These key points are used to define patches between the nostril and eye, between the eye and ear, and between the ear and shoulder. The patches are only approximately rigid but are assumed as such and are normalized in orientation and scale. A capture may have up to six patches, three for each side. SIFT features [13] are extracted from the patches and corresponding patches on the same side of the animal are compared against each other. The maximum score for any patch is used for ranking, removing the

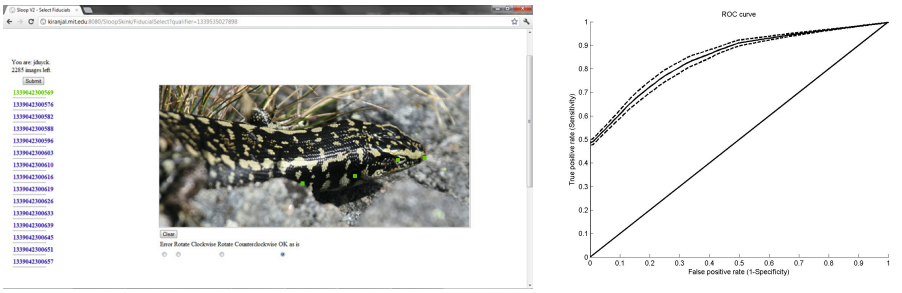


Fig. 5. Otago Skink Dataset with DEI interface showing key points (left) and ROC curve (right)

effect of low-scoring patches. For example, an animal may be partially behind another object, giving a patch an irregularly low score even for photographs of the same individual. After each capture has a ranking for every other capture, workers verify highly-ranked pairs of captures.

The above ROC curve (Figure 5) was found using 1,000 captures (pairs of left and right photographs of an animal) from the Otago Skink dataset. The following groups of animals were used: 282 animals with one capture, 108 animals with two captures, 58 with three captures, 34 with four captures, 17 with five captures, six with six captures, three with seven captures, four with eight captures and two with nine captures. The results indicate that this approach is promising. Additional improvements are possible in terms of rank aggregation, relevance feedback and deformable matching.

4.2 Whale Shark

One application of the SLOOP system is to the whale shark species. The dataset used is a subset of the full data (14,500 images/3,000 individuals). It consists of underwater images of a spot patterning behind the gills of the whale shark on one or both sides. Coordinates of the spots have been specified by users (see Figure 6), and the identification algorithm matches pairs of coordinate sets rather than the images themselves.

The coordinate sets are first normalized for scaling and mean-value. The algorithm then iteratively calculates a correspondence between the sets of points and aligns the points based on this correspondence. In the first iteration, the correspondence is calculated using doubly-stochastic normalization [14] and the points are aligned by a translation. In the second, correspondence calculation is followed by an affine transform calculated with the RANSAC algorithm [15]. The final aligned point sets are scored by determining the area between two CDF curves: the cumulative distribution of distances between corresponding points and that of an ideal perfect match. Thus, lower scores indicate closer matches.

The results of this basic matching algorithm are shown the ROC curves above (see Figure 6). The algorithm was run on seven randomly chosen individuals

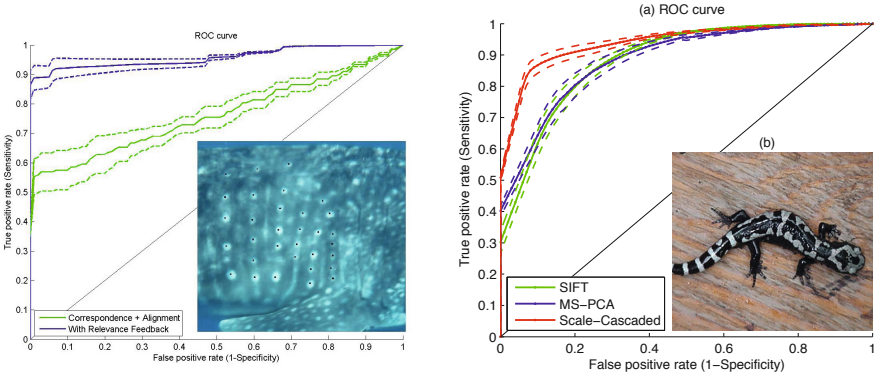


Fig. 6. ROC curves for Whale Shark (left) and Marbled Salamander datasets (right)

each with 5-10 matching images in the database, totaling 56 coordinate sets. The ranks were calculated across all 5,200 or 9,300 photos from the database of left or right side photos respectively. Twenty of these photos and their rankings were randomly chosen to calculate the ROC. To incorporate relevance feedback into the algorithm, the user identifies matches from the 10 best ranking photos, and the ranks are recalculated by taking the minimum score across the new cohort of matching individuals. The user is then shown the 10 new top-ranked photos. This process is repeated until no new matches are identified in the top 10 photos or until the user has viewed a maximum of 50 photos per query. The second ROC curve in Figure 6 reports the improvement achieved through this relevance feedback method. The results show that even for a simple matching procedure, relevance feedback is a powerful addition. Although the improvements to ROC are remarkable, we anticipate adding additional iterations to deal with local deformations in a cascaded framework.

4.3 Marbled Salamander

The marbled salamander (10,000 images/2,000 individuals) was the first species for which SLOOP was developed and applied to [3]. The earliest technique used a multiscale local feature histogram method [1, 2], and a later technique used multiscale PCA (MSPCA) with multiscale Gaussian derivative filter responses on rectified images of animals. MSPCA has also been applied to Fowler's Toad and experimentally to Skinks. It is comparable in performance to SIFT (see Figure 6).

Although the use of rectified images in MSPCA provides some robustness to pose variations, for highly deformable bodies (e.g. salamanders) there is still a substantial amount of local, nonlinear deformation that cannot easily be rectified in advance. Some kind of deformable matching is needed. We developed a scale-cascaded alignment (SCA) method [4] for non-parametric deformable matching which shows considerable promise. To see how it works, consider $J(q) = ||A - T \circ q||_R^2 + L(q)$ an objective that is minimized for deformation q between a template image T and target image A, subject to constraints L, where $T \circ q = T(p - q(p))$.

The vector-field q is dense (and defined at every pixel) and can admit a variety of deformations. For completeness, when the norm R is isotropic and $T \circ q$ is linearized and L represents smoothness, we have classic variational optic flow. If T is not linearized, we have Laplace-Beltrami flow. If R is non-isotropic, we have Ravela's Field Alignment [4] method.

It is useful to note that the objective J can produce a highly selective solution (for example, if L only admits translations) or highly invariant ones (e.g., as a Laplace-Beltrami flow). It follows that the more invariant the matcher is, the less it is able to distinguish dissimilar objects. If the matcher is made more selective, only a few classes of deformations and, thus, only a few objects, can be matched. To optimally adapt invariance and selectivity, we developed Scale-Cascaded Alignment (SCA) [4]. SCA parameterizes the deformation as a weighted sum of Gabor bases with the weights representing a power law turbulence spectrum. This representation can be interpreted as the Green's function of a suitable deformation constraint L. In SCA, the solution is obtained in a coarse to fine manner; the lowest wave number motion solution (translation) is found first followed by higher modes. The total deformation solution or deformation budget is thus dispersed into different modes. Deformations in the low and very high wave numbers are ones to which invariance is desired because they usually correspond to large scale motions or noise. Deformations in middle wave numbers are ones to which selectivity or sensitivity is desired because they help distinguish the natural modes from abnormal ones.

We apply this technique on the Marbled Salamander [3, 4] and the result is shown in Figure 6. SCA outperforms MSPCA and SIFT on the salamander problem and we believe that similar performance improvements could be obtained for skinks and other species. A version for point features is also being developed. If the relevance feedback experiments from Whale-Sharks are also included, we believe all species would substantially benefit.

5 Conclusion and Future Work

The MIT SLOOP system is an extensible community image retrieval system with application to large-scale conservation. Its main benefits include independent components in the IT, biology and vision realms, and ease of incorporation of new research methods including a number of tools and new algorithms developed by our group. SLOOP is being deployed on multiple species, with operational use in two. In these developments we realize the potential for hybrid systems that optimally utilize human interaction and machine skill to deliver high performance recognition systems.

We are developing virtual machines that would simplify future SLOOP deployments. We are investigating rank aggregation from multiple matching methods and a scale-cascaded procedure for iterative correspondence and alignment in the flavor of SCA. We are incorporating graph-based and example-based methods in preprocessing. We invite motivated vision researchers to join SLOOP in an Earth Vision endeavor to develop and apply vision and learning tools for effective stewardship of our Earth System.

References

1. Ravela, S.: On Multi-scale Differential Features and their Representations for Recognition and Retrieval. PhD thesis, University of Massachusetts at Amherst (2002)
2. Ravela, S., Gamble, L.: On recognizing individual salamanders. In: Proc. Asian Conference on Computer Vision (ACCV 2004), vol. 2, pp. 741–747 (2004)
3. Gamble, L., Ravela, S., McGarigal, K.: Multi-scale features for identifying individuals in large biological databases: an application of pattern recognition technology to the marbled salamander ambystoma opacum. Journal of Applied Ecology 45(1), 170–180 (2008)
4. Yang, C., Ravela, S.: Spectral control of viscous alignment for deformation invariant image matching. In: Proceedings of International Conference on Computer Vision, vol. 1, pp. 1303–1310 (2009)
5. Staff: Program will identify skinks. Otago Daily Times (June 2012), `http://www.odt.co.nz/news/dunedin/215070/program-will-identify-skinks`
6. Kumar, N., Belhumeur, P.N., Biswas, A., Jacobs, D.W., Kress, W.J., Lopez, I.C., Soares, J.V.B.: Leafsnap: A computer vision system for automatic plant species identification. In: Fitzgibbon, A., Lazebnik, S., Perona, P., Sato, Y., Schmid, C. (eds.) ECCV 2012, Part II. LNCS, vol. 7573, pp. 502–516. Springer, Heidelberg (2012)
7. Mizroch, S., Beard, J., Lynde, M.: Individual recognition of cetaceans: Use of photo-identification and other techniques to estimate population parameters. In: Hammond, P.S., Mizroch, S.A. (eds.) Computer assisted photo-identification of humpback whales. 12. International Whaling Commission, Cambridge, UK, pp. 63–70 (1990)
8. Araabi, B., Kehtarnavaz, N., McKinney, T., Hillman, G., Wursig, B.: A string matching computer-assisted system for dolphin photoidentification. Annals of Biomedical Engineering 28, 1269–1279 (2000)
9. Arzoumanian, Z., Holmberg, J., Norman, B.: An astronomical pattern-matching algorithm for computer-aided identification of whale sharks rhincodon typus. Journal of Applied Ecology 42, 999–1011 (2005)
10. Hiby, L., Lovell, P.: A note on an automated system for matching the callosity patterns on aerial photographs of southern right whales. Journal of Cetacean Research and Management 2, 291–295 (2001)
11. Kelly, M.: Computer-aided photograph matching in studies using individual identification: an example from serengeti cheetahs. Journal of Mammalogy 82, 440–449 (2001)
12. Runge, J.: Reducing spectral reflections through image inpainting. Master's thesis, Massachusetts Institute of Technology (2009)
13. Lowe, D.: Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
14. Rangarajan, A., Chui, H., Mjolsness, E., Pappu, S., Davachi, L., Goldman-Rakic, P.S., Duncan, J.S.: A robust point matching algorithm for autoradiograph alignment. Medical Image Analysis 1 (1997)
15. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision, 2nd edn. Cambridge University Press (2004) ISBN: 0521540518
16. Ravela, S., Duyck, J., Yang, C., Gamble, L.: Sloop system web (June 2009), `http://tinyurl.com/mitsloop`

A Fuzzy Scheme for Gait Cycle Phase Detection Oriented to Medical Diagnosis

Mario I. Chacon-Murguia, Omar Arias-Enriquez, and Rafael Sandoval-Rodriguez

Visual Perception Applications on Robotic Lab, Chihuahua Institute of Technology, Mexico
{mchacon,oarias,rsandova}@itchihuahua.edu.mx

Abstract. Gait cycle phase detection provides useful information to diagnose possible problems on walking. The work reported here proposes the analysis of gait kinematic signals, extracted from videos, through fuzzy logic to automatically determine the different phases in the human gait cycle. The function of the fuzzy system is to detect the gait phases, loading response, mid-stance, terminal stance, pre-swing, initial swing, mid-swing, and terminal swing, using 2D information from a sagittal plane. The system was tested with normal and non-normal gait cycles. Experimental findings proved that the fuzzy detection system is able to correctly locate the phases using only 2D information. The maximum phase timing shift error generated was 2%. Thus, it may be concluded that the proposed system can be used to analyses gait kinematic and detect gait phases in normal cycle and absences of them in non-normal cycles. This information can be considered for gait anomaly detection and therapeutic purposes.

Keywords: gait phase analysis, fuzzy systems, video segmentation.

1 Introduction

Nowadays, it is well known that vision systems have increased their contribution in areas related to human motion analysis. One of these areas is human gait analysis which is related to the style or characteristics involved in a person's walking [1]. Gait analysis has proved to be relevant to several fields, including biomechanics, robotics, sport analysis, rehabilitation engineering, etc. [2-5]. In the medical field gait analysis has been oriented to medical diagnosis where different types of application have been developed [6]. A specific area in medical application is human movement analysis, where systems are designed to perceive and analyze human movement of upper and lower limbs for the purpose of providing diagnosis and/or therapy [7]. Gait analysis is a paramount aid for therapist because they can analyze different gait pathologies and thus determine appropriate therapies for their patients [8]. These pathologies may be a consequence of spinal cord lesion, stroke, multiple sclerosis, rheumatoid arthritis etc.

Currently, there exist different systems, with advantages and disadvantages, which perform gait dynamic or kinematic quantifications. Dynamic evaluations are achieved through accelerometers, force platforms, etc. The kinematic quantification is done by electrogoniometers and 2D and 3D vision systems [9],[10]. However, not many are related to detect gait phase detection oriented to medical diagnosis. The aim of this paper is to present the development of a fuzzy system to automatically detect gait

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 20–29, 2013.

phases based on the analysis of information provided by a non-invasive vision system [11]. There are some works related to this kind of systems like the one reported in [12] which uses signal from the sagittal plane like, hip, knee and ankle. Another work [13] rely on 3D information. In our work it is proved that a 2D system can be used instead a 3D, and that the input variables generated with the 2D system are suitable for gait phase detection. The information used in the proposed system corresponds to the kinematic of the sagittal plane of the knee, hip, and variables time and stage.

The organization of the paper is structured as follows. Section 2 provides the definition of gait cycle and its phases. The development of the proposed system is explained in Section 3. Experimental tests of the system are presented in Section 4 and the results and conclusions are commented in section 5.

2 Gait Cycle Definition

Human walking is a locomotion method which involves the use of lower limbs allowing the displacement of the body under a stable state. In this process, one lower limb serves as the support, meanwhile the other one provides propulsion. The cycle of a human gait is divided in two main periods; the stance which takes approximately 60% of the cycle and the swing that covers the remaining time. In turn, the stance period is divided into the phases: Loading response, *LR*, mid-stance, MS_s, terminal stance, TS_s, and pre-swing, PS_w. The swing period is composed of: Initial swing, *IS*, mid-swing, MS_w, and terminal swing, TS_w. Table 1 shows the different gait phases and their corresponding timing and cycle percentages [1].

Table 1. Normal gait phase timing

Phase number	Phase	Percentage of phase occurrence in full cycle
1	Loading response	0-10%
2	Mid-stance	10-30%
3	Terminal stance	30-50%
4	Pre-swing	50-60%
5	Initial swing	60-73%
6	Mid-swing	73-87%
7	Terminal swing	87-100%

3 Fuzzy System Scheme for Gait Phase Detection

The inputs to the system correspond to sagittal information related to knee, hip, time and an extra input called stage. The variable time is related to the gait cycle percentage. The information of these variables is acquired by an image processing method described in [14] which is a method to obtain this information without using artificial marks. The system can be formalized as follows. Given kinematic signals of the knee and hip, $x_k(n)$, $x_h(n)$, derive two time variables $x_T(n)$ and $x_S(n)$ to design the mapping

$$F:\{x_k(n),x_h(n),x_T(n),x_S(n)\}\Rightarrow P \tag{1}$$

where $x_k(n)$, $x_h(n)$, $x_T(n)$, $x_s(n)$ and P will be fuzzified and will have the next fuzzy values

$$x_k(n) = \{ x_{kLow}, x_{kHigh} \} \tag{2}$$

$$x_h(n) = \{ x_{hLow}, x_{hMedium}, x_{hHigh} \} \tag{3}$$

$$x_T(n) = \{ x_{TLow}, x_{TMedium}, x_{THigh} \} \tag{4}$$

$$x_S(n) = \{ x_{SLR}, x_{SMS}, x_{SSS} \} \tag{5}$$

$$P = \{P_{LR}, P_{MSS}, P_{TSS}, P_{Sw}, P_{ISw}, P_{MSw}, P_{TSw} \} \tag{6}$$

The mapping F is accomplished by a Mamdani type fuzzy system through the implication relation

$$R\left(\{x_k(n), x_h(n), x_T(n), x_S(n)\}, P\right) \tag{7}$$

3.1 Input and Output Variables and Their Fuzzy Definition

The knee information is obtained from an average knee kinematic signal shown in Figure 1a. Its membership functions represents the different intervals in the y axis. The trajectory can be divided into two intervals; low $x_{k\ Low}$, and high x_{kHigh}. Figure 1b corresponds to an average hip movement and it can be divided in three intervals, $x_{h\ Low}$, $x_{h\ Medium}$, $x_{h\ High}$. The intervals for the input variable time are also low, medium and high as shown in Figure 2. The membership functions for the input variable stage are not determined based on intervals because it is used to distinguish the phases loading response and initial swing and between mid-stance and terminal swing. They are defined based on phase occurrence during the gait cycle.

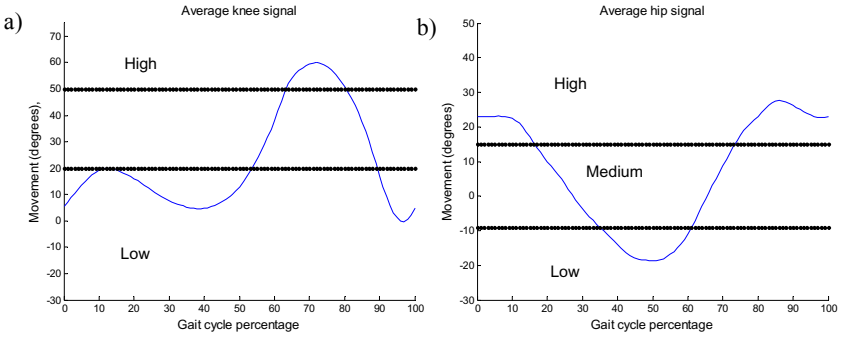


Fig. 1. a) Grade movement intervals of an average knee signal. b) Grade movement intervals of an average hip signal.

It is noteworthy that the inputs to the fuzzy systems need to be normalized, where zero correspond to the minimum value and 1 to the maximum. The membership functions, MFs, of the input variables are shown in Figure 3. The function parameters, mean μ and variance σ are as follow. For the hip input the gaussians are G_{Low}(0, 0.175), G_{Medium}(0.5, 0.175), G_{High}(1, 0.175). For the knee the MFs are similar except the value medium is not used. The gaussian parameters for the input time are G_{Low}(0, 0.256), G_{Medium}(0.5, 0.175) and G_{High}(1, 0.175). As commented before the

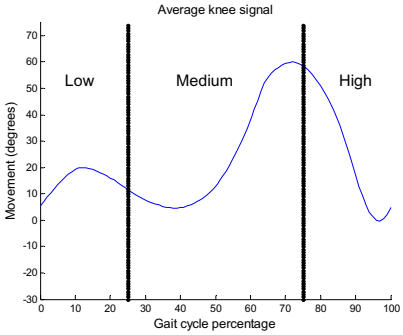


Fig. 2. Time intervals of an average knee signal

input stage is incorporated to the system in order to distinguish between the phases loading response and initial swing and between mid-stance and terminal stance. This is because some inference rules are fired for similar conditions. Thereby, the variable stage allows the differentiation among those fired rules so the correct output is obtained from the system. The stage variable is in fact other time variable with three possible fuzzy values loading response, x_{LR}, mid-stance x_{MSs}, and swing – stance *SS*. The function x_{SLR} is used to differentiate between loading response and mid-swing and it is represented through a *Z* function with parameters $Z_{LR}(0.1, 0.11)$. The function x_{SMS} , $G_{MS}(0.208, 0.05)$ has the purpose to distinguish between mid-stance and terminal stance. Finally the function x_{SSS}, represented by a *S* function $S(0.405, 0.632)$, is used to resolve between the swing and stance phases.

The gait phases, defined in Section 2, correspond to the values of the output variable Phase of the system. These values are represented by triangular membership functions with the following parameters and illustrated in Figure 4. $P_{LR}(0,1,2)$, $P_{MSs}(1,2,3)$, $P_{TSs}(,2,3,4)$, $P_{PSw}(3,4,5)$, $P_{IS}(4,5,6)$, $P_{MSw}(5,6,7)$, $P_{TSw}(6,7,8)$.

3.2 Fuzzy System Rules for Phase Detection

The fuzzy system designed is a Mamdani system defined by rules

$$R_i : if\ x_1\ is\ A_{i1}\ \mathbf{AND}, \ldots, x_{ni}\ is\ A_{in}\ THEN\ \mathrm{y}_i\ is\ B$$

The defuzzification scheme used was the mean of maximum. The rules are determined by an intuitive reasoning. That is, the output is just the consequence of a specific condition of the lower body section. For example, $Rule_1$ corresponds to the phase loading response, *LR*, and it is derived by knowing that in a normal gait cycle the position of the hip is approximately 20 degrees. If we locate this value for the average hip movement, Figure 1b, we found that correspond to a value of *High*. In the case of the knee during the same phase, it has a low value for the intervals defined in Figure 1a.

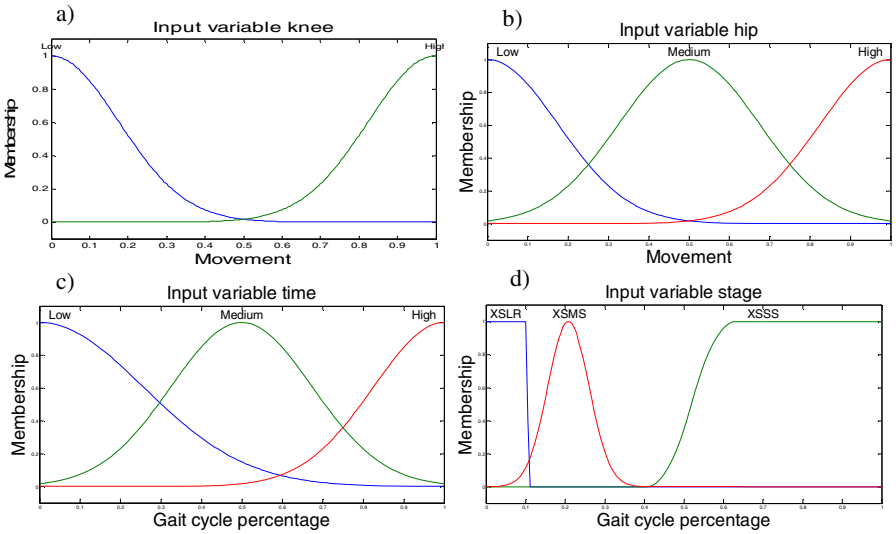


Fig. 3. Membership function definition for input variables. a) Knee, b) Hip, c) Time, d) Stage.

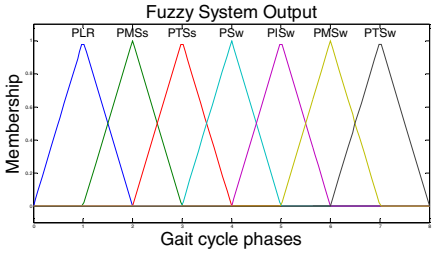


Fig. 4. Output membership function definition

The value of variable stage is located in the interval *LR*, meanwhile the variable time is located in the interval low. Inference using this information yields the system output loading response, P_{LR}. The same methodology was used to determine the other rules expressed next.

R1. **If** *Hip* is x_{hHigh} **AND** *Knee is* x_{kLow} **AND** *Stage is* x_{SLR} **AND** *Time is* x_{TLow} **Then** *Phase is* P_{LR}

R2. **If** *Hip* is *Not* x_{hHigh} **AND** *Knee is* x_{kLow} **AND** *Stage is* x_{SMS} **AND** *Time is* x_{TLow} **Then** *Phase is* P_{MSS}

R3. **If** *Hip* is $x_{hMedium}$ **AND** *Knee is* x_{kLow} **AND** *Stage is* x_{SMS} **AND** *Time is* x_{TLow} **Then** *Phase is* P_{MSS}

R4. **If** *Hip* is x_{hLow} **AND** *Knee is* x_{kLow} **AND** *Time is* $x_{TMedium}$ **Then** *Phase is* P_{TSS}

R5. **If** *Hip* is x_{hLow} **AND** *Knee is Not* x_{kLow} **AND** *Stage is* x_{SSS} **AND** *Time is* $x_{TMedium}$ **Then** *Phase is* P_{Sw}

R6. **If** *Hip* is $x_{hMedium}$ **AND** *Knee is Not* x_{kLow} **AND** *Stage is* x_{SSS} **AND** *Time is* $x_{TMedium}$ **Then** *Phase is* P_{ISw}

R7. **If** *Hip* is x_{hHigh} **AND** *Knee is* x_{kHigh} **AND** *Stage is* x_{SSS} **AND** *Time is* x_{THigh} **Then** *Phase is* P_{MSw}

R8. **If** *Hip* is x_{hHigh} **AND** *Knee is* x_{kLow} **AND** *Stage is* x_{SSS} **AND** *Time is* x_{THigh} **Then** *Phase is* P_{TSw}

4 Experimental Results

4.1 Experimental Data

The system was tested with two set of data representing two cases: normal gait and non-normal gait. The normal data is to verify that the system is able to detect the gait phases and the non-normal data to prove that the system does not detected the phases because they do not exist in these cases or they are misplaced. The different kinematic cases analyzed with the fuzzy system are as follows. Case 1 represents the kinematic of a patient with amputation below the knee. An average gait cycle is included in the cases 2, 4, 7, and 8. Case 7 includes average values in the upper bound and case 8 in the lower case. Case 3 is a person that presents spinal cord injury. Cases 5 and 6 correspond to patients with cerebral palsy.

Findings of the experiments using the previous cases indicate that the system was able to correctly detect the gait phases in the normal cases, meanwhile the system did not report some of the phases in the non-normal cases. Specific detail of each experiment is provided next.

Case1. In the case of the patient with amputation the kinematic tends to present a signal similar to a normal gait cycle, however the kinematic values are not always in the range of normal average values, as shown in Figure 5a. This observation can be visualized in Figure 5b, which presents the information of the occurrence of the gait phases. In a normal gait cycle the two traces must completely overlap. It can be noticed that the system detects phase 5, initial swing in the 84% of the gait cycle, when in this percentage the phase mid-swing should be detected. During the other gait cycle percentage the systems detects a normal behavior with respect phase occurrence and a small difference on the detection of phases 3, 4 and 7.

Case 2. First normal gait cycle. The signals corresponding to the knee and hip of this normal case are shown in Figure 6a. The result of the fuzzy system is illustrated in Figure 6b. The loading response phase is correctly detected in the percentage 0% to 10%, as well as the mid-stance, 11% to 30%. The terminal stance is located in the interval 31% to 49% and the pre-swing in the 50% to 59%, which indicates a normal behavior because the average occurrence of this phase is between 50% - 62%. The initial swing is found in the range 63%-74%, the mid-swing at 75% to 87% and the terminal swing at 88% - 100%. Comparing these results with the data in Table 1, normal timing, it can be observed that the system reports very alike timing with just a few but not significant differences.

Case 3. Spinal injury case. The results of this case are presented in Figure 7a and 7b. It can be noticed from these figures that the system detected an anomalous behavior because the phase occurrence differences are large, in some occasions up to 6 percentage grades. Besides, an initial swing is detected in the terminal swing phase. Changes as the previously mentioned indicate a non-normal gait cycle.

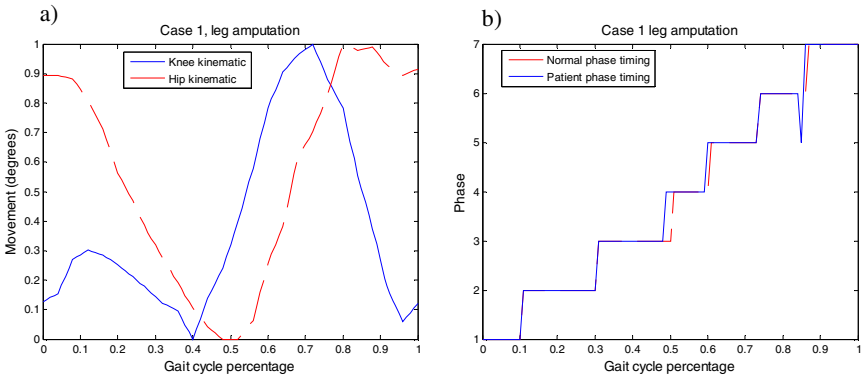


Fig. 5. Case 1 patient with leg amputation. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

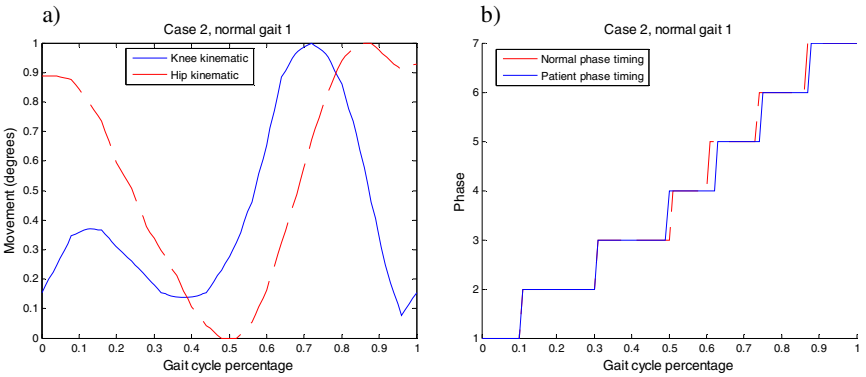


Fig. 6. Case 2 normal gait 1. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

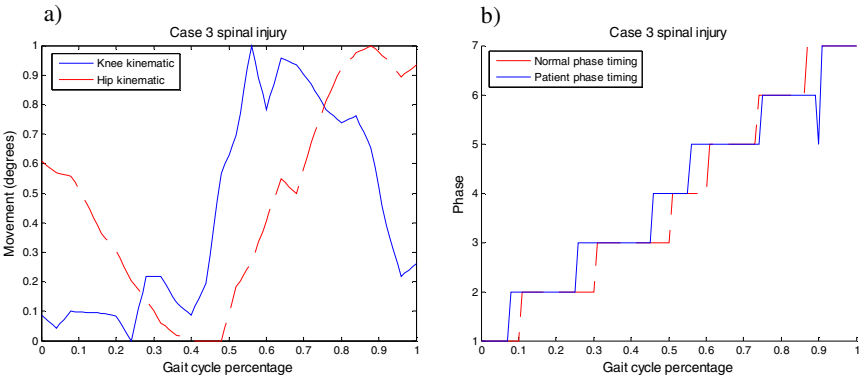


Fig. 7. Case 3 spinal injury. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

Case 4. Second normal gait cycle. This case and the cases 7 and 8 are incorporated into the experiments to warranty the robustness of the system to adequately detect normal gait cycle with normal variations. This case is not illustrated due to space constraints. The first three cycle phases, loading response, mid-stance and terminal stance are detected on time with the average gait cycle phases. Pre-swing, the four phase, is detected at 51% - 62%, the initial swing at 63%- 74%, mid-swing in the interval 75% - 87% and finally the terminal swing in 88% to 100%. The fuzzy system results present very similar time phase detection as for a normal gait cycle, being 2% the maximum difference.

Case 5. Cerebral palsy 1, see Figure 8. The phase detection timing generated by the system presents very high deviation as well as an inadequate sequence from the normal cycle. For example, the loading response phase is detected only during the 0% to 2% interval, when it must be identified from 0% to 10%. Other observation is with regard the terminal stance phase, where an initial swing is reported.

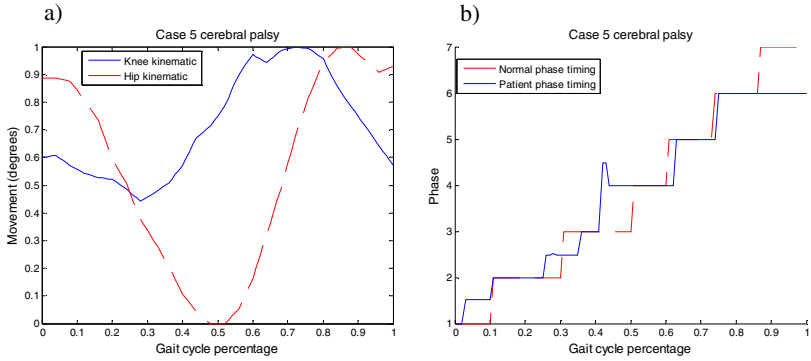


Fig. 8. Case 5 cerebral palsy 1. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

Case 6. Cerebral palsy 2. The kinematic and phase detection results are reported in Figure 9. As in the first cerebral palsy case, it can be observed how phase occurrence does not follow the normal timing. The loading response phase, as in the previous case is only detected during the 0% to 2%, besides and erroneous detection of a pre-swing phase during mid-stance.

Case 7. Normal gait cycle upper bound. This case corresponds to high average normal kinematic values. The kinematic is illustrated in Figure 10a and the phase detection results in Figure 10b. Results show that the first 3 phases are correctly detected following a normal phase timing. The fourth phase, pre-swing, was detected in 50% - 62% which is considered correct. The initial swing is found in the 63% to 74%, mid-swing at 75% - 87% and the terminal swing at 88% - 100%, thus having a maximum deviation of 1% with respect the correct timing.

Case 8. Normal gait cycle lower bound. This case covers the lower kinematic values for a normal gait cycle. This case is not illustrated due to space constraints. The phase loading response, mid-stance and terminal stance are detected at the correct timing. The other phases, pre-swing, initial swing, mid-swing and terminal swing

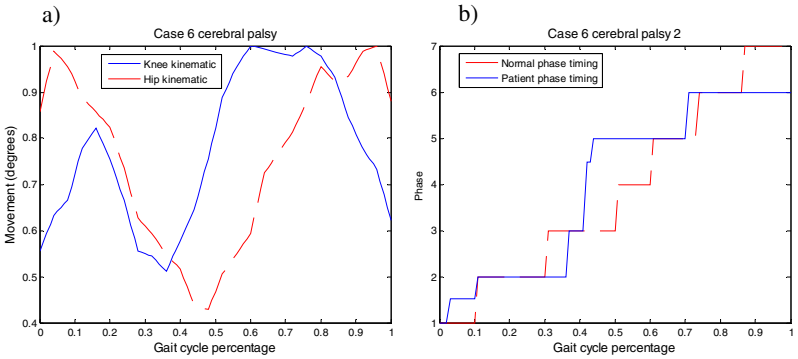


Fig. 9. Case 6 Cerebral palsy 2. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

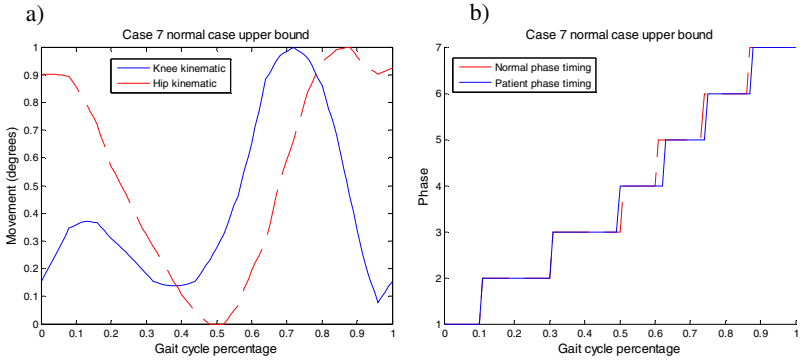


Fig. 10. Case 7 normal case upper bound. a) Knee and hip kinematic. b) Phase detection, normal vs. system timing.

were located at 51% - 62%, 63% - 74% , 75% - 86% and 87% - 100% respectively. Only the phases pre-swing and initial swing present a small time shift. The maximum phase shift was 2%.

5 Results and Conclusions

The reported work proved that a fuzzy gait phase detection system is able to correctly detect the phases using only 2D information. The system was tested with non-normal as well as variation of normal gait cycle cases, in both circumstances the system showed acceptable results on the detection of the gait cycle phases for the normal cases. The maximum timing phase shift error generated was 2% for normal cases. In the non-normal cases the system reported misplaced phases as expected as a result of inadequate knee and hip kinematic due to the physical problems of the patients. Thus, it may be concluded that the proposed fuzzy system can be used to analyses gait kinematic by detecting gait phases in normal cycle and absences of them in non-normal

cycles. This information may be considered for gait anomalies detections as well as therapeutic purposes.

Acknowledgements. The authors thanks to Fondo Mixto de Fomento a la Investigación Científica y Tecnológica CONACYT-Gobierno del Estado de Chihuahua and DGEST, by the support of this research under grant CHIH-2009-C02-125358 and CHI-MCIET-2011-103, respectively.

References

1. Perry, A.J.: Gait Analysis: Normal and Pathological Function. Slack, NJ (1992)
2. Senanayake, B.C.M., Senanayake, S.M.: Computational Intelligent Gait Phase Detection System to Identify Pathological Gait. IEEE Transactions on Information Technology in Biomedicine 14(5), 1173–1179 (2010)
3. Cardenas Maciel, C.S.L., Castillo, O., Aguilar, L.T.: Generation of Walking Periodic Motions for a Biped Robot Via Genetic Algorithms. Appl. Soft Comput. 11(8), 5306–5314 (2011)
4. Cardenas Maciel, D.S.L., Castillo, O., Aguilar, L.T., Castro, J.R.: A TS Fuzzy Logic Controller for Biped Robot Walking Based on Adaptive Network Fuzzy Inference System. In: Proceeding of the IJCNN, pp. 1–8 (2010)
5. Courtney, J., Paor, A.M.: A Monocular Marker-Free Gait Measurement System. IEEE Transactions on Neural Systems and Rehabilitation Engineering 18(4), 453–460 (2010)
6. Kirtley, F.C.: Clinical gayt analysis, http://www.clinicalgaitanalysis.com/
7. Lauer, G.R., Smith, B.T., Betz, R.R.: Application of a Neuro-fuzzy Network for Gait Event Detection Using Electromyography in the Child With Cerebral palsy. IEEE Transactions on Biomedical Engineering 52(9), 1532–1540 (2005)
8. Pappas, O.I., Popovic, M.R., Keller, T., Dietz, V., Morari, M.: A reliable gait phase detection system. IEEE Trans Neural Systems and Rehabilitation Engineering 9(2), 113–125 (2001)
9. Goffredo, I.M., Nixon, M., Carter, J.: 2D Markerless Gait Analysis. In: Proceedings of the 4th European Congress for Medical And Biomedical Engineering (2008)
10. Yuan, Q., Chen, I., Lee, S.P.: SLAC: 3D localization of human based on kinetic human movement capture. In: Proceedings of IEEE Robotics and Automation, pp. 848–853 (2011)
11. Arias-Enriquez, O.: Human Gait Analysis Based on 2D/Kinect Visual Perception and Diagnosis With Fuzzy Logic. Master thesis, Chihuahua Institute of Technology (2012)
12. Senanayake, C., Senanayake, S.M.: Evaluation of gait parameters for gait phase detection during walking. In: IEEE Proceedings of Multisensor Fusion and Integration for Intelligent Systems, Salt Lake City, UT, pp. 127–132 (2010)
13. MacDonald, C., Smith, D., Brower, R., Ceberio, M., Sarkodie-Gyan, T.: Determination of Human Gait Phase Using Fuzzy Inference. In: Proceedings of IEEE Rehabilitation Roboticsonal, pp. 661–665 (2007)
14. Chacon-Murguia, M.I., Sandoval-Rodriguez, R., Arias-Enriquez, O.: Human Gait Feature Extraction Including Kinematic Analysis Toward Robotic Power Assist. International Journal of Advanced Robotic Systems 9(68) (2012)

Virtual Target Formulation for Singularity-Free Visual Control Using the Trifocal Tensor

H.M. Becerra[1], J.B. Hayet[1], and C. Sagüés[2]

[1] Centro de Investigación en Matemáticas (CIMAT),
C.P. 36240, Guanajuato, Gto., Mexico
{hector.becerra,jbhayet}@cimat.mx
[2] Instituto de Investigación en Ingeniería de Aragón,
Universidad de Zaragoza, C/ María de Luna 1, E-50018, Zaragoza, Spain
csagues@unizar.es

Abstract. We present a novel approach for visual control of wheeled mobile robots, extending the existing works that use the trifocal tensor as source for measurements. In our approach, singularities typically encountered in this kind of methods are removed by formulating the control problem based on the trifocal tensor and by using a virtual target vertical translated from the real target. A *single* controller able to regulate the robot pose towards the desired configuration without local minima is designed. Additionally, the proposed approach is valid for perspective cameras as well as catadioptric systems obeying a central camera model. All these contributions are supported by convincing simulations.

Keywords: Visual control, virtual target, trifocal tensor.

1 Introduction

The use of machine vision in robotics is still a challenging task. In this context, the problem of visual control (visual servoing) of robots is an interesting application of computer vision techniques, control theory and robotics. This problem consists in computing suitable velocities to drive a robot to a desired location associated to a target image previously acquired. Feedback is obtained from visual data by comparing current and target images. Visual servoing (VS) schemes are classified as image-based when image data is used directly in the control loop, or position-based, when pose parameters are needed [1]. Classical approaches use image points as visual features, given that they are easily extracted. Advanced approaches use other geometrical primitives (lines, moments, etc.) or geometric constraints to improve robustness of the control scheme [2–4].

Recently, multiple-view geometry have been exploited for the visual control of mobile robots [5–8]. When designed properly, these schemes avoid the local minima problems of the classical schemes where overdetermined solutions are obtained. The homography-based control relies on planar scenes [5, 6], so that, more general constraints like the ones induced by epipolar and trifocal geometries have been preferred. However, the epipolar geometry is ill-conditioned with short

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 30–39, 2013.

baseline. Moreover, controllers based on the epipolar constraint are subject to singularities. This has been tackled by using control schemes that switch between two different controllers when degeneracies and singularities appear [3, 7].

The present work has been inspired by [9], where the authors suggest the use of a virtual target in order to avoid some degeneracies of the essential matrix and singularities of an epipolar visual control. In that work, a virtual target is generated relying on the transfer relations given by the essential matrix. However, the transfer of points can fail for certain configurations, like collinear projection centers [10]. In the work herein, we propose the use of the trifocal tensor (TT), which is known to be more robust and better defined than the epipolar geometry. The transfer relations associated with the TT can be used to transfer points and lines from two source images into a third one without degenerate cases. In the literature, the TT has been exploited for image-based visual servoing but with some limitations related to local minima and switching control policies [4, 8].

The contribution of this paper is the formulation to generate and use a virtual target (virtual image) from the 2D TT in the context of visual control of a wheeled mobile robot. The virtual target provides additional information that avoids the need of switching to a different controller in contrast to [3, 7, 8]. A single controller, free of singularities, is derived in order to show the feasibility of using the virtual target. This controller achieves regulation of the robot pose, i.e., position and orientation are simultaneously driven to their desired values. An additional benefit of the proposed VS scheme is that it is valid for different types of cameras, in particular, those obeying the unified projection model [11].

The paper is organized as follows. Section 2 describes the mathematical modeling of the robot and the vision system. Section 3 details the generation of the virtual target relying on the TT. Section 4 describes the controller design from the TT using the virtual target. Section 5 shows the performance of the proposed approach through realistic simulations and Section 6 summarizes the conclusions.

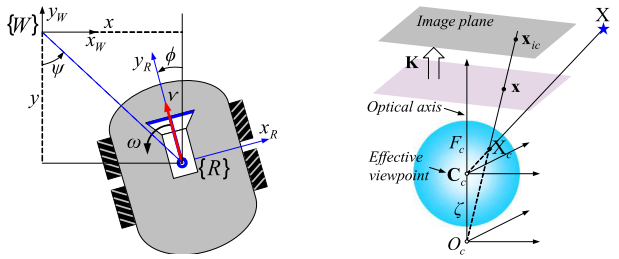
2 Modeling for Visual Control of a Mobile Robot

2.1 Robot Kinematics

Let $\chi = (x, y, \phi)^T$ be the state vector of a differential drive robot shown in Fig. 1(a), where x and y are the robot position coordinates in the plane, and ϕ is its orientation. Assume that a central camera is fixed to the robot in such a way that the robot and camera reference frames coincide. The kinematic model of the camera-robot system expressed in state space can be written as follows:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} -\sin\phi & 0 \\ \cos\phi & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix}, \tag{1}$$

being ν and ω the translational and angular velocities, respectively. In the sequel, the notation $s\phi = \sin\phi$ and $c\phi = \cos\phi$ is used.



(a) (b)

Fig. 1. Representation of the robot model and the camera model. (a) Robot frame definition. (b) Generic camera model of central cameras [11].

2.2 The Trifocal Tensor for Generic Cameras

A desirable feature for a VS scheme is its applicability for different types of cameras, e.g. conventional or omnidirectional. Cameras with wide field of view are preferred to avoid the loss of visual features during motion. Geometric constraints have provided a good strategy to achieve generic control schemes. Hence, we use the generic model of central cameras [11] to exploit the computation of a geometric constraint, like the TT, in the same way for any central vision system.

Let $\mathbf{X}$ be the coordinates of a 3D point. Under the unified projection model, its corresponding point on the unit sphere $\mathbf{X}_c$ can be computed from point $\mathbf{x}$ on the normalized image plane (see Fig. 1(b)) and the sensor parameter ζ as:

$$\mathbf{X}_c = \left(\eta^{-1} + \zeta\right) \bar{\mathbf{x}} \quad , \bar{\mathbf{x}} = \left[\mathbf{x}^T \ \tfrac{1}{1+\zeta\eta}\right]^T \tag{2}$$

where $\eta = \frac{-\gamma-\zeta\left(x^2+y^2\right)}{\zeta^2\left(x^2+y^2\right)-1}$, $\gamma = \sqrt{1+\left(1-\zeta^2\right)\left(x^2+y^2\right)}$. We assume that the camera is calibrated if omnidirectional vision is used, which allows to exploit the representation of the points on the unit sphere. With a conventional camera, calibration is not needed and the TT can be computed from normalized points [10].

The TT encodes the geometry between three views, independently of the scene structure [10]. It has 27 elements (18 independent) and can be expressed using three 3×3 matrices $\mathbf{T} = \{\mathbf{T}_1, \mathbf{T}_2, \mathbf{T}_3\}$. Here, we use points as image features: Consider three corresponding points projected on the unit sphere $\mathbf{p}$, $\mathbf{p}'$ and $\mathbf{p}''$ in three views of a 3D scene, in homogeneous coordinates, i.e. $\mathbf{p} = (p^1, p^2, p^3)^T$. The incidence relation between them is given by

$$[\mathbf{p}']_\times \left(\sum_i p^i \mathbf{T}_i\right) [\mathbf{p}'']_\times = \mathbf{0}_{3\times 3} \tag{3}$$

where $[\mathbf{p}]_\times$ is the common skew symmetric matrix.

Consider images taken from three different *coplanar* locations, i.e., with a camera moving at a fixed distance from the ground. In this case, several tensor elements are zero and only 12 elements are in general non-null. Fig. 2 depicts the upper view of three cameras with global reference frame in the third view,

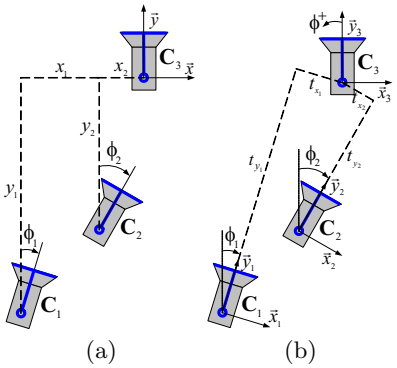


Fig. 2. Geometry between three camera locations in the plane. (a) Absolute locations with respect to a reference frame in $\mathbf{C}_3$. (b) Relative locations.

in such a way that the camera locations are $\mathbf{C}_1 = (x_1, y_1, \phi_1)$, $\mathbf{C}_2 = (x_2, y_2, \phi_2)$ and $\mathbf{C}_3 = (0, 0, 0)$. The TT non-null elements can be analytically deduced [4] as:

$$\begin{aligned}
&T^m_{111} = -t_{x_1}c\phi_2 + t_{x_2}c\phi_1, T^m_{112} = t_{x_1}s\phi_2 + t_{y_2}c\phi_1, T^m_{121} = -t_{y_1}c\phi_2 - t_{x_2}s\phi_1,\\
&T^m_{122} = t_{y_1}s\phi_2 - t_{y_2}s\phi_1, T^m_{211} = -t_{x_1}s\phi_2 + t_{x_2}s\phi_1, T^m_{212} = -t_{x_1}c\phi_2 + t_{y_2}s\phi_1,\\
&T^m_{221} = -t_{y_1}s\phi_2 + t_{x_2}c\phi_1, T^m_{222} = -t_{y_1}c\phi_2 + t_{y_2}c\phi_1,\\
&T^m_{313} = -t_{x_1}, T^m_{323} = -t_{y_1}, T^m_{331} = t_{x_2}, T^m_{332} = t_{y_2}
\end{aligned} \tag{4}$$

where $t_{x_i} = -x_i c\phi_i - y_i s\phi_i$, $t_{y_i} = x_i s\phi_i - y_i c\phi_i$ for $i = 1, 2$ and where the superscript m indicates metric information. The estimated tensor has an unknown scale factor, changing as the robot moves. We set a common scale by normalizing each tensor element as $T_{ijk} = T^e_{ijk}/T_N$, where T^e_{ijk} are the estimated TT elements obtained, T_{ijk} are the normalized elements, and T_N a suitable normalizing factor. We can see from Eq. 4 that T_{313} and T_{323} are constant and non-null, if $\mathbf{C}_1 \neq \mathbf{C}_3$. Hence, any of these two elements can serve as normalizing factor.

3 A Virtual Target from the TT

In the sequel, as described in Fig. 2, $\mathbf{C}_1$, $\mathbf{C}_2(t)$, and $\mathbf{C}_3$ are respectively the initial, current (at time t) and target camera-robot locations. Notice that $\mathbf{C}_1$ and $\mathbf{C}_3$ remain fixed during the motion. The pose regulation problem consists in driving the robot to $\mathbf{C}_2(t) = \mathbf{C}_3$, where the current image observed by the camera (corresponding to $\mathbf{C}_2(t)$) is the same as the previously acquired target image. On the one hand, it is numerically troublesome to estimate the TT if two images are the same and some elements must be discarded for control purposes in that case [4]. On the other hand, the use of the radial TT (first 8 expressions of Eq. 4) has resulted in the need of a few controllers in order to accomplish the pose regulation task [8]. Inspired by [9], we use a TT relative to a new, virtual target location $\bar{\mathbf{C}}_3$. This location is the same as before but *shifted* by a vertical

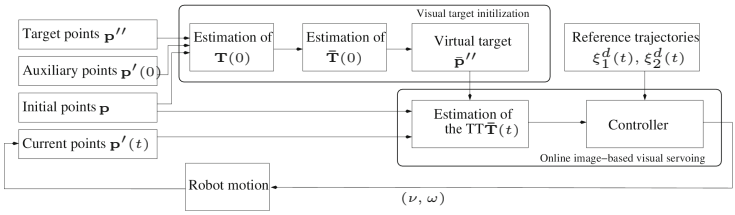


Fig. 3. Overview of the TT-based visual servoing methodology with virtual targets

translation ($t_z > 0$) as shown in Fig. 4. Thus, the *planar* geometry among $\mathbf{C}_1$, $\mathbf{C}_2(t)$ and $\bar{\mathbf{C}}_3$ stays the same as in Fig. 2 and the image at the end of the motion is different to the virtual target image ($\bar{\mathbf{C}}_3$). The benefits of using a virtual target are that any non-null element of the new tensor $\bar{\mathbf{T}}(t)$ (corresponding to views at $\mathbf{C}_1$, $\mathbf{C}_2(t)$, and $\bar{\mathbf{C}}_3$) may be chosen for control purposes, as they are well-defined.

As depicted in Fig. 3, we define points configurations $\mathbf{p}$ (initial) and $\mathbf{p}''$ (target). Then, the robot moves to a second configuration from which the control loop starts ($t = 0$). From this configuration, the points $\mathbf{p}$ have been tracked into points $\mathbf{p}'(0)$, and the TT $\mathbf{T}(0)$ (at time 0) is computed. From section 3.1, we deduce the TT associated to the virtual target, $\bar{\mathbf{T}}(0)$, and from section 3.2, we estimate the virtual target position $\bar{\mathbf{p}}''$ to be used in the control loop. Then, for $t > 0$, the control loop uses the entries of the TT computed from $\mathbf{p}, \bar{\mathbf{p}}''$ and $\mathbf{p}'(t)$ (current image) to drive the robot to the target location (see Section 4).

3.1 From Real TT to Virtual TT

At time 0, the TT, $\bar{\mathbf{T}}(0)$ (relating $\mathbf{p}, \mathbf{p}'(0), \bar{\mathbf{p}}''$) differs from $\mathbf{T}(0)$ (as in Eq. 4, computed from $\mathbf{p}, \mathbf{p}'(0), \mathbf{p}''$) by the following elements:

$$\begin{aligned} &\bar{T}^m_{113}(0) = t_{z_2} c\phi_1, \bar{T}^m_{123}(0) = -t_{z_2} s\phi_1, \bar{T}^m_{131}(0) = -t_{z_1} c\phi_2(0), \bar{T}^m_{132}(0) = t_{z_1} s\phi_2(0), \\ &\bar{T}^m_{213}(0) = t_{z_2} s\phi_1, \bar{T}^m_{223}(0) = t_{z_2} c\phi_1, \bar{T}^m_{231}(0) = -t_{z_1} s\phi_2(0), \bar{T}^m_{232} = -t_{z_1} c\phi_2(0) \end{aligned} \quad (5)$$

where $t_{z_1} = t_{z_2} = t_z$, as the global reference frame is now $\bar{\mathbf{C}}_3$. The distance t_z is arbitrary but we recommend the unity. The angles ϕ_1 and $\phi_2(0)$ (evolving as $\phi_2(t)$ in the control loop) are estimated from $\mathbf{T}(0)$ (Eq. 4) as follows:

$$\phi_1 = \arcsin\left(\frac{T_{323}T_{212} - T_{313}T_{121}}{T_{323}T_{332} + T_{313}T_{331}}\right), \quad \phi_2(0) = \arccos\left(\frac{T_{332}T_{121} + T_{331}T_{212}}{T_{323}T_{332} + T_{313}T_{331}}\right). \quad (6)$$

Hence, by using both Eq. 5 and Eq. 6, we can deduce $\bar{\mathbf{T}}(0)$.

3.2 Generating the Virtual Target

Generating the virtual target implies transferring the points seen at $\mathbf{C}_1$ and $\mathbf{C}_2(0)$ into the view corresponding to $\bar{\mathbf{C}}_3$. If $\mathbf{p}$ and $\mathbf{p}'(0)$ are two such points at

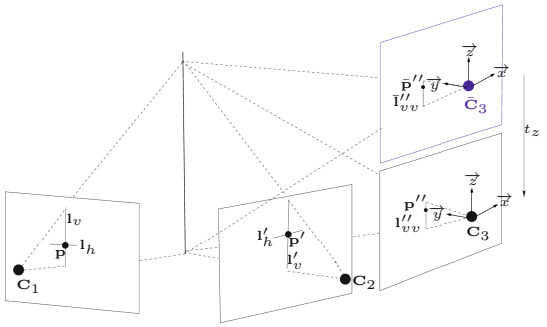


Fig. 4. Generation of a virtual target. From $\mathbf{T}$, associated to $\mathbf{C}_1$, $\mathbf{C}_2$, $\mathbf{C}_3$, and t_z, the tensor $\bar{\mathbf{T}}$ associated to $\mathbf{C}_1$, $\mathbf{C}_2$, $\bar{\mathbf{C}}_3$ is estimated and used for point transfer.

$\mathbf{C}_1$ and $\mathbf{C}_2(0)$, detected independently, then in general, they do not satisfy the epipolar constraint induced by the first two views. Hence, Eq. 3 has no solution.

A simple workaround uses the fact that the TT can give a unique transferred straight line for the third view, given two straight lines equations in the first and second views. As depicted in Fig. 4, consider a pair of horizontal/vertical lines $(\mathbf{l}_h, \mathbf{l}_v)$ through $\mathbf{p}$, and a second pair of lines $(\mathbf{l}'_h, \mathbf{l}'_v)$ through $\mathbf{p}'(0)$. By construction, transferred points $\bar{\mathbf{p}}''$ should belong to any straight line made from pairs $(\mathbf{l}_h, \mathbf{l}'_h), (\mathbf{l}_h, \mathbf{l}'_v), (\mathbf{l}_v, \mathbf{l}'_h), (\mathbf{l}_v, \mathbf{l}'_v)$, each of which being computed with $\bar{\mathbf{T}}(0)$. In Fig. 4, the process is illustrated for the pair $(\mathbf{l}_v, \mathbf{l}'_v)$, which image is given by [10]: $\bar{\mathbf{l}}''_{vv} = (\mathbf{l}_v^T \bar{\mathbf{T}}_1(0)\mathbf{l}'_v, \mathbf{l}_v^T \bar{\mathbf{T}}_2(0)\mathbf{l}'_v, \mathbf{l}_v^T \bar{\mathbf{T}}_3(0)\mathbf{l}'_v)^T$. Each of these straight line pairs leads to a linear constraint on the coordinates of $\bar{\mathbf{p}}''$. Hence, a linear system can be formed to determine the coordinates of $\bar{\mathbf{p}}'' \in \mathbb{R}^3$ in a least square sense $\bar{\mathbf{p}}'' = \min_{\|\bar{\mathbf{q}}''\|=1} \|\mathbf{L}\bar{\mathbf{q}}''\|^2$ where $\mathbf{L}$ is a 4×3 matrix containing the line equations. Then, $\bar{\mathbf{p}}''$ is extracted as the singular vector of $\mathbf{L}$ with the smallest singular value.

4 Visual Control Using the Virtual Target

In the literature, switching control laws have been proposed to solve the pose regulation problem of mobile robots by exploiting geometric constraints [3,7,8]. At least two different controllers with an appropriate switching policy are used in such approaches to deal with degeneracies and singularities. In this section, we present a single controller capable to drive the robot to the target pose (position and orientation). Hereafter, we denote $\mathbf{T} = \bar{\mathbf{T}}(t)$ for clarity.

4.1 Input-Output Linealization

After analysis of the information provided by the TT estimated from the virtual target, we have chosen the following measurements as outputs of our system:

$$\xi_1 = T_{332}, \quad \xi_2 = \frac{T_{132}}{T_{131}}. \tag{7}$$

Note that the first tensor element comes from the "true" trifocal tensor, while the last two elements come from the virtual one. Hereafter, we will denote $x = x_2$, $y = y_2$ and $\phi = \phi_2$, the robot pose in the plane w.r.t. the reference frame attached to $\bar{\mathbf{C}}_3$. The outputs are related to the camera-robot state as follows:

$$\xi_1 = \alpha\left(xs\phi - yc\phi\right), \quad \xi_2 = -\tan\phi$$

where α is an unknown scale factor. An important consideration is that these outputs are valid in the range $|\phi| < \pi/2$, so that, we assume that the initial orientation of the robot accomplish such condition. Notice that both outputs are null if the robot has reached the target location. It can be seen that $\xi_1 = 0$ and $\xi_2 = 0$ imply that $\phi = 0$, $y = 0$ and x is a degree of freedom of the solution, which means that the orientation and longitudinal error are corrected while the lateral error may be different from zero (zero dynamics from the control theory point of view [12]). However, given that ξ_1 is related to the longitudinal position and ξ_2 depends directly on the orientation, the lateral deviation can be corrected with the tracking of an adequate trajectory for ξ_2 as the robot moves forward. It is desired to drive the outputs to zero in a fixed time horizon, which is a trajectory tracking control problem. Let us define the following error functions:

$$e_1 = \xi_1 - \xi_1^d, \qquad e_2 = \xi_2 - \xi_2^d$$

where ξ_1^d and ξ_2^d are smooth desired trajectories with null final value. The tracking problem can be faced by using the input-output linearization technique [12]. It needs the time derivatives of the error functions, given by:

$$\begin{bmatrix} \dot{e}_1 \\ \dot{e}_2 \end{bmatrix} = \begin{bmatrix} -\alpha & -T_{331} \\ 0 & -\frac{T_{131}^2 + T_{132}^2}{T_{131}^2} \end{bmatrix} \begin{bmatrix} \nu \\ \omega \end{bmatrix} + \begin{bmatrix} \dot{\xi}_1^d \\ \dot{\xi}_2^d \end{bmatrix}. \tag{8}$$

This system can be written as $\dot{\mathbf{e}} = \mathbf{J}\mathbf{u} + \dot{\xi}^d$, where $\mathbf{J}$ is the interaction matrix that relates the robot velocities to the rate of change of the visual measurements of Eq. 7. In order to find out adequate robot velocities to track the desired trajectories, the error system must be inverted, which is possible given that $\det(\mathbf{J}) = \alpha\frac{T_{131}^2 + T_{132}^2}{T_{131}^2} = \frac{\alpha}{\cos^2\phi} \neq 0$. Hence, the robot velocities are given by:

$$\begin{bmatrix} \nu \\ \omega \end{bmatrix} = \begin{bmatrix} -\frac{1}{\alpha} & \frac{T_{331}T_{131}^2}{\alpha\left(T_{131}^2 + T_{132}^2\right)} \\ 0 & -\frac{T_{131}^2}{T_{131}^2 + T_{132}^2} \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \end{bmatrix} \tag{9}$$

where v_1, v_2 are auxiliary control inputs that define the error functions convergence. Different options exist to assign the auxiliary controls but their derivation is not in the scope of the paper. In the results section, the super-twisting control technique [13] is used to show the applicability of the controller of Eq. 9.

4.2 Desired Trajectories

The desired trajectories must drive smoothly the outputs of the system from their initial values to zero in a fixed time horizon. Thus, the desired trajectory for ξ_1 is always defined as follows:

$$\xi_1^d = \frac{\xi_1(0)}{2}\left(1+\cos\left(\frac{\pi t}{\tau}\right)\right), \quad 0 \le t \le \tau$$
$$\xi_1^d = 0, \qquad\qquad t > \tau$$

where τ is a user-defined time horizon in which the robot reaches the target location. Given that the orientation control also must drive the robot to correct the lateral deviation, a desired trajectory ξ_2^d related to x is proposed. Let us define the angle ψ as in Fig. 1(a), related to the lateral deviation:

$$\psi = \arctan\left(-\frac{T_{332}s\phi - T_{331}c\phi}{T_{332}c\phi + T_{331}s\phi}\right)$$

where ϕ is given by Eq. 6. Then, the desired trajectory for ξ_2^d is given by:

$$\xi_2^d = \xi_2(0)\frac{\psi(t)}{\psi(0)}, \quad 0 \le t \le \tau$$
$$\xi_2^d = 0, \qquad t > \tau. \tag{10}$$

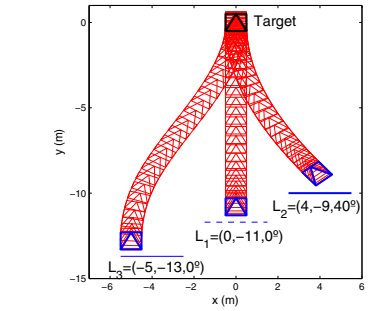
This trajectory is used if $|\phi(0)| > |\psi(0)|$, otherwise, an initial motion is induced by smoothly increasing the desired value ξ_2^d for an initial period $\tau_1 < \tau$.

5 Simulation Results

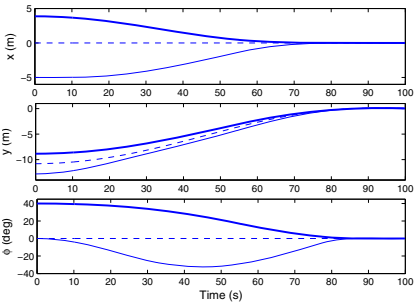
The performance of our virtual target-based approach is evaluated via simulations. The results have been obtained by using Matlab scripting with a closed loop time of 0.3s. The TT is estimated from synthetic images of size 800×600 pixels. Omnidirectional images are generated through the generic camera model [11]. The time to complete the regulation task is set to $\tau = 90$s.

The resulting paths, from three different initial locations, can be seen in Fig. 5(a). The case of the initial location at $\mathbf{L}_1 = (0, -11, 0^o)$ is special, given that $\phi(0) = \psi(0) = 0$. In such a case, $\xi_2^d = 0$ during the navigation. This case is also special because an epipolar control is not able to solve it. In the case of the initial location at $\mathbf{L}_2 = (4, -9, 40^o)$, ξ_2^d is defined by Eq. 10 and similarly for $\mathbf{L}_3 = (-5, -13, 0^o)$, by including an initial increment of ξ_2^d. In all cases, the robot reaches the target with good precision and carries out a smooth motion, as shown in Fig. 5(b). This behavior is obtained with the velocities of Fig. 5(c), given by the trajectory tracking scheme of Eq. 9. In Fig. 5(d), the evolution of the visual measurements of Eq. 7, taken as outputs of the system, are shown. Due to the normalization of ξ_1, the plots for the three cases look similar.

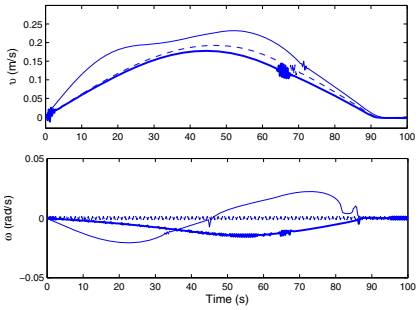
In order to validate the proposed VS scheme for different cameras, the motion of the point features is also presented in Fig. 5(e) for the control from $\mathbf{L}_2$ with a hypercatadioptric projection. For the control from $\mathbf{L}_3$, a paracatadioptric vision system is used and the motion of the image points is shown in Fig. 5(f). Note that the points at the end of the motion, marked with "×", are overimposed to the real target points, marked with "O". The difference between the former points



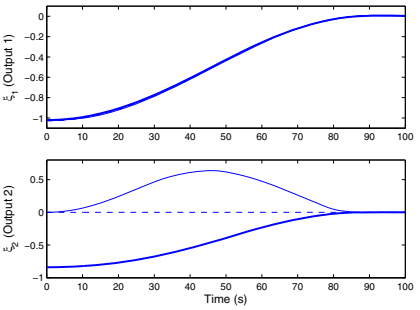
(a) Paths on the $x - y$ plane



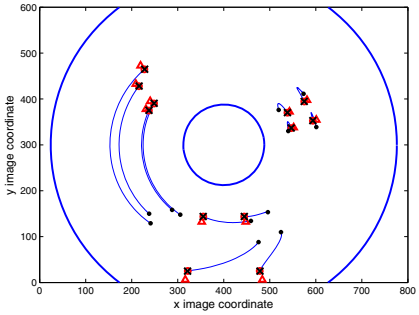
(b) Evolution of the robot pose



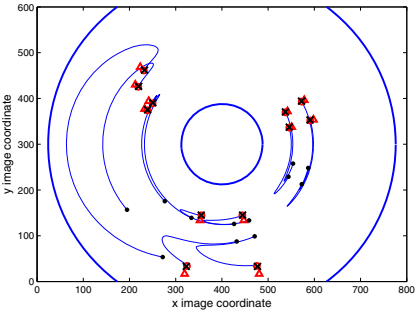
(c) Computed velocities



(d) Evolution of the visual measurements



(e) Image points motion for $\mathbf{L}_2$.



(f) Image points motion for $\mathbf{L}_3$.

Fig. 5. Simulation results for vision-based pose regulation using images from hypercatadioptric (e) and paracatadioptric (f) vision systems. The markers represent: "·"=initial image, "O"=real target image, "△"=virtual target image and "×"=image at the end.

and the virtual target points, marked with "△", avoids numerical problems at the end of the task in the estimation of the TT and makes possible the derivation of our singularity-free controller. Note that we use a calibrated camera but our control scheme is also valid for *uncalibrated* conventional cameras.

6 Conclusions

We have proposed a novel image-based visual servoing scheme that relies on the evaluation of the trifocal tensor computed by using an automatically generated virtual target. The pose regulation problem of a wheeled mobile robot is solved efficiently using the new control scheme, which is based on a few elements of the trifocal tensor relating the initial view from the robot, the current one and the target view. Contrary to other multiple view geometry-based approaches, we avoid control singularities by using the virtual target, easily deduced from the real one by a vertical translation. Thus, a single controller, tracking adequate trajectories, solves the pose regulation problem without the need of switching to a different controller as in previous works. We have illustrated the pertinence of this approach with very promising simulation results. Our ongoing work focuses on real-world experiments with this approach, with different camera models.

References

1. Chaumette, F., Hutchinson, S.: Visual servo control part I: Basic approaches. IEEE Robotics and Aut. Mag. 13(14), 82–90 (2006)
2. Chaumette, F., Hutchinson, S.: Visual servo control, part II: Advanced approaches. IEEE Robotics and Aut. Mag. 14(1), 109–118 (2007)
3. Mariottini, G.L., Oriolo, G., Prattichizzo, D.: Image-based visual servoing for nonholonomic mobile robots using epipolar geometry. IEEE Transactions on Robotics 23(1), 87–100 (2007)
4. López-Nicolás, G., Guerrero, J.J., Sagüés, C.: Visual control through the trifocal tensor for nonholonomic robots. Robotics and Auton. Syst. 58(2), 216–226 (2010)
5. Fang, Y., Dixon, W.E., Dawson, D.M., Chawda, P.: Homography-based visual servo regulation of mobile robots. IEEE Transactions on Systems, Man, and Cybernetics, Part B 35(5), 1041–1050 (2005)
6. Usai, A., Di Giamberardino, P.: Visual feedback for nonholonomic mobile robots: Homography based approach. In: Rigatos, G. (ed.) Intelligent Industrial Systems, pp. 152–181. IGI Global (2010)
7. Becerra, H.M., López-Nicolás, G., Sagüés, C.: A sliding mode control law for mobile robots based on epipolar visual servoing from three views. IEEE Transactions on Robotics 27(1), 175–183 (2011)
8. Becerra, H.M., López-Nicolás, G., Sagues, C.: Omnidirectional visual control of mobile robots based on the 1D trifocal tensor. Robotics and Autonomous Systems 58(6), 796–808 (2010)
9. López-Nicolás, G., Sagüés, C., Guerrero, J.J.: Parking with the essential matrix without short baseline degeneracies. In: IEEE Int. Conf. on Robotics and Automation, pp. 1098–1103 (2009)
10. Hartley, R.I., Zisserman, A.: Multiple View Geometry in Computer Vision. 2. Cambridge University Press (2004)
11. Geyer, C., Daniilidis, K.: A unifying theory for central panoramic systems and practical implications. In: Vernon, D. (ed.) ECCV 2000. LNCS, vol. 1843, pp. 445–461. Springer, Heidelberg (2000)
12. Slotine, J.J.E., Li, W.: Applied nonlinear control. Prentice Hall (1991)
13. Levant, A., Fridman, L.: Higher order sliding modes. In: Perruqueti, W., Barbot, J. (eds.) Sliding Mode Control in Engineering, Marcel Dekker, N.Y., USA, pp. 53–101 (2002)

Assessing the Quality Level of Corn Tortillas with Inductive Characterization and Digital Image Analysis*

Marco A. Moreno-Armendáriz, Salvador Godoy-Calderon, Hiram Calvo, and Oscar M. Rojas-Padilla

Centro de Investigación en Computación, Instituto Politécnico Nacional, AV. Juan de Dios Bátiz S/N, Unidad Profesional "Adolfo López Mateos", México, D. F. C. P. 07738, México
mam.armendariz@gmail.com, sgodoyc@gmail.com, hcalvo@cic.ipn.mx, oscarrjs@gmail.com

Abstract. Characterization and classification of corn tortillas turns out to be an extremely delicate and difficult process when dealing with regulations for import/export and production process certification. In this paper we present a method for non-invasive feature extraction, based on digital imaging and a series of procedures to characterize different qualities of corn tortillas for their later classification. The novelty in this whole method lies in the extremely reduced set of features required for the characterization with only geometrical and color features. Nonetheless, this set of features can assess diverse quality elements like the homogeneity of the baking process and others alike. Experimental results on a sample batch of 600 tortillas show the presented method to be around 95% effective.

Keywords: Inductive characterization, digital image analysis, corn tortilla.

1 Introduction

Tortillas are a fundamental element in the diet of Mexican people. Tortillas have been hand-made since ancient times. As a result of the increasing demand of this product, even from other countries, new production methods have become a necessity for massively producing tortillas; their quality must be assessed to comply with industrial production standards as well as import/export regulations [17]. It has been shown [12, 9] that sensory characteristics are of great importance for consumers, however, for small and mid-size producers there is currently no protocol for quality-control; there is an absence of standards and guides to evaluate the sensory impression of consumers, *i.e.,* color, shape, spots, etc., or some other kind of imperfections resulting from their elaboration and baking process, and thus, tortillas are evaluated manually resulting in the fact that the quality of the tortilla is not uniformly assessed.

* The authors wish to thank the support of the Mexican Government (CONACYT, SNI, SIP-IPN, COFAA-IPN, and PIFI-IPN).

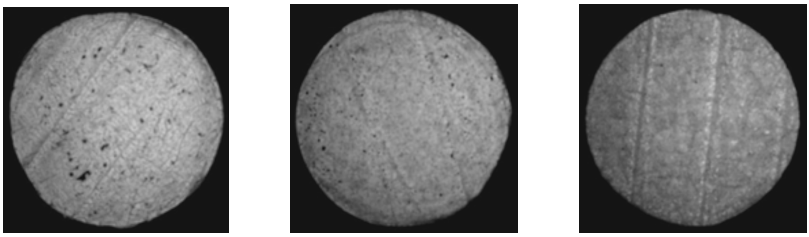
J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 40–53, 2013.

In this work we propose to classify tortillas using digital image analysis, a non-invasive sensor that does not cause damage to the produce.

In Section 2 we describe the techniques used for extracting the features used for tortillas classification; in Section 3 we describe our method of inductive classification; in Section 4 we show results and discussion, and finally we draw our conclusions.

2 Feature Extraction

In this section we will introduce the methods we use for classifying tortillas. Sample tortillas will be taken from three different sources: small producer, supermarket and industrialized. To begin, a digital image of each side of the corn tortilla is taken. See Figure 1.



(a) (b) (c)

Fig. 1. Tortillas sampled from: (a) small shop, (b) supermarket, (c) industrial batch

Geometric and color features are extracted from a sample of corn tortilla images to determine the quality level of the tortilla. Geometric features include curvature, symmetry and border deformations, as well as the presence of some defects like ruptures, holes or bends. Color features include dark/bright spots, color homogeneity and average color.

2.1 Color Features

Each image is separated into Red, Green and Blue channels (RGB), and then three interest areas are determined for each image by color-reducing each channel. Those areas bring evidence of the cooking degree on the tortilla surface. A tortilla can be considered raw (undercooked), well cooked or burnt (excessively cooked) by counting the surface area occupied by bright spots, average color spots or dark spots respectively.

Since images have a color-depth of eight bits, every pixel within each channel holds an intensity value in the range [0, 255]. To reduce the number of intensity levels in an image channel, we establish a threshold in the following manner: each pixel's intensity value is mapped to one of three pre-determined levels by two thresholds. If the intensity level of the pixel reads under 30, then it is mapped to a zero value

(pure black), if it reads any value between 31 and 180, then it is mapped to a value of 64 (mid-value color) and finally, if it reads any value between 181 and 255, it is mapped to a value of 128(high value). These thresholds were determined based on the tortilla's image histograms, so that the areas of interest are successfully extracted. Background pixels are also mapped to a zero-intensity value, so that they appear to be part of the dark spots area, but they are filtered out during the color-features extraction process, see Figure 2. In this Figure f_1 to f_8 are the features used for the classification process described in Section 3.

Once the three interest areas within each image are selected, the following processes take place: contour identification, tortilla surface area filtering and edge characterization; then, features are extracted.

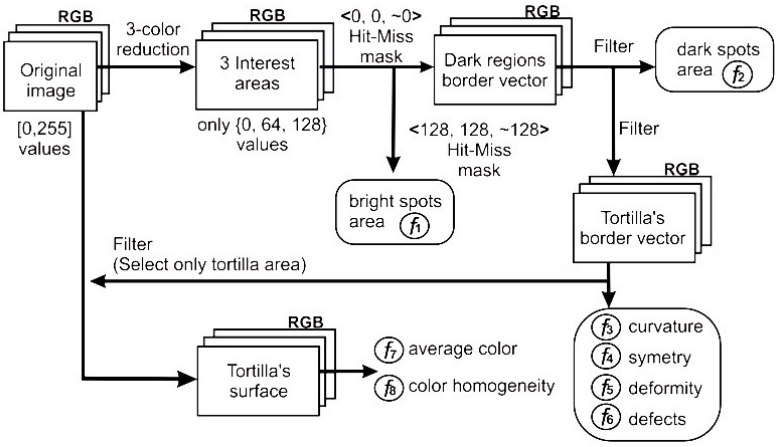


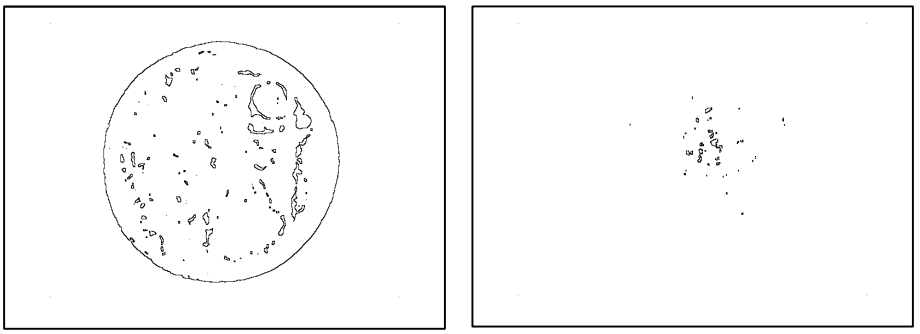
Fig. 2. Image processing process for feature extraction

Fig. 3. An example of a hit-or-miss transformation. It shows that it is necessary for both reference points to meet at the desired target for the operation to be performed, otherwise a gap appears in the region of the resulting image (Taken from [19]).

Contour Identification. Two morphological *hit-or-miss* operators (see Figure 3) are proposed. Each structure element is applied four times, rotating 90° each time, and then we use the union of the results. The first structure element, composed by the values B_{N1} = {0, **¬0**, ¬0} looks for image regions matching one black pixel (the background) followed by two consecutive non-zero values (the tortilla surface). The second structure element B_{C1} = {¬128, **¬128,** 128} looks for regions matching

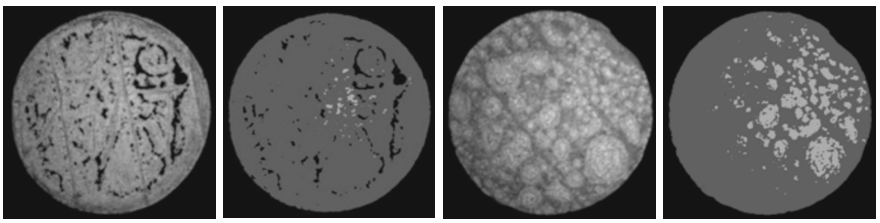
two consecutive non high-intensity pixels followed by a high-intensity pixel, thus, finding raw areas inside the tortilla. See Figure 4.

Excessively cooked regions on the tortilla surface appear as dark spots on the digital image, while raw or undercooked regions appear as bright spots. Both kinds of spots can have varying sizes as can be seen on Figure 5.



(a) (b)

Fig. 4. Result of applying the structure element $B_{N1} = \{0, \neg 0, \neg 0\}$ (a) and $B_{C1} = \{\neg 128, \neg 128, 128\}$ (b) on one sample



(a) (b) (c) (d)

Fig. 5. (a) Tortilla with dark spots from excessively cooked regions (b) interest regions from the same tortilla; (c) Tortilla with bright spots from raw regions (d) interest regions detected.

We propose the following color features:

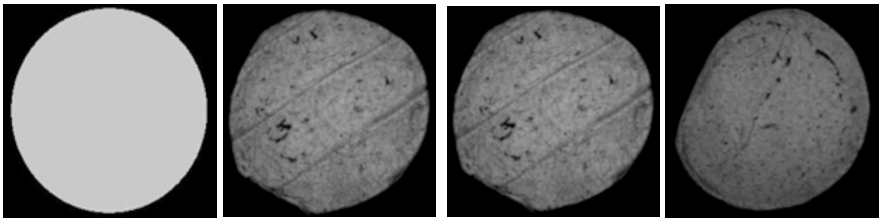
Color homogeneity (***cHom***) describes a uniform temperature distribution during the cooking process and, of course, the average color depends on the corn type the tortilla is made of. A correctly baked tortilla must show a uniform specific color with no darker or brighter regions. A raw or incompletely baked tortilla will show some brighter regions, while a burned or excessively baked one will show some darker ones. Color will be less homogeneous in the last two cases.

Average Brightness (*Lavg*). This feature identifies, as the name implies, the amount of light from the surface, which ideally should be equivalent throughout the surface. This value varies depending on the type of dough used for making the product and may go from dark yellow hues, to whitish yellow. This parameter is obtained by the average of all pixels' color value contained in the tortilla surface.

Variance of light intensity (*Lvar*). The non-homogeneous colouration on the surface is caused by burnt or raw areas and poorly processed corn leftovers. These defects cause the brightness of the surface not to be homogeneous, which may vary dramatically from one pixel to another. In the ideal case, the variance should be zero, so that for a uniform brightness a value close to zero would be expected. For the calculation of this feature, as with the average brightness, all pixels of the surface are used and then the variance of these values is used.

Burned areas (*burntA*). Burned areas represent sections of varying sizes of burnt dough produced by an excess of cooking. Coloring of these areas is common to any hue of the tortillas, making it easy to identify. For its identification we use the border extracted as described previously with the *hit-miss* transformation, by filtering the edges of the burned areas from the tortilla edges. Obtaining this feature is done in an indirect way, because by counting the edge borders we infer that as there are more edge pixels, the size of the area would be grater.

Raw areas (*rawA*). The raw areas represent sections of dough that did not reach the proper cooking and have a lighter color than the rest of the surface. As with the burnt areas, the extraction of the obtained raw areas is obtained from the surface segmentation, and calculation is carried out indirectly by counting the pixels of the edges.



(a) (b) (c) (d)

Fig. 6. (a) Model of a perfect tortilla, (b) tortilla without symmetry, (c) tortilla with a slight defect along its border, (d) tortilla with a large defect along its border

2.2 Geometric Features

In order to get a discrete representation for the tortilla's border, we divide it into 64 arc segments. Each arc is projected over the horizontal axis and the length of the projection is measured [22, 2]. A characteristic descriptor for each tortilla side is made up with the lengths of the 64 projected arcs [8].

Geometry-related features aim to capture those attributes related to the shape of the tortilla including curvature, symmetry and continuity. According to export regulations, a high quality tortilla must have a nearly perfect circular shape with no holes and no bends or breaks along its border (See Figure 6). Since a discrete characteristic chain is used to describe the border of the tortilla, this chain is compared with the corresponding chain of a perfect circle as described in [7] (see Figure 7). If both chains are identical then the tortilla border has the best quality available. The following features were extracted:

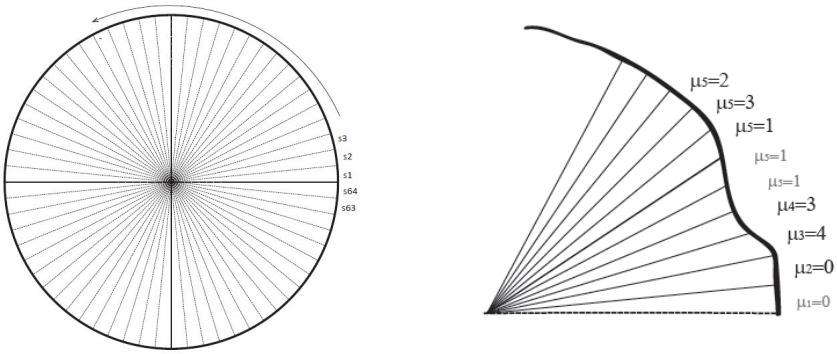


Fig. 7. Projected arcs for measuring the 'roundness' of tortilla. A defect is shown to the right.

Defects (dfct). A defect in a tortilla occurs when a small irregularity appears on the edge, perceptibly altering the circularity of the edge section. This change is often abrupt and has a short distance of no more than one centimeter (no more than 1/8 of the circumference.) Convexity is affected.

Deformations (dfrm). The deformation in a tortilla occurs where one segment has a non-circular tendency. In this segment the edge is straight (no more than ¼ of the circumference.) Convexity is not affected.

Circularity (circ). Its value is calculated by adding up the differences of each one of the diameters with regard to an average.

3 Inductive Classification

One of the most common techniques used to obtain the rules of inductive learning is known as "divide and conquer" [5]. This technique, which appeared in the early 80's, is named after the method applied to construct a rule induction, dividing the initial set of knowledge rules and selecting the rules that provide better coverage rates. Important works using this technique are Michaslki [14, 16], Clark [3], and Rivest [18]; several innovations and/or adaptations of the techniques proposed by them arose for nearly a decade after the publication of these mentioned works.

One of these works to improve learning technique was performed by the same Michalski in the mid-1980s. The STAR method is based on the principle of "divide and conquer" and it further allowed the resolution of everyday problems or applications that have large numbers of possible solutions. Finally it emerged and positioned itself as the BOUNDSTAR and REDUSTAR methodologies (see [13] for an in-depth explanation).

3.1 Tortilla Characterization by Inductive Learning

Once the features for all samples are defined and calculated, we reach the stage of inductive learning where the characterizing rules are learnt. The learning process is based on the model of the BOUNDSTAR algorithm [13] that seeks to obtain a characterization from events calculated by parameterizing features, with the conjunctions of these, trying to get a generalization and thereby obtaining rules which shape knowledge and allow characterization. The last step of this process is to implement the rules to a new group of samples and the evaluation of the classification results are obtained.

3.1.1 Parameterization of Features

Each dataset of characteristics, both color and texture and geometric data has a different distribution for each class along each feature domain. Based on this premise, the parameterization is performed for each set of data for a range in which each of the classes is covered in greater proportion with the rest of them, *i.e.*, this class has a percentage of maximum coverage in this range. Thus, *r* will be obtained for each value of each feature, being *r* the number of classes that we seek to characterize.

The set of all observed events will form an initial knowledge base. Although not all of those may be able to completely cover any class above the others. Some of the features contained in this knowledge base will be discarded, as they are of marginal use for the creation of knowledge rules, since their coverage percentage is very strong on some classes, and very weak for other classes at the same time.

Each event is calculated relying on the histogram of each feature as well. Figure 8 shows the distribution of the feature *Lvar*. We can see that the class 2 (asterisk symbol) will get high percentage of coverage and a considerable difference with respect to the percentage of coverage of other classes, a situation that does not happen for Class 3 (circular symbol) and even less for Class 1 (square symbol), because the samples of these classes are overlapping.

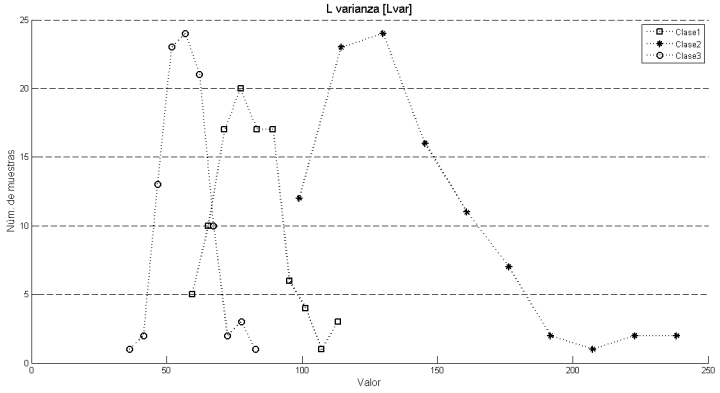


Fig. 8. Lvar feature histograms. As noted, it is possible to achieve a separation of the class 2 which intersects only in a small range with samples of class 1.

An event, either from the original knowledge base or from a later one, can be considered, like in the case of a complex feature, stronger than any other characterization if it achieves the characterization of all the elements of a class under parameters which only the elements of this class meet, while none of the remaining classes' elements meet. So if we happen to have an event that accomplishes these goals, then this event is considered the strongest possible characterization for the elements of the class, and may even be considered as a stop condition for the algorithm.

4 Experimental Results

We obtained a sample of 600 tortillas, from three kinds of commercial establishments (see Figure 1), 200 from each kind. With help from some experts in food engineering, a model of the perfect tortilla made from each possible type of corn was defined (see Figure 6) and then we proceeded to apply the induction classification. We learned from the first half of 300, and then we evaluated with the remaining 300. The extraction of the first two color features of the 300 images of tortilla during the learning phase can be seen in Figure 9. In the case of *Lavg*, it can be seen that the data of class 2 are distant with respect to the other classes. This phenomenon is generated by the use of different materials (different masses or corn flour) which impact directly on the color and brightness of the product. Furthermore, although *Lvar* and *Lavg* are directly related, the distribution of such data is not necessarily the same. As shown in Figure 9, the variance of color (light) samples among those of the same class can be wider. For both features at least one class is easily identifiable above the rest; this will be an advantage for the parameterization and search of characteristic features of each class.

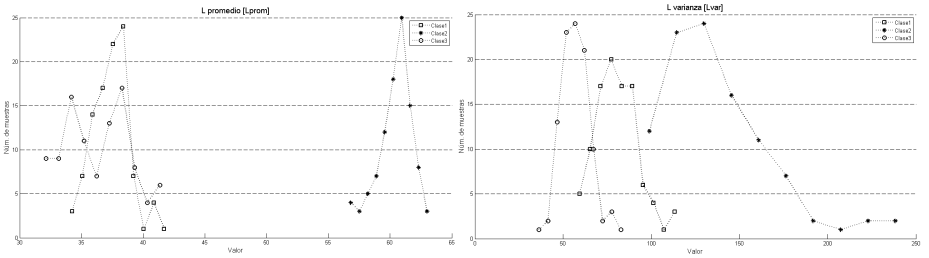


Fig. 9. The first graph shows the data distribution of the three different classes with regard to the average L

The other two color characteristics *burntA* and *rawA* are shown in Figure 10. There it can be seen that for both features the three classes have a poor dispersion, so that a differentiation between samples of a particular group will not be easy, or even impossible to achieve because there is little scatter in the data. Then, this feature is not considered as very important for the classification.

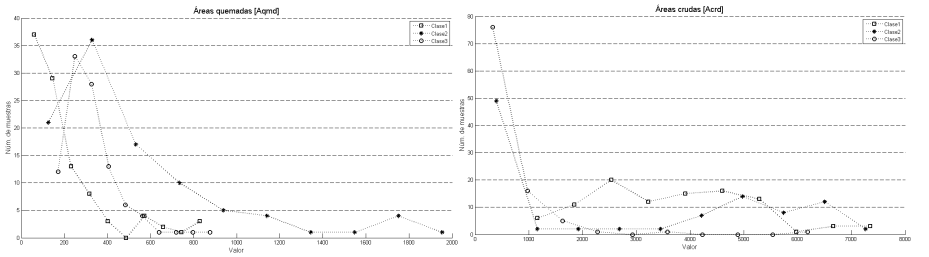


Fig. 10. The first graph shows the distribution of the *burntA* data, which the lower one represents *rawA*; both show a difficult separation of these classes

With regard to the geometrical features, the results extracted from the images can be seen in Figure 11 and Figure 12. *dfct* seems to provide good information for characterization, while as *defm* and *circ* exhibit little disjoint accumulation. A further analysis of the distribution and these results is offered in the discussion of results section.

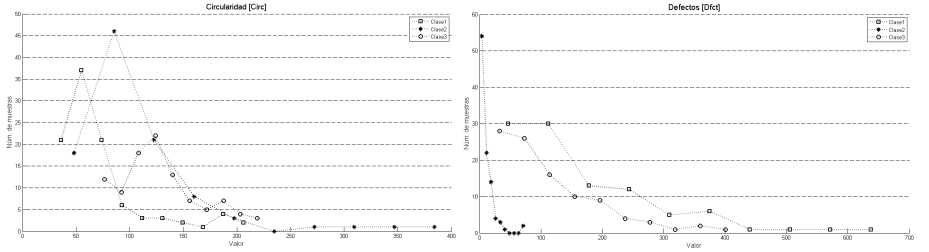


Fig. 11. The circularity *circ* (pictured left), presents some difficulty differentiating classes, on the contrary, the defects *dfct* (right) samples differentiate one class above the rest

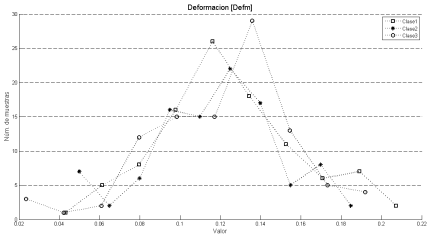


Fig. 12. The distribution of strain (*defm*) in the samples of the three classes presents difficulties for the separation of samples

Other features such as correlation, homogeneity, and entropy distribution are not shown, since they were are not considered in the process of inductive learning.

4.1 Characterization

Once we have the data shown in the histograms above, in order to obtain knowledge rules using inductive learning, it is necessary to first perform the parameterization of the extracted features.

Because the inductive learning method used is BUNDSTAR, and that this is based on a set of events, it is necessary first to define events together. These events will form the initial knowledge base. Table 1 shows the coverage results of these events. The best percentage of coverage obtained for the each class of the feature is highlighted in bold. From Table 5 we can see some interesting phenomena in covering classes:

1. The *Lavg* complex feature under the proposed parameters, is the strongest possible because it manages to strongly characterize to class 2 and thus creates two disjoint sets. The first is that all the elements of the class 2 and the other is formed by all elements in classes 1 and 3.
2. The *Lvar* complex feature can characterize two different classes, using different parameters, classes 2 and 3, being stronger characterization of class 2.
3. The *circ* complex feature, despite all samples characterize class 3 does not achieve it in a convenient way, since 88% of characteristic samples of class 2 are characterized too.
4. The complex feature *dfct* can almost perfectly characterize the class 2.
5. There are other features that were not included in Table 5 because the coverage percentages are not suitable, for example, the feature called *dfrm* is not strong enough to characterize a class because the difference is minimal.

Table 1. Coverage pct. of the proposed complex features from the learning stage

#	Complex feature	Class1	Class 2	Class 3
1	*Lavg* >50	0	**100**	0
2	*Lvar* >91	17	**100**	0
3	*Lvar* <85.5	67	0	**100**
4	*burntA* <220	**70**	21	14
5	220 < *burntA* <450	19	37	**72**
6	*rawA* >1200	**96**	49	10
7	*circ*>60	48	88	**100**
8	*dfct*<45	9	**98**	21
9	0.3262 < *dfrm* < 0.4452	**100**	2	37

Table 2. Set of solutions that characterize better coverage percentages classes. The rules to be used in the generalization are shown in bold.

Class	Solution set	C 1	C 2	C 3
1	*burntA* <220 ∧ *circ*<60 ∧ *Lavg* <50	38	4	0
	***rawA* >1200 ∧ *Lavg* <50**	**96**	**0**	**10**
	0.3262 < *dfrm* < 0.4452 ∧ *Lavg* <50	52	0	0
2	***Lavg* >50**	**0**	**100**	**0**
3	*Lvar* <85.5 ∧ 0.3262 > *dfrm* < 0.4452	0	0	63
	220 < *burntA* <450 ∧ *Lvar* <91	15	0	72
	***Lavg* <50 ∧ *rawA* <1200 ∧ *circ* >60**	**0**	**0**	**90**

Table 3. Coverage percentages obtained using complex features after re-learning

Learning Rules	C 1	C 2	C 3
Lavg >50	0	100	0
(Lavg <50) ^ (rawA>1200)	96	0	10
(Lavg <50) ^ (rawA <1200) ^ (circ >60)	0	0	90

4.2 Evaluation

To validate the proposed rule of knowledge, we used the 300 remaining samples. As mentioned in the previous section, it is possible that during the classification of new samples some of them are not classified in any of the three classes defined. These phenomenon occurs when the samples in question are outside of the parameters of the features and then they can be considered as out of context objects.

The results of the classification of the new group of samples is shown in Table 4.

Table 4. Classification results of 300 new samples using complex features listed in Table 2

	Class 1	Class 2	Class 3	Not classified
Classified Samples	84	100	113	3

Table 5. Classified samples. The rows are read the actual class and columns are the class to which they were assigned according to their features and the learnd rules of knowledge.

		Classification			
		Class 1	Class 2	Class 3	No class
Real Class	Class 1	82	0	16	2
	Class 2	0	100	0	0
	Class 3	2	0	97	1

Table 4 shows that there are samples which are not classified correctly by the learnt rules. This is evident if we see the class 3, which has more samples that it might contain. For validating the results we need to determine to which class the samples really belong, and to which class they were assigned due to their properties. This confusion-matrix is shown in Table 5.

4.3 Discussion of Results

The results of the characterization of the three classes proposals can be seen in Table 1, where the percentages of coverage are achieved by early complex features. From this learning stage it can be appreciated, as anticipated in the extraction of features, that not all features that are useful for the characterization, however, there are complex features capable of characterizing a class above the other with excellent results. Examples are traits *Lavg*> 50 and *Lvar*> 91, which are close to 100% coverage in the class of interest, class 2.

After the implementation of the BOUNDSTAR algorithm, we obtained the ruleset presented in Table 2. This ruleset represents the best solutions for each class and it can be used independently for different purposes.

To validate the learnt rules, we evaluate their performance and determined the following values: memory, precision, specificity and accuracy for each of the classes determined, these results are:

1. The rule for class 1 (tortillas package) is: (*Lavg* <50) ^ (*rawA*> 1200) having an accuracy of 93%, a precision of 97%, a specificity of 99% and 82% memory; this latter being affected mainly by the 18 samples that were classified in another class.
2. The rule that characterizes the class 2 (supermarket tortillas) is *Lavg*> 50, this rule has the distinction of being an ideal case, it is a strong rule. This property can be seen in the histogram of *Lavg*, see Figure 8; this graph shows the classes 1 and 3 separated by a wide range of values with respect to Class 3, this means that visually the samples of class 2 are much clearer and this is enough to characterize.
3. The characterization rule for class 3 (tortilla from a neighborhoodstore) is: (*Lavg* <50) ^ (*rawA* <1200) ^ (*circ*> 60), which has an accuracy of 93%, Memory 97%, specificity of 92% and a precision of 85%. This low precision is due to the 16 samples of Class 1 that are classified in this class. This problem is due to the similarity of its luminosity. In this case use the circularity is used for distinguishing these two kinds of samples; however this is obtained yielding a lower precision with regard to that of the other classes.

5 Conclusions

We have found that the inductive characterization of three classes of producers is achieved with high percentages of coverage and precision; particularly in the case of

Group 2 with a coverage of 100%. Importantly, despite the diversity of variables involved in making tortillas, it was possible to find distinctive patterns of each manufacturer in production. The proposed color and geometric features were useful for achieving this classification. The most important features in the classification, given the high percentage of coverage for the desired classes were color features. Significantly, color features become extremely important in characterizing when considering the preference of the consumer.

In general, we could say that it is possible to make a quantification of the visual features of tortillas, using only images, and using these features to obtain an inductive characterization of different producers. This is shown in the results of Table 2, since the exposed ruleset characterizations represent the best for each class, and they are themselves a numerical representation of the visual properties of each class of common tortillas. Despite the good results obtained by the learned knowledge rule, characterization results could be improved by using not exclusively visual attributes, but also considering other features resulting from a biochemical analysis of the samples, such as moisture, density, elasticity, among others. This latter is left as a future work.

References

1. Adams, J., Parulski, K., Spaulding, K.: Color processing in digital cameras. IEEE Micro. 18(6), 20–30 (1998)
2. Cai, W., Yu, Q., Wang, H.: A fast contour-based approach to circle and ellipse detection. In: Fifth World Congress on Intelligent Control and Automation, vol. 5, pp. 4686–4690 (2004)
3. Clark, P., Tim, N.: The CN2 induction algorithm, Tahoe City,CA (1989)
4. Cortés-Goméz, A., Martín-Martínez, E.S., Martínez-Bustos, F., Vázquez-Carrillo, G.M.: Tortillas of Blue Maize (zeamays l.) Prepared by a Fractioned Process of Nixtamalization: analisys using response surface methodology. J. of Food Eng. 60(3), 273–281 (2001)
5. Domingos, P.: Unifying Instance-Based and Rule-Based Induction. Machine Learning 24(2) (1996)
6. Gonzalez, R., Woods, R.: Digital Image Processing, 3rd edn. Prentice Hall, Pearson (2008)
7. Gupta, L., Srinath, M.D.: Contour Sequence Moments for the Classification of Closed Planar Shapes. Pattern Recognition 20(3), 267–272 (1987)
8. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer (2003)
9. Herrera-Corredor, J., Saidu, J., Khachatryan, A., Prinyawiwatkul, W., Carballo-Carballo, A., Zepeda-Bautista, R.: Identifying Drivers for Consumers Acceptance and Purchase Intent of Corn Tortilla. J. of Food Sc. 72(9), 727–730 (2007)
10. Ibarra-Manzano, M.A., Devy, M., Boizard, J.L.: Real-time classification based on color and texture attributes on an FPGA-based architecture. In: Conference on Design and Architectures for Signal and Image Processing, pp. 250–257 (2010)
11. Malamas, E.N., Petrakis, E.G.M., Zervakis, M., Petit, L., Legat, J.D.: A Survey on Industrial Vision Systems, Aplications and Tools. Image and Vision Computing 21(3), 171–188 (2003)

12. Mery, D., Chanona-Pérez, J.J., Soto, A., Aguilera, J.M., Vélez-Rivera, N., Arzate-Vázquez, I., Gutiérrez-López, G.F.: Quality Classification of Corn Tortillas using Computer Vision. J. of Food Eng. 101, 357–364 (2010)
13. Michalski, R.S., Stepp, R.: Automated Construction of Classifications: Conceptual Clustering versus Numerical Taxonomy. IEEE Trans. on Pattern Analysis and Machine Intelligence PAMI-5(4), 396–410 (1983)
14. Michalski, R.S.: A theory and methodology of inductive learning 20 (1983)
15. Michalski, R.S., Chilausky, R.L.: Learning by being told and learning form examples: an experimental comparison of the two methods of knowledge adquisition in the context of developing an extra system for sorbeandesease diagnosis. Policy Analysis and Information Systems 4(2), 125–160 (1980)
16. Michalski, R.S., Mozetic, I., Hong, J., Lavrac, N.: The multi-purpose incremental learning system AQ15 and its testing application to three medical domains. In: Proceedings of the Fifth National Conference on Artificial Intelligence, Philadelph (1986)
17. Norma-Oficial-Mexicana, NOM-187-SSA1/SCFI-2002: Productos y servicios. Masa, tortillas, tostadas y harinas preparadas para su elaboración y establecimientos donde se procesan. Especificaciones sanitarias. Secretaría de Economía, Estados Unidos Mexicanos (2003)
18. Rivest, R.L.: Learning decision lists. Machine Learning, 229–246 (1987)
19. Soille, Morphological Image Analisys. Springer (2004)
20. Tahir, M.A., Bouridane, A.: An Fpga Based Coprocessor for Cancer Classification Using Nearest Neighbour Classifier. In: IEEE International Conference on Acoustics, Speech and Signal Processing, vol. 3, pp. III-1012—III-1015 (2006)
21. Terasic Inc., `http://www.terasic.com.tw`
22. Watanabe, T., Matsumoto, M.: Recognition of Circle Form Using Fuzzy Sequential System. In: Twenty-First International Symposium on Multiple-Valued Logic, pp. 85–92 (1991)

3D Representation for Object Detection and Verification

Luis Villavicencio, Carlos Lopez-Franco, Nancy Arana-Daniel, and Lilibet Lopez-Franco

Computer Science Department, Exact Sciences and Engineering Campus, CUCEI, University of Guadalajara, Av. Revolucion 1500, Col. Olimpica, C.P. 44430
felipe.vive@gmail.com,
{carlos.lopez,nancy.arana,lilibet.lopez}@cucei.udg.mx

Abstract. In this paper we introduce a representation for object verification and a system for object recognition based on local features, invariant moments, silhouette creation and a 'net' reduction for depth information. The results are then compared with some of the most recent approaches for object detection such as local features and orientation histograms. Additionally, we used depth information to create descriptors that can be used for 3D verification of detected objects. Moments are computed from a 3D set of points which are arranged to create a descriptive object model. This information showed to be of matter in the decision whether the object is present within the analyzed image segment, or not.

Keywords: object detection, object verification, visual pattern recognition.

1 Introduction

Object recognition is a challenging task that involves several steps aimed to find a relation between an input image and a set of previously known objects [1]. Recent work has brought techniques that allow detection at real-time. Nevertheless, any method applied to object recognition is likely to find the desired object where there is not, that is, it could throw false positive responses. The false positive rate increases when objects are relatively small or the train data is not rich enough or does not describes the object very accurately. The detection step consist in matching an area or points in the image to a known object model, whilst, a verification step goes further and reinforces the decision taken decreasing the false positive rate. Thus, a verification step refines the object detection checking whether the areas really contain the target objects [2].

As information is added to the detection step it becomes slower and heavier. We analyze the inclusion of 3D information in a verification step after detection with the main objective of reducing the false positive rate. The addition of depth information helps to create descriptors that can be easily used to undertake a verification step successfully. This data is added to a set of keypoints forming a

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 54–63, 2013.

3D model from which we compute invariant moments. This approach generates models that are light and descriptive

In this paper, we propose to get a small descriptor based on invariant moments to increases the effectiveness of classification. SURF keypoints [3], and Support vector machine classifiers (SVM) [4] are used. We describe algorithms for contour definition and 3D information reduction into a grid over the object. The paper is organized as follows. In section 2, we describe the tools put together to integrate the recognition system. Section 3 presents the proposed methods. In section 4, we present experimental results. Finally, the conclusions are given in section 5.

2 Background

The use of visual features for object detection is very common and functional. For short, features are points of interest that describe the image and make the correspondence problem able to be solved [5]. Robust features identify objects despite changes in illumination, orientation, translation, scale, noise and distortions. Feature descriptors add information of the neighborhood surrounding the key point. The Speeded-Up Robust Features algorithm (SURF) proposed by Bay et al. [3] finds features using integral images and Haar-like features. SURF features provide robustness and speed compared to similar approaches.

Invariant Moments. Moments provide compact information of a data set. A pattern may be represented by a density distribution function, moments can be obtained for a pattern representing an object, and they can be used to discriminate between objects (or classes)[6]. This technique has been previously used in pattern and object recognition as far as the early 60s [7,6,8,9]. Nevertheless, they were usually applied on measures from an RGB image and the classifiers used were simple. The general two dimensional equation for moments is

$$m_{pq} = \sum_x \sum_y x^p y^q f(x, y) \tag{1}$$

where $f(x, y)$ is a function of the variables, commonly used functions for images are the gray scale function and histograms, this last one is related to the density distribution function. The order of moments is $(p + q)$ [7,9]. The first order moments can be used to locate the centroid of the set of points.

$$\bar{x} = \frac{m_{10}}{m_{00}}, \quad \bar{y} = \frac{m_{10}}{m_{00}} \tag{2}$$

If we compute moments considering a translation to the centroid, we generate central moments.

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y) \tag{3}$$

Central moments, μ_{pq}, can be made invariant under scale dividing them by μ_{00}^{γ}. μ_{00} defines the area so this is a scaling normalization.

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \quad \gamma = [(p + q)/2] + 1 \tag{4}$$

We use a set of 3D points for the moment functions. If we compute a density function of the form $F_x(X) = \int f(u)\ du$ counting the number of occurrences of every point, the probability value assigned to them will be equal because each point appears only once. Thus $f(x, y)$ is assumed as 1 all the time.

The first four moments are descriptive measures for a distribution. Using moments for visual pattern recognition not on the RGB images but on depth information gives a new perspective on the use of moments, since 3D information describes the distribution of points that conform an object.

3 Object Recognition System

This section is focused on describing the recognition system. First, SURF points are extracted from an input image and matched to known models. Small areas around matches are subject to further inspection. Using features we construct a contour taking points with a sliding orientation window based on the magnitude measure. Next, we proceed to extract depth information within the boundaries of the contour and reduce the number of points that will be used to compute the moments. The reduction is intended to generate a smaller set with rich information and less heavy. This is done adjusting a mesh grid over the object. To do this, a fixed number of points are initialized in coordinates inside the bounding box of the contour then they are migrated to enhance the model. Depth information is attached to the coordinates. Then, we calculate the invariant moments over this reduced 3D set to form a small descriptor. Finally, the descriptor is used for classification using a linear SVM. This process is seen in Fig. 1.

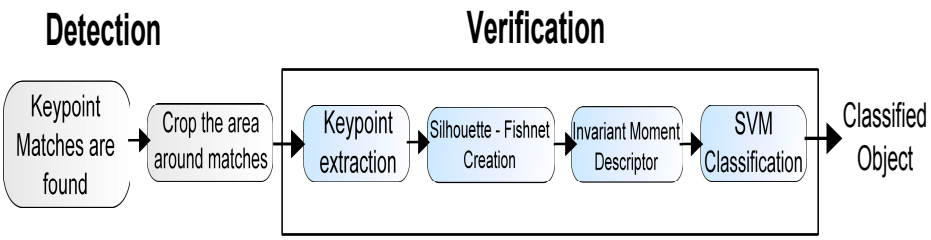


Fig. 1. Steps for the object recognition system

Keypoints. The first step is the extraction of keypoints using the SURF algorithm in an image containing the object, these keypoints are later used to generate a contour. Some other alternatives were tested, including morphological operations and edge or corner detectors but they were not as robust. Feature description resulted the most efficient method for the robustness of points through changes in scene variations. The SURF keypoints used skip descriptor formation which increases speed. Sample SURF keypoints can be seen in Fig. 2.

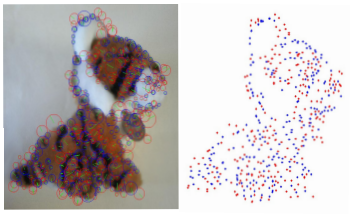


Fig. 2. SURF points extracted for an object

3.1 Silhouette Extraction Algorithm

To create the silhouette from the set of keypoints, we start defining the bounding box of keypoints and find the centroid. It is used to translate and scale the points. Magnitude and orientation are calculated for all points using

$$m(x, y) = \sqrt{(x - x_0)^2 + (y - y_0)^2} \tag{5}$$

$$\theta(x, y) = tan^{-1}\left(\frac{y - y_0}{x - x_0}\right) \tag{6}$$

Where (x_0,y_0) is the centroid. Next, we break the 360 degrees into a number of bins corresponding to the number of points in the contour. For instance, if we want 72 points, bins represent 5 degrees each. A sliding orientation window is used to take points with the highest magnitude for every bin.

Occasionally, some points may be zero because SURF key points were not triggered in certain zones. We scan for blanks and make a linear interpolation to infer missing information using the previous and next known points in the silhouette. Examples of the silhouettes obtained can be seen in Fig. 3.

The number of points in the silhouette has to be carefully chosen. Taking only a few points will create contours with points distant from each other not describing objects correctly. So, small sizes may cause lack of information for descriptors. Very large sizes may cause much sparse information and overuse of interpolation. Since points are selected from SURF features, the contour is limited to the number of keypoints. Taking a large number will produce many blanks.

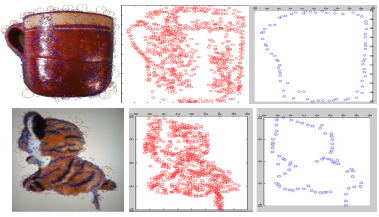


Fig. 3. Sample Silhouettes for two objects

3.2 Fishnet Reduction Algorithm

After silhouette creation, we add depth information. If the area occupied by the object is big, it leads to a large set of points. We perform a reduction of the information adjusting a net over the object. Having depth and RGB images aligned, 3D information is cropped to retain only the area contained inside the bounding box of the silhouette.

First, we define the number of points and generate (x,y) coordinates by sectioning the bounding box evenly. This creates a set of bidimensional points $Grid = \{(x_1, y_1), (x_2, y_2), ..., (x_n, y_n)\}$, the value n is a constant that indicates the number of points in a single direction. Besides,

$$\begin{gathered} x_i \in [x_{min}, x_{max}],\ y_i \in [y_{min}, y_{max}] \\ x_1 = x_{min}, \qquad y_1 = y_{min} \\ x_n = x_{max}, \qquad y_n = y_{max} \\ x_{i+1} = x_i + x_{inc},\ \ y_{i+1} = y_i + y_{inc} \\ x_{inc} = \frac{x_{max} - x_{min}}{n}, y_{inc} = \frac{y_{max} - y_{min}}{n} \end{gathered} \tag{7}$$

minimum and maximum values are defined by the contour bounding box. The result of this process can be seen in Fig. 4 (up).

Since we take the bounding box measures some points might lie outside the contour. The second step is to check that points lie inside the contour. Points are valid if there exist both higher and lower values in points from the silhouette on the vicinity of x and y dimensions. Then, for each non-valid point we take two valid points and move the invalid one between them, this can be seen as a biased migration. The resulting net can be seen in Fig. 4 (down).

For the third step, we add the z coordinate (depth) to the valid (x, y) points. Since depth and RGB images are aligned we append the depth information

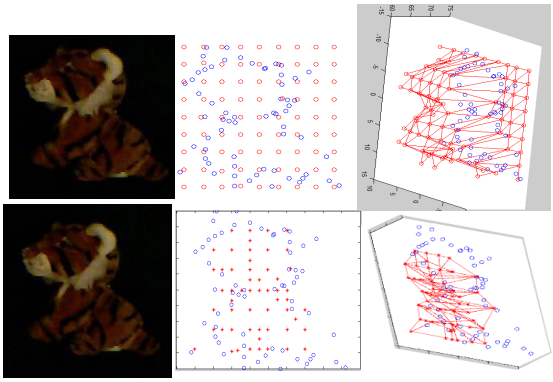


Fig. 4. Net creation. Up, Some points lie outside the contours. Down, points outside the silhouette are migrated.

located at or closest to (x,y). We check that z is a non-zero quantity. When we find null depth points, we look for the next closest point to the coordinates. At the end we have a 2D set of points for the silhouette and a set of 3D points arranged in a net over the object.

3.3 Descriptor Formation

We use the four first invariant moments of each x, y, and z to create the descriptor. The first moment is the mean of the distribution. The second moment is the variance of the set which tells their spread. The third moment determines skewness which measures the symmetry on the shape. The fourth moment defines kurtosis which measures how flat or peaked a distribution is. This results in a 12 elements vector with rich information about the structure of the distribution. After the moments are calculated the vector is used as an input to a linear SVM which classifies the object.

4 Experimental Results

A data set was created composed by 9 objects (cups, totems, tigers, tennis, console controls, globes, shoes, hair dryers and irons) with around 50 images each, see Fig. 5. The images included changes in the scene conditions. For every object 15 images were used to train the SVMs and 25 images for the test stage. The silhouettes created were formed by 72 points, using bins of 5 degrees. On 3D experiments, the fishnet consisted of an 81 point (9x9) mesh. The classification tests were made using cropped, segmented images that included one object and had a discriminative background.

4.1 2D Classification Results

The first experiment consisted in the comparison of 2D contours for invariant moments and Histograms of Oriented Gradients (HOG) descriptors [10] using

Fig. 5. Objects conforming the data set

SVMs classifiers for both. The descriptor was constructed using the information from the first four moments and SURF keypoints (x, y, gradient orientation and scale) generating a 296 element vector. The test consisted of binary classification, with one object being discriminated from another, 1 vs 1, see table 1. In the second test, the long descriptor was compared to a small one with only invariant moments. This test consisted of binary classification using the 4 objects with an object being discriminated from the rest, 1 vs all, see table 2. The results are summarized in result tables. Each row indicates, the percentage of: (a) correct recognitions (CR), (b) false positives (FP), a different object is classified as the target, and (c) false negatives (FN), the target object is classified as a different one.

Table 1. 2D 1 vs 1 Test

1 vs 1 Classification Percentages						
Test	SURF-Moment Desc.			HOG descriptor		
Name	CR	FN	FP	CR	FN	FP
Cups and Tigers	73%	9%	18%	76%	10%	14%
Cups and Tennis	97%	3%	0%	94%	3%	3%
Cups and Totems	92%	8%	0%	97%	0%	3%
Tennis and Tigers	88%	4%	8%	96%	0%	4%
Tennis and Totems	100%	0%	0%	95%	5%	0%
Tigers and Totems	97%	3%	0%	73%	9%	18%

Table 2. 2D 1 vs All Test

1 vs all Classification Percentages						
Test	SURF-Moment Desc.			Moment descriptor		
Name	CR	FN	FP	CR	FN	FP
Cups	88%	10%	2%	84%	6%	10%
Tigers	88%	10%	2%	86%	9%	5%
Tennis	84%	0%	16%	86%	0%	14%
Totems	80%	10%	14%	86%	4%	10%

For the first test, which compares contours and HOG, both SURF-moment and HOG descriptors achieved similar results which shows that moments are a good measure for classification. The second test analyzes classification using moments alone. We can see that SURF information inclusion does not cause a great difference in results. Since Moment-only descriptors throw similar results, we could spare the addition of SURF information. A final test was made in multi-class classification where objects were classified all at once comparing again SURF-moment and HOG descriptors. In this case HOG descriptors made a correct classification of 56% of the objects while the Silhouette-Fishnet-Moment approach obtained a 65% of correct classifications.

4.2 3D Classification and Detection Results

The data set was extended to include depth information. RGB and depth images were aligned and the whole silhouette-fishnet-moment approach was applied. Classification was performed using a very long descriptor (SURF and moments information) and then a simple 12 element vector that only included the first four moments for x, y ans z. In the first experiment, table 3, we compared the two descriptors in order to determine the effect of 3D data inclusion on the computation of moments. Binary classification with two objects was performed. The second experiment, table 4, shows a comparative between the reduced set and the complete 3D set for moment generation, to determine how the reduction affected the pattern. Binary classification with 5 objects was performed for each test, 1 vs all.

Table 3. Results for the 3-D 1 vs 1 Tests

1 vs 1 Classification Percentage						
Test	SURF-Moment Desc.			Moment descriptor		
Name	CR	FN	FP	CR	FN	FP
Hair Dryer and Tigers	85%	5%	10%	100%	0%	0%
Iron and Cups	88%	8%	4%	98%	2%	0%
Shoes and Cups	82%	8%	10%	95%	0%	5%
Iron and Tigers	80%	10%	10%	92%	4%	4%
Globe and Tigers	78%	12%	10%	89%	11%	10%

Table 4. Results for the 3D 1 vs all Tests

1 vs all Classification Percentage						
Test	Reduced Set			Full set		
Name	CR	FN	FP	CR	FN	FP
Hair Dryer	90%	6%	4%	92%	2%	6%
Iron	90%	8%	2%	78%	12%	10%
Tigers	92%	4%	4%	85%	10%	5%
Cups	100%	0%	0%	80%	10%	10%
Shoes	90%	6%	4%	82%	10%	8%

In the first test, the extra information did not increase efficient classification but only made the descriptor heavier. The classification performed by the descriptors using only moments was better for all tests, 3D information turned out to enhance the classification process greatly. In the second test, we can see that using the full set for descriptors did not end in better results. Using the reduced set also reduces time needed to compute moments and improves classification. Finally, a multi-class 3D classification test was performed using the 12 element descriptor and a 70% of correct classification was attained, which shows that depth information enriches the descriptor.

When classification tests were finished, we created a system for object recognition. Using a set of images taken at not preprocessed scenes. The recognition system described in the previous section was implemented. Detection was performed by SURF feature matching and for verification we used the silhouette-fishnet-moment approach. The experiments detected only a class of object at once. An example of the process is seen in Fig. 6. Results are shown in table 5, where we can see that the verification step helps to reduce false positives and enhances detection.

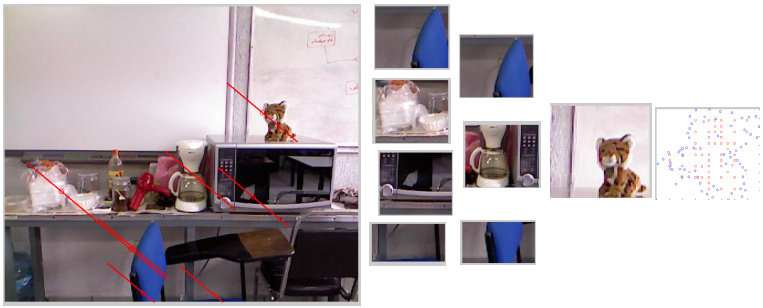


Fig. 6. Recognition system steps exemplification

Table 5. Results for the Object Recognition

Recognition system results				
Object	Matches Found	CR	FN	FP
Hair Dryer	51	94%	2%	4%
Iron	38	89%	6%	5%
Tigers	57	98%	0%	2%
Cups	41	86%	7%	7%
Shoes	36	86%	6%	8%

5 Conclusions

In this paper we presented an object recognition system based on an invariant moment descriptor that includes depth information. Due to the addition of 3D information, invariant moments make small and robust descriptors, outperforming larger descriptors. The inclusion of depth information improves object classification tasks in a huge way. Finally, when a verification step is performed after detection, an important reduction on false positives is achieved. This is very important in many computer vision applications

Acknowledgments. The authors would like to thank CONACYT CB-156567, CB-106838 and University of Guadalajara.

References

1. Jain, R., Kasturi, R., Schunck, B.G.: Machine Vision. McGraw-Hill, Inc. (1995)
2. Baerveldt, A.J.: A vision system for object verification and localization based on local features. Robotics and Autonomous Systems 34(2-3), 83–92 (2001)
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). Computer Vision and Image Understanding 110(3), 346–359 (2008)
4. Cortes, C., Vapnik, V.: Support-vector networks. Machine Learning 20, 273–297 (1995)
5. Ma, Y., Soatto, S., Kosecka, J., Sastry, S.S.: An Invitation to 3-D Vision: From Images to Geometric Models. Springer (2003)
6. Hu, M.K.: Visual pattern recognition by moment invariants. IRE Transactions on Information Theory 8(2), 179–187 (1962)
7. Belkasim, S., Shridhar, M., Ahmadi, M.: Pattern recognition with moment invariants: A comparative study and new results. Pattern Recognition 24(12), 1117–1138 (1991)
8. Mercimek, M., Gulez, K., Mumcu, T.V.: Real object recognition using moment invariants. Sadhana 30, 765–775 (2005)
9. Rizon, M., Yazid, H., Saad, P., Yeon, A., Shakaff, M., Saad, A.R.M., Mamat, R., Yacoob, S., Desa, H., Karthigayan, M.: Object detection using geometric invariant moment. American Journal of Applied Sciences 2, 1876–1878 (2006)
10. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proceedings of the 2005 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, pp. 886–893. IEEE Computer Society, Washington, DC (2005)

3D Tracking and Control of UAV Using Planar Faces and Monocular Camera

Manlio Barajas, José Pablo Dávalos-Viveros, and J.L. Gordillo

Center for Intelligent Systems, Tecnológico de Monterrey, Monterrey, México
{mf.barajas.phd.mty,a00805951,jlgordillo}@itesm.mx
http://www.itesm.edu

Abstract. A method for tracking the 3D pose and controlling an unmanned aerial vehicle (UAV) is presented. Planar faces of target vehicle are tracked using the Efficient Second Order Minimization algorithm, one at a time. Homography decomposition is used to recover the 3D pose of the textured planar face that is being tracked. Then, a cuboid model is used to estimate the homographies of the remaining faces. This allows switching faces as the object moves and rotates. Cascade and single PID controllers are used to control the vehicle pose. Results confirm that this approach is effective for real-time aerial vehicle control using only one camera. This is a step towards an automatic 3D pose tracking system.

Keywords: 3D Tracking, 3D Pose Estimation, Aerial Vehicle Control, Homography Decomposition, Cuboid Tracking, Polygon Mesh Tracking.

1 Introduction

For autonomous navigation, a localization method and a control strategy are determinant elements to achieve success in navigation tasks. Visual 3D pose estimation of objects in real environments has been an important topic in literature because cameras have shown to be a reliable source of environment information. Common visual approaches use a source of depth information, such as stereo setups [15,10] and laser range data [14].

In addition to cameras, Unmanned Aerial Vehicle (UAV) control approaches rely on multiple sensor data for pose estimation and control. Here we're interested in those approaches that work using image data as the main source of information. For example, in [13,6,9,11], visual servoing is done in order to control the vehicle pose.

3D object tracking can deliver 3D pose information to the vehicle controller. 3D Tracking may be accomplished by using full featured models [12,4] or by using approximated models [7]. While full featured models allow having a very exact representation of the real world entity, it is not well suited in all situations having any sort of known 3D model. Approximated models can be used to overcome this constraint. This was validated at theoretical level in [2].

In this article, the method presented in [2] for 3D pose tracking is validated with the control of an UAV. An aerial vehicle is controlled using the visual pose

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 64–73, 2013.

estimation obtained using method [2] as feedback, in a Position Based Visual Servoing (PBVS) system.

This article is divided as follows: section II will introduce work related to the problem of 3D tracking and visual servoing of UAVs. On section III, a method for 3D object tracking is reviewed. Section IV presents the control strategy used to achieve control over the vehicle for then, in Section V, presenting the experimental results. In section VI conclusions are remarked.

2 Related Work

There are two main disciplines related to this work. First, 3D tracking, which should provide the pose estimation that is used frame to frame to effectively control the aerial vehicle. But also, visual servoing, since output from the 3D tracking is used to directly feed a control loop.

Visual servoing of aerial vehicles has been a well developed topic and, because of its complexity, some works are focused on specific parts of the process. For example, in [6], a pure Image Based Visual Servoing (IBVS) for landing and take-off process is developed. While this work has the advantage of not requiring a full 3D reconstruction, it can't be used for full flight control (executing paths). In a similar manner, in [9], a planar patch is used for 3D pose estimation of the UAV. This is closer to our work, but it's restricted in space. A more robust work is done in [13], where the full object pose is controlled. Unfortunately, this approach requires a stereo setup for 3D reconstruction. A common pattern in these works is the use of cameras on the UAV.

In the context of 3D tracking, Cobzas and Sturm [5] presented a method for tracking that uses standard 2D tracking in a 3D pose parameter space (6 parameters, one for each DOF). A drawback is that, when changing the parametrization to handle 3D pose changes, it results that even some movements that could be tracked without a problem by tracking planes individually, can produce failure when tracking them using this parametrization. Another idea introduced by them are constraints between planes to make tracking more stable.

On the same line of parametrizing directly in euclidean space, Panin and Knoll [12] proposed a method for tracking objects that uses Mutual Information as similarity measure, instead of the common SSD, and perform a Levenberg-Marquardt optimization. Authors report performance of 2 *fps* because of the Mutual Information step. Moreover, Panin's method requires at each iteration projecting a full CAD model to image plane (and a z-test for determining visibility).

Another approach to 3D pose tracking is that of Manz et al. [7]. In their work, they use a custom simplified model for 3D pose tracking based on feature points. Their work shows robust performance on real-time vehicle tracking. But this method has the disadvantage of requiring a fine tuning of descriptors and the model for each different tracked target.

In this work the method introduced in [2] is used for tracking and controlling an UAV using a remote camera. Since this 3D tracking method assumes

that tracked object has planar faces, a cuboid shaped object was installed over the UAV. At any given time, one of the faces of the cuboid is tracked. Homography decomposition is used for a 3D reconstruction. The 3D reconstruction is necessary since, when the object rotates, it's used to predict the adequate transformation for the new face to track.

3 3D Object Tracking Using Planar Faces

In this section, a method for tracking objects that move in a 3D world is reviewed. This method uses planar faces to achieve full 3D object tracking. Only one face is tracked at image level using the ESM algorithm. Once a 2D homography between the reference template and the current image is obtained, homography decomposition is used to obtain a 3D reconstruction of that face. Then, a set of precomputed 3D transforms between the different faces of the model allow obtaining its full 3D reconstruction. A plane selection method is used to determine which face is the most convenient to track in the next iteration. If the face that is being tracked needs to be switched, then a 3D to 2D projection is done to obtain the current homography for the new face to be tracked. This steps are illustrated in figure 1. Please refer to [2] for more details on this method.

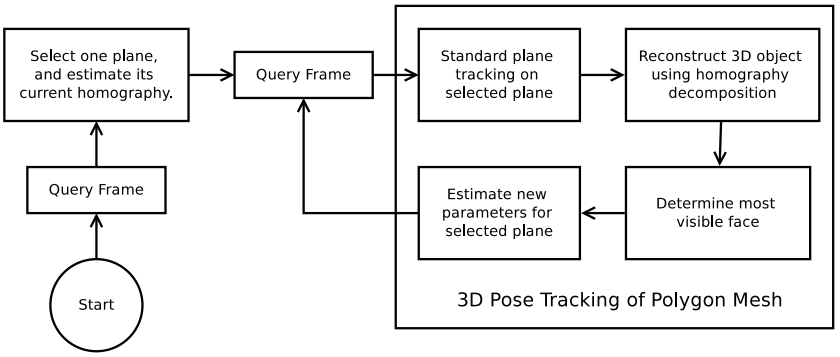


Fig. 1. General overview of the 3D Object tracking method [2]. In the initialization stage, a human operator has to select one plane to trigger the tracking loop.

3.1 Plane Tracking of Planar Faces

In order to recover the full pose of the target object, visual tracking over a single face is done. Because faces are planar, it's possible to use existing plane tracking methods for image alignment. For this work, the Efficient Second Order Minimization proposed by Malis [8] was implemented. This method has proven to provide higher convergence with less global error [8]. Also, this algorithm is fast enough to achieve real-time performance.

This method works by iteratively updating parameters $\mathbf{p} := \mathbf{p} \circ \mathbf{\Delta p}$ where $\mathbf{\Delta p}$ can be evaluated as:

$$\Delta \mathbf{p} \approx -2(\mathbf{J}(\mathbf{e}) + \mathbf{J}(\mathbf{p}_c))^{+}(\mathbf{s}(\mathbf{p}_c) - s(\mathbf{e})) \tag{1}$$

Where $\mathbf{p}$ are the current parameters, $\mathbf{J}$ is the Jacobian, as presented in [1], and $\mathbf{s}$ is the transformed current image. $\mathbf{e}$ is the identity parameter set, in this case $\mathbf{0}$. This method assumes the following homography parametrization:

$$\mathbf{W}(\mathbf{x}; \mathbf{p}) = \mathbf{G} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = \begin{pmatrix} 1 + p_1 & p_3 & p_5 \\ p_2 & 1 + p_4 & p_6 \\ p_7 & p_8 & 1 \end{pmatrix} \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \tag{2}$$

3.2 3D Object Reconstruction

The approach proposed in [2] is implemented to 3D reconstruct the object provided a single face of it and, in this case, assuming the real world dimensions of the object are known. In the 3D reconstruction stage, two main steps are done. First, a selected plane is tracked and then 3D reconstructed using homography decomposition. Then transformation matrices are used to obtain the full 3D object reconstruction.

Homography decomposition deals with reconstructing the 3D pose of a planar surface given its projection on image plane. When working with plane tracking, usually only 8 parameters are used to accomplish tracking, which are mapped to the 3×3 homography transform, as shown in equation 2.

The transformation matrix of extrinsic parameters $\mathbf{P} = [\mathbf{R} \mid \mathbf{t}]$ is a matrix composed of rotation $\mathbf{R}$ and translation $\mathbf{t}$. In order for $\mathbf{P}$ to be a valid transformation, $\mathbf{R}$ must have an orthonormal basis.

Using individual elements $h_{ij} : 1 \leq i, j \leq 3$ of $\mathbf{G}$, the following relations can be obtained:

$$\mathbf{P} = [\mathbf{r_1}, \mathbf{r_2}, \mathbf{r_3}, \mathbf{t},] \tag{3}$$

$$\mathbf{P} = \begin{bmatrix} \alpha \frac{h_{11} - c_x r_{31}}{f_x}, & \alpha \frac{h_{12} - c_x r_{32}}{f_x}, & \mathbf{r}_1 \times \mathbf{r}_2, & \alpha \frac{h_{13} - c_x}{fx} \\ \alpha \frac{h_{21} - c_y r_{31}}{f_y}, & \alpha \frac{h_{22} - c_y r_{32}}{f_y}, & & \alpha \frac{h_{23} - c_y}{fy} \\ h_{31} & h_{32} & & 1 \end{bmatrix} \tag{4}$$

Where c_x, c_y are camera center offset parameters. f_x and f_y are focal distances of the camera. Normalizing factor α must be estimated such that the magnitude of $\mathbf{r_1}$ and $\mathbf{r_2}$ be unitary. An advantage of this decomposition method is that it only requires one view, and camera intrinsic parameters.

Once the decomposition and 3D reconstruction of the tracked plane is obtained, the full object model may be reconstructed. For this work, the model is a cuboid. A cuboid has 6 planar faces and can be defined in terms of the planes that conform it. That is:

$$C = \{\mathbf{P}_i \mid 1 \leq i \leq 6\} \tag{5}$$

Where $\mathbf{P}_i$ refers to the homogeneous transformation matrix for plane π_i from its current referential to camera referential. For all faces in the cuboid C there exist at least one transformation matrix ${}^{i}\mathbf{T}_k$ that maps face referential k to face referential i. This can be expressed as:

$$\forall \pi_i \in C \; \exists \; {}^{i}\mathbf{T}_k \mid \mathbf{P}_i = \mathbf{T}_k \mathbf{P}_k \tag{6}$$

In the case of the cuboid used, these transforms can be estimated by assuming certain dimensions (up to scale factor) and then applying the corresponding rotations and translations. In fact, since there are 6 faces, a total of 30 transforms may be precomputed (5 for each of the 6 faces) so that it becomes straightforward obtaining the transform for face i from face k.

3.3 Face Selection

With the full 3D object reconstruction of the tracked object, the next step consist in determining which face is the most suitable for tracking in subsequent images. For this, the method outlined in [2] is used. This method works by finding the normal of the visible planes that is most aligned to the camera principal axis. In that article it's demonstrated that this is equivalent to finding face i such that:

$$\underset{i}{\operatorname{argmax}} \; l(i) = \{\mathbf{c} \cdot \mathbf{n}_i \mid \forall i \; : \; 1 \leq i \leq 6\} \tag{7}$$

Where $\mathbf{c}$ refers to the center of the tracked object relative to the camera coordinate system. $\mathbf{n}_i$ is a normal vector to plane i. It must be taken into account that the selected face will not always be the one with the most suitable position for tracking.

4 Visual Servoing of Aerial Vehicle

An *Ar.Drone Parrot* quadracopter was tracked and controlled using the presented approach. Since planar faces are required, a cuboid shaped object was adapted and installed on top of it (see figure 5). The 3D object tracking approach that was used allows obtaining a 4×4 transformation matrix that relates camera referential to object referential.

The objective of this research is having an aerial vehicle to execute a given trajectory, provided a set of 3D poses that are part of the path. In the case of the Ar.Drone vehicle, possible manipulations are: *roll*, *pitch*, *yaw* and *vertical speed*. It must be noted that since the vehicle may move freely in all directions, only set-points on translation are required. Yaw control can be used in order to have the vehicle's camera to look at certain places.

For the roll and pitch manipulated variables, a cascade PID control strategy was employed. This is required to have control over the speed at which the vehicle

moves. The input is the desired x-y position. From the difference between the set-point and the measurement, a speed set-point is generated. A second controller takes this input and transforms it to a roll and pitch values. This controller is shown in figure 4.

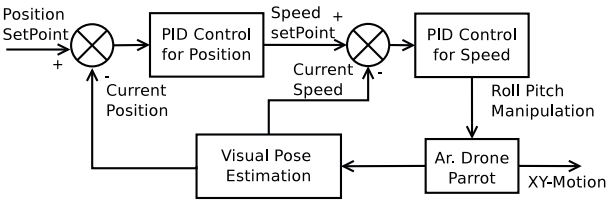


Fig. 2. Control loop for x-y position tracking. The first control loop controls the target speed of the drone. The second loop manipulates the pitch and roll to reach the target speed.

For the roll and pitch controller, the integral component of the speed controller plays a key role when perturbations are present, since in case of strong wind conditions, it will help to reduce the error by increasing the manipulation when the vehicle is having difficulty to move regardless of the pitch and roll manipulation.

For the yaw and vertical speed, simple proportional controllers were used, since built-in orientation and height control is good for allowing soft movements.

5 Experiments

Two exercises were run. First, the UAV was instructed to remain in "hover" state. Since there exist different perturbations on the environment (for example, the pressure effect produced by walls), the vehicle will not remain in that state by itself if no manipulation is applied. Results are shown in figure 3.

For the hover exercise, only one face of the cuboid is tracked, since perturbations are not capable to producing a face switch.

The second exercise consisted in a set of rotations over the yaw axis of the vehicle. Because the starting face is not visible at all stages of the test, the algorithm had to switch faces, consistently with employed face selection strategy. Figure 4 shows the result of this exercise.

Regarding 3D cuboid reconstruction and face selection, current results show that pose estimation presents substantial noise. Error in the reconstruction affects how the cuboid is projected from $\Re^3$ to $\Re^2$. The reconstruction error is related to the plane tracking process and the camera calibration used for homography decomposition.

Face selection could also be validated in the rotation experiment. In figure 5, that corresponds to exercise two, when the vehicle rotated, the face selection

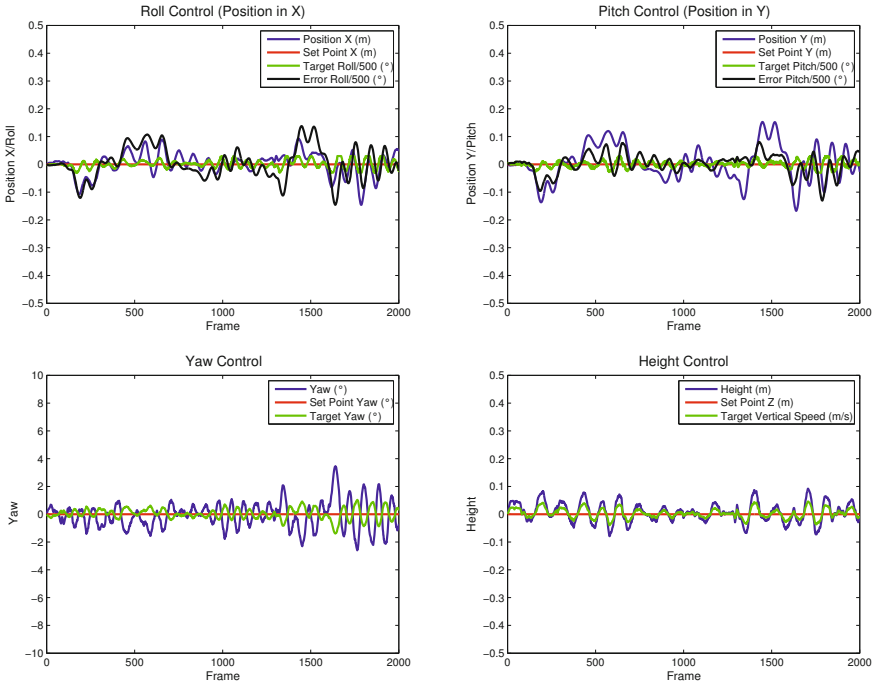


Fig. 3. Control variables and manipulation for the "hover exercise". The drone should remain at a fixed position, but perturbations will affect the control. For this exercise, only one face was tracked.

algorithm selected the face that was more suitable for plane tracking. An implementation note is that the face selection should allow the tracker to completely converge, since using a non convergent solution can produce a race condition, where each time, a worst reconstruction produces a new face switch, and so on, until the tracker diverges.

Experiments confirm that our approach, that uses only one remote camera, is able to produce a stable control.

5.1 Discussion

Vision based UAV control is usually handled via on board cameras [6,9,11,13], requiring landmarks on the floor for pose and position control. These methods also rely on data provided by the IMU to get a complete pose estimation of the vehicle. However we have presented an approach that uses a single remote camera for 3D pose tracking. This gives the possibility of tracking multiple objects with the same camera.

This work can be compared to that presented in [13] where a hover maneuver is presented. A significant difference is that they require a visual stereo system for

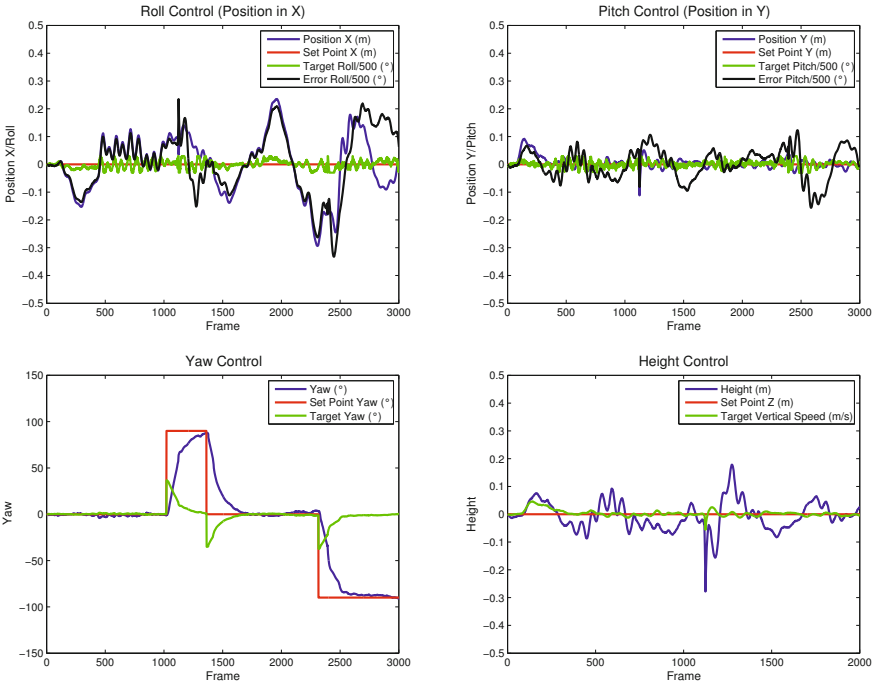


Fig. 4. Control variables and manipulation for the "rotation exercise". The drone should remain at a fixed position while the yaw should reach three provided set-points: 90°, 0° and −90°. It must be noted artifacts present in plots around frames 1220, 1460 and 2420. These are produced when tracked face is switched to a different face.

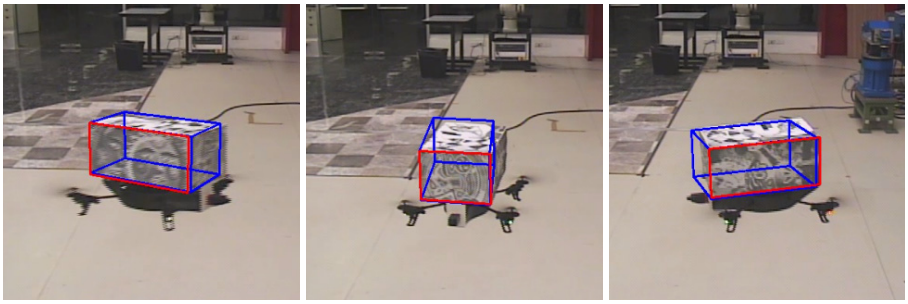


Fig. 5. Sample poses of the Parrot and the corresponding 3D reconstructions (wire-frame). Selected face is marked with red. Reconstructions do not fit exactly to the object. Nevertheless, retro-projections of 3D transforms to image are close enough to be a good starting point for the first iteration of the tracker.

translational dynamics and an IMU for rotational dynamics. Our method shows comparable results in the control while relying solely on the visual feedback provided by the external camera.

It must be mentioned that our 3D tracking approach works at real-time (more than 24 *fps*), clearly better than the proposal of Panin et al. [12] (2 *fps*). A major drawback from our method is that it will be limited to track polygon meshes.

6 Conclusions and Future Work

We have presented a method for visual servoing of a flying drone in a 3D environment. The performance of the method is completely real-time and capable of feeding a control loop that assumes no delay.

Even though the calibration did not produce 3D reconstructions such that, when projected back to image plane were a perfect fit for the pose of the cuboid object, the method was able to handle this small differences during face switching.

It also must be highlighted that only one camera was used. We showed how using only homography decomposition and simple transforms models (in $\Re^3$) it is possible to construct a 3D model of the target object.

An area of opportunity for this research is the fusion of data from drone IMU and other sensors with that of the vision. This could help reducing the noise in the pose estimation and face switching stage.

This work is aligned with our general objective of building automatic 3D trackers. Future work will include the connection of an automatic plane detection method with this approach for doing automatic object tracking.

Acknowledgement. Authors want to acknowledge Consejo Nacional de Ciencia y Tecnología (CONACyT) and e-Robots Research Chair from Tecnológico de Monterrey, Monterrey for supporting this research.

References

1. Baker, S., Matthews, I.: Lucas-Kanade 20 Years On: A Unifying Framework: Part 1. International Journal of Computer Vision 56, 221–255 (2004)
2. Barajas, M., Esparza, J., Gordillo, J.L.: Towards Automatic 3D Pose Tracking through Polygon Mesh Approximation. In: Pavón, J., Duque-Méndez, N.D., Fuentes-Fernández, R. (eds.) IBERAMIA 2012. LNCS, vol. 7637, pp. 531–540. Springer, Heidelberg (2012)
3. Bouchafa, S., Zavidovique, B.: Obstacle Detection "for free" in the C-velocity space. In: 14th IEEE International Conference on Intelligent Transportation Systems (ITSC), pp. 308–313 (2011)
4. Brown, J.A., Capson, D.W.: A Framework for 3D Model-Based Visual Tracking Using a GPU-Accelerated Particle Filter. IEEE Transactions on Visualization and Computer Graphics 18, 68–80 (2011)
5. Cobzas, D., Sturm, P.: 3D SSD Tracking with Estimated 3D Planes. In: The 2nd Canadian Conference on Computer and Robot Vision, pp. 129–134 (2005)

6. Daewon, L., Ryan, T., Kim, H.J.: Autonomous landing of a VTOL UAV on a moving platform using image-based visual servoing. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 971–976 (2012)
7. Manz, M., Luettel, T.: Monocular model-based 3D vehicle tracking for autonomous vehicles in unstructured environment. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 2465–2471 (2011)
8. Malis, E.: Improving vision-based control using efficient second-order minimization techniques. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 1843–1848 (2004)
9. Mondragón, I.F., Campoy, P., Martínez, C., Olivares-Méndez, M.A.: 3D pose estimation based on planar object tracking for UAVs control. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 35–41 (2010)
10. Munozsalinas, R., Aguirre, E., Garciasilvente, M.: People detection and tracking using stereo vision and color. Image and Vision Computing 25, 995–1007 (2007)
11. Ozawa, R., Chaumette, F.: Dynamic visual servoing with image moments for a quadrotor using a virtual spring approach. In: IEEE International Conference on Robotics and Automation (ICRA), pp. 5670–5676 (2011)
12. Panin, G., Knoll, A.: Mutual Information-Based 3D Object Tracking. International Journal of Computer Vision 78, 107–118 (2007)
13. Salazar, S., Romero, H., Gomez, J., Lozano, R.: Real-time stereo visual servoing control of an UAV having eight-rotors. In: 6th International Conference on Electrical Engineering, Computing Science and Automatic Control (CCE), pp. 1–11 (2009)
14. Shao, X., Zhao, H., Nakamura, K., Katabira, K., Shibasaki, R., Nakagawa, Y.: Detection and tracking of multiple pedestrians by using laser range scanners. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pp. 2174–2179 (2007)
15. Tonko, M., Nagel, H.-H.: Model-Based Stereo-Tracking of Non-Polyhedral Objects for Automatic Disassembly Experiments. International Journal of Computer Vision 37, 99–118 (2000)

Third-Eye Stereo Analysis Evaluation Enhanced by Data Measures

Verónica Suaste[1], Diego Caudillo[1], Bok-Suk Shin[2], and Reinhard Klette[2]

[1] CIMAT and the University of Guanajuato, Mexico
[2] The *.enpeda..* Project, The University of Auckland, New Zealand

Abstract. Third-eye stereo analysis evaluation compares a virtual image, derived from results obtained by binocular stereo analysis, with a recorded image at the same pose. This technique is applied for evaluating stereo matchers on long (or continuous) stereo input sequences where no ground truth is available. The paper provides a critical and constructive discussion of this method. The paper also introduces data measures on input video sequences as an additional tool for analyzing issues of stereo matchers occurring for particular scenarios. The paper also reports on extensive experiments using two top-rated stereo matchers.

1 Introduction

Modern applications of stereo analysis require that stereo matchers work accurately on long or continuous binocular input video data. For example, in vision-based driver assistance, those data are recorded for any possible traffic scenario [9]. Robust matchers need to work accurately for various scenarios. In general it is expected that there is no single best matcher; an adaptive selection of a matcher (within a given 'toolbox') appears to be a possible solution.

The third-eye method of [11] provides stereo analysis performance evaluation for long or continuously recorded stereo sequences. For a current application of this method, see [12]. We provide in this paper a critical and constructive discussion of this method, pointing to weaknesses and also outlining ways how to overcome those. Video data measures are used to discuss solutions and to propose ways for a detailed analysis of situations where a stereo matcher fails (and should be improved accordingly), extending our initial discussion of data measures in [10].

For testing, the eight long trinocular stereo sequences of Set 9 on EISATS [4] have been used (each 400 stereo frames long, except the 'People' sequence which is only 234 frames long); see Fig. 1. The tested stereo matchers are *iterative semi-global matching* (iSGM) [7] and *linear belief propagation* (linBP) [10]. Both apply the census transform as the data cost function, and linBP uses a truncated linear smoothness constraint [5]. Both stereo matchers, iSGM and linBP, rank high on the KITTI stereo benchmark suite (`www.cvlibs.net/datasets/kitti/`).

The paper is structured as follows: Section 2 provides used notations and definitions. Section 3 illustrates interesting cases when using the third-eye approach. Section 4 discusses the use of data measures for solving critical cases and for discussing stereo performance more in detail. Section 5 concludes.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 74–83, 2013.

Fig. 1. Examples of frames of the eight sequences in Set 9. Upper row, left to right: sequences called 'Barriers', 'Bridge', 'Dusk', and 'Midday'. Lower row, left to right: 'Night', 'People', 'Queen', and 'Wiper'.

2 Fundamentals

The third-eye method [11] requires that three calibrated cameras record time-synchronized the same scene. In case of Set 9 on EISATS [4], the cameras are placed on a bar on the left, center, and right position behind the windscreen of the *ego-vehicle* (i.e. the car the stereo-matcher is operating in). Two of the images, the center or reference image, and the right or match image, are used to calculate a disparity map by the chosen stereo matching algorithm. The disparity map is used to map all the pixels in the reference image into that position in the left or control image where the pixel value would be visible from the pose of the left camera. This calculated *virtual image* V is then compared with the control image C, for example by using the normalized cross correlation (NCC) index used as a quality measure:

$$M_{NCC}(V,C) = \frac{1}{|\Omega|}\Sigma_{(x,y)\in\Omega}\frac{[V(x,y)-\mu_V][C(x,y)-\mu_C]}{\sigma_V\sigma_C} \tag{1}$$

The domain Ω contains only those pixels which are successfully mapped from the reference image into the domain of the virtual image (i.e. non-occluded pixels); μ and σ represent mean and standard deviation of the corresponding images.

Due to possibilities of a misleading influence of homogeneous intensity regions, [6] suggested to use a further restricted set Ω which only contains pixels locations which are in distance of 10 pixels or less to an edge pixel in the reference image. We use this modified NCC-mask measure, called M_{NCC_Mask}, as our standard measure for the third-eye approach.

Having stereo sequences of length 400 (or 234 in one case) in our test data, the measure M_{NCC_Mask} produces a real-valued function for each used stereo matcher on such a sequence. We also define data measures on input data of one of the cameras (e.g. variance of intensities or of Sobel edge values), or by comparing images recorded by two of the cameras (e.g. M_{NCC} between left and center image). Those data measures also define real-valued functions, and they are motivated as follows:

Homogeneous images are a difficult case for stereo matching, thus we considered the variance of intensity values in reference and match image. Typically those variances of reference and match image are about the same, so we only use the standard deviation $Sigma\_left$ of the reference image.

Stereo matching is supported by having image features such as edges or corners. There are complex edge detectors such as that of [2], or very simple edge detectors such as the Sobel operator. For avoiding a bias introduced by the edge detector we use the simple Sobel operator and measure the standard deviation $Sigma\_Sobel$ of operator values $|C_x| + |C_y|$ on control image C.

An important assumption for stereo matching is that both images are captured in the same environment, with just small changes due to a minor variation in view point or viewing direction. For example, see [3] where this is discussed for stereo vision in astrophysics. We decided to use the above NCC measure between reference and match image as our third data measure $NCC\_leftright$.

We intend to compare two of such real-valued functions having the same domain, not in a rigorous sense of direct analysis, but with respect to the curves 'behavior', such as the distribution of local minima or maxima. For discussions about special kinds of, or comparisons between functions, see, for example, [1,3,8,14]. One option is to study or compare the derivatives of the functions. But, working in discrete domains, that implies a need to choose a neighborhood of some size and a method for approximating derivatives.

Thus we decided to keep one function f fixed as a reference, and to apply a transformation to the other functions g which allows us to define an analytical distance between the new function g_{new} and f, thus defining an alternative relationship between functions. The defined distance will not be a metric in the mathematical sense because we do not aim at symmetry, and the distance between two different functions (e.g. two constant functions with different values) can be zero when applying our distance measure.

Let μ_f and σ_f be the mean and standard deviation of function f. Given are two real-valued functions f and g with the same discrete domain and non-zero variances. Let

$$\alpha = \frac{\sigma_g}{\sigma_f}\mu_f - \mu_g \quad \text{and} \quad \beta = \frac{\sigma_f}{\sigma_g}$$
$$g_{new}(x) = \beta(g(x) + \alpha) \tag{2}$$

As a result, g_{new} has the same mean and the same variance as function f. Now we define our distance function in the common L_1 way, as, for example, described by [1]:

$$d_1(f_1, f_2) = \int |f_1(x) - f_2(x)| \, \mathrm{d}x \tag{3}$$

Our distance is then defined by $d(f, g) = d_1(f, g_{new})$. Indeed, this distance measure is not symmetric. However, we identify the value of $d(f, g)$ with the *structural similarity* between both functions: lower values for $d(f, g)$ mean that g is *structurally close* to f.

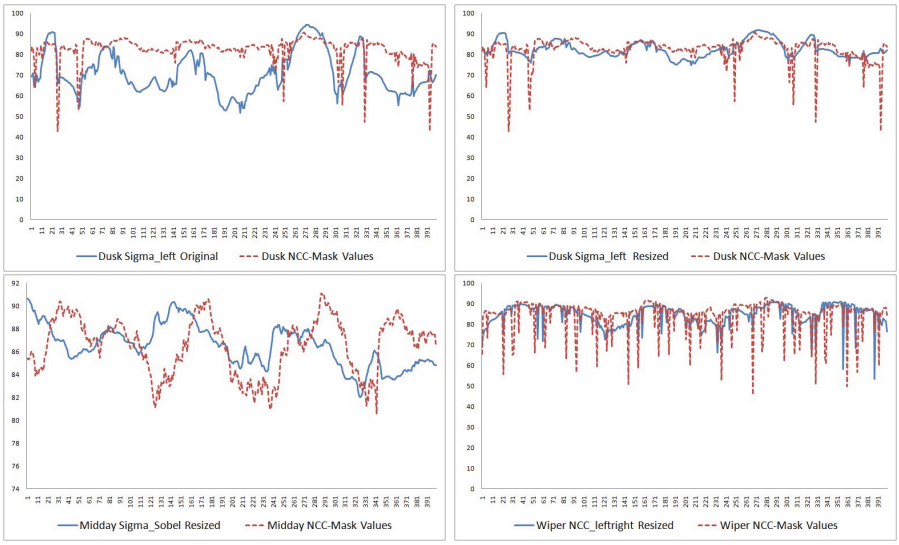


Fig. 2. Illustration of four pairings of functions

Figure 2 shows pairs of real valued functions on a discrete domain. Function *Sigma_left* represents the standard deviation of intensity values in the reference image, and function *Sigma_Sobel* represents the standard deviation of Sobel edge values for the reference image. Function *NCC_leftright* represents the NCC-measure when comparing the reference and the match image of the sequence. Function *NCC_Mask* is the defined standard measure for the third-eye approach.

The two functions in the upper left image of Fig. 2 have different means and different variances. The upper right shows both functions after Sigma_left was scaled to have the same mean and the same variance as NCC_Mask. The lower row shows two already scaled pairings of functions. Subjective inspection shows inconsistencies between both functions in the lower left, but 'fairly good structural similarity' for both functions in the lower right (when zooming into the figure). Obviously, those subjective inspections can also be replaced by an analytical analysis by calculating the sum of absolute differences in function values, as specified by Equ. (3) and by our distance measure $d(f, g) = d_1(f, g_{new})$. In general, if a distance value $d(f, g)$ is less than half of the standard deviation σ_f used for scaling then both functions are considered to be structurally similar.

3 Discussion of Third-Eye Results

Concluding from visual inspections of calculated disparity maps for all the eight test sequences (and many sequences for earlier projects) we see a very close correspondence between calculated values of the NCC-Mask measure and the actual performance of studied stereo matchers. So far the assumption was that

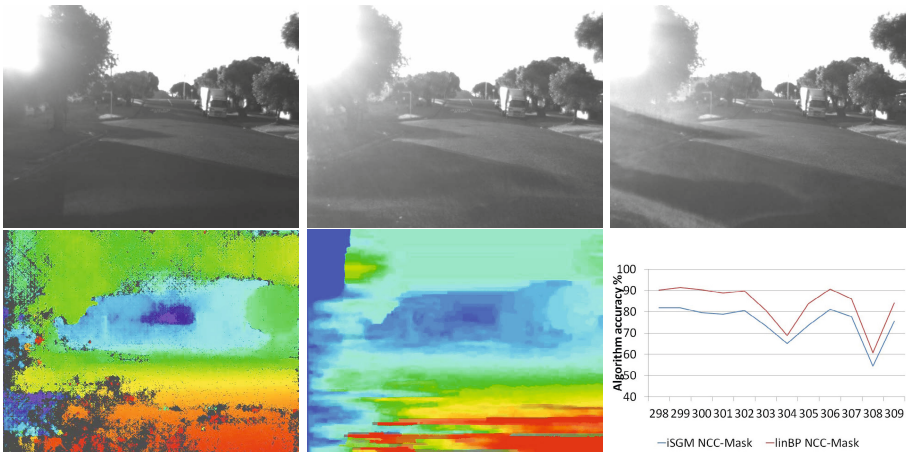


Fig. 3. Dusk sequence. Upper row, left to right: Control image, reference image, and match image. Lower row, left to right: Disparity maps for iSGM, linBP, and NCC-Mask plot for eleven frames from 298 to 309.

a value of the NCC-Mask measure below the 70% mark is an indication for a failure. However, in the current study we refined this threshold: we recommend to define it in dependency of data measure values for the given stereo frame, for example on the standard deviation of the gradient in the reference image.

As a first case we show a situation where a failure is properly detected. We present Frame 304 from the 'Dusk' sequence; see Fig. 3. Temporarily around this frame, sun strike creates a difficult lighting situation. There are no considerable changes in occluded pixels between the three camera views. Therefore, the third-eye evaluation performance is not considerably affected by occluded pixels. Figure 3 shows the color-encoded disparity maps of iSGM and linBP for this frame. Visually we observe that the performance of both matching algorithms is not good. Both NCC-Mask measures have local minima of about 70% at Frame 304. The third-eye approach works fine in this case.

In the same sense, the NCC-Mask measure also indicates good or bad performances as illustrated in Fig. 4. The upper row illustrates depth maps resulting from iSGM. The left map is for Frame 176 in the sequence 'Queen' showing an excellent result; the right map is for Frame 382 in the sequence 'Dusk' showing a failure. The lower row shows depth maps for linBP for those two frames. However, here, linBP performs not well for Frame 176 in 'Queen', but better than iSGM for Frame 382 in 'Dusk'. The third-eye approach also works fine in general for indicating the 'current winner' (of all participating stereo matchers) for a given situation. There is no all-time winner so far for the tested stereo matchers.

The third-eye approach provides a summarizing single value for each frame, and these summarizing values may not correspond to subjective visual evaluations. From the appearance of the depth maps, iSGM performs better in

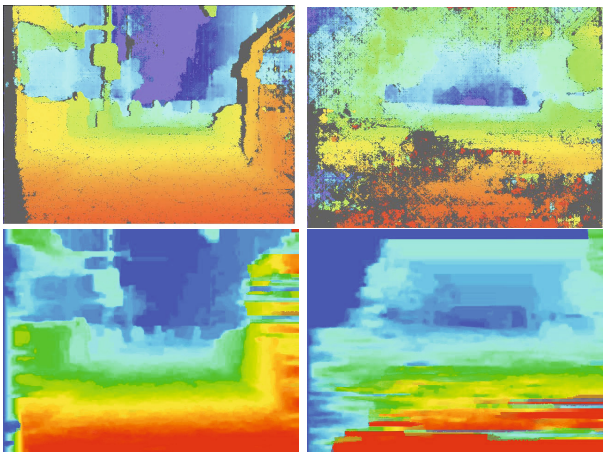


Fig. 4. Depth maps provided by iSGM (upper row) and linBP (lower row), for Frame 176 of 'Queen' (left column) and for Frame 382 of 'Dusk' (right column)

detecting depth discontinuity edges, thin vertical structures, or other rapid changes in depth. However, linBP is often performing better on large homogeneous areas. See Fig. 5 for plots of NCC-mask measures for sequences 'Barriers', 'People', 'Queen', and 'Night'. In nearly all of the shown 1,434 frames, the value of linBP is above that of iSGM. (For plots for the remaining four sequences of Set 9 of

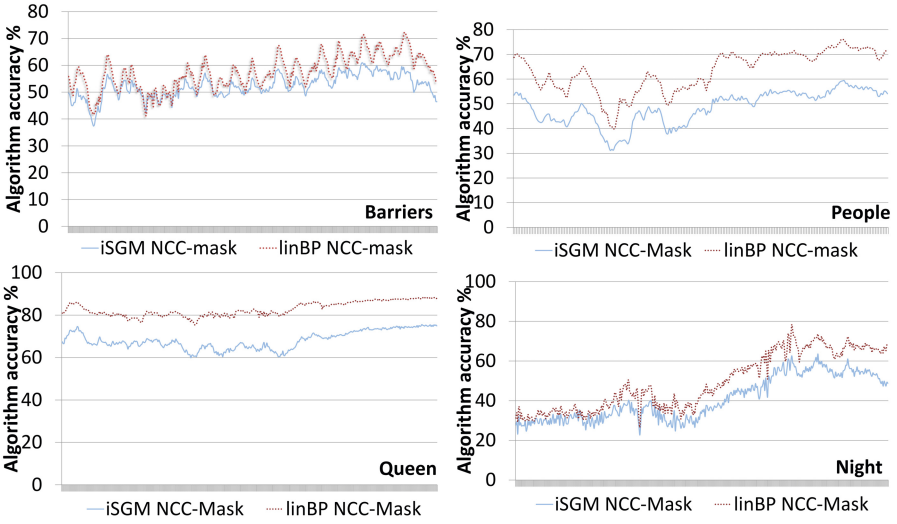


Fig. 5. Plots of the NCC-Mask measure for iSGM and linBP for four of the eight sequences of Set 9 of EISATS

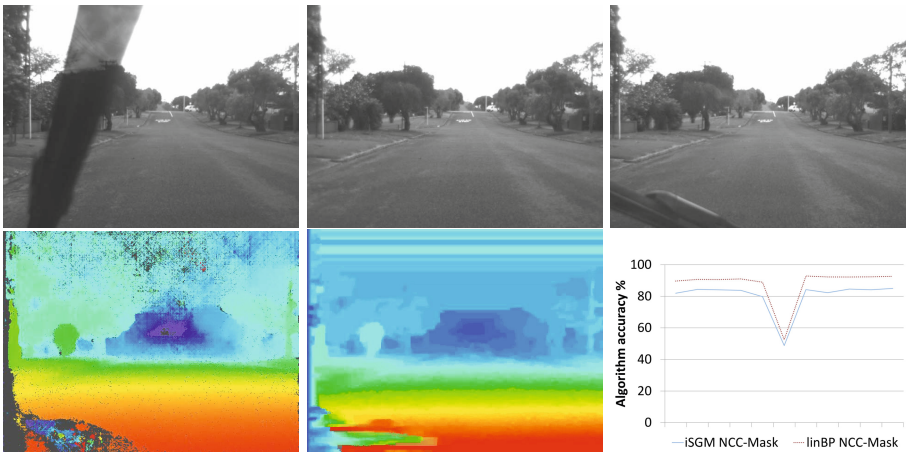


Fig. 6. Wiper sequence. Upper row, left to right: Control image, reference image, and match image. Lower row, left to right: Disparity maps for iSGM, linBP, and NCC-mask plot for eleven frames from 272 to 282.

EISATS, see [10].) The standard deviations of NCC-Mask for linBP, and the distance to the scaled NCC-Mask for iSGM are (5.22, 2.58) for 'Barriers' (2.58 is about 49% of 5.22), (0.87, 0.77) for 'Bridge' (88%), (3.77, 1.91) for 'Dusk' (50%), (1.61, 1.16) for 'Midday' (72%), (13.03, 3.15) for 'Night' (24%), (7.14, 4.27) for 'People' (59%), (2.96, 2.24) for 'Queen' (75%), and (4.86, 1.55) for 'Wiper' (32%). According to our 50% rule defined at the beginning of the next section, we consider both NCC-Mask curves as being structurally similar for 'Barriers', 'Dusk', 'Night', and 'Wiper', on the other four sequences both stereo matchers behave 'qualitatively different'. This analysis is our first important contribution to the application of the third-eye approach.

A second important comment about the third-eye approach: summarizing numbers do have limitations when interpreting. The more accurate detection of 3D details by iSGM compared to linBP is not (!) expressed in the obtained number, but the NCC-mask curves express in general accurately the ups and downs in a matcher's performance.

Finally, as a third important comment, there are cases where the measure provided by the third-eye approach does not coincide with what we see in disparity maps of a stereo matcher due to differences between control and reference image. In order to illustrate this phenomena, we show as an example a trinocular frame of the 'Wiper' sequence. In Fig. 6 we have the disparity maps given by iSGM and linBP for Frame 277 of this sequence and the local plot of the NCC_Mask measures. From the shown disparity maps we would expect a high NCC_Mask value, but there are local minima in the functions. If we observe the complete information of this particular frame given by all the three cameras (see Fig. 6, upper row) we notice that the control image differs significantly from reference and match images, not by different lighting (as for the example in Fig. 3) but

due the wiper position in the control image. Therefore, NCC_Mask gives us a low value. In the 'Wiper' sequence we observe the same effect for Frames 31, 185, and 216, where a low NCC_Mask is incorrectly indicating a failure of the stereo matcher.

Regarding the third comment, a simple solution is to use the NCC measure for quantifying similarity between the control and the reference image; if the similarity value goes below a defined threshold then the third-eye NCC_Mask value is insignificant. Typically this is only true for a very short time (as for the moving wiper).

4 Analysis Using Data Measures

We use the NCC_Mask measure for iSGM as the reference function and compare data measures with this function using our distance definition. Table 1 shows those distances, also listing the standard deviation $\text{Sigma}_{NCCMask}$ of the NCC_Mask measure for comparison. *Our 50% rule*: If a distance is below 50% of this standard deviation then we call the functions *structurally similar*.

Table 1. Distance values between NCC_Mask (for iSGM) and data measures

	Barriers	Bridge	Dusk	Midday	Night	People	Queen	Wiper
Sigma_left	2.28	1.91	3.69	1.52	9.27	6.90	3.78	5.16
Sigma_Sobel	2.25	1.81	4.24	2.50	9.96	9.03	3.49	6.53
NCC_leftright	2.75	2.62	4.24	**1.26**	10.69	5.21	3.32	4.59
$\text{Sigma}_{NCCMask}$	2.85	2.24	6.09	2.47	11.72	7.44	5.35	7.79

The closest to 50% is NCC_leftright for 'Midday'. This shows that structural similarity is low between NCC_Mask and the used three data measures in general. It appears that the complexity of video data for stereo matching cannot simply be estimated by just using a summarizing distance value for one of those three global data measures for a whole sequence, showing (in our case) 400 frames of one particular situation.

A more refined approach is to study the graphs of the functions of the data measures, as we already did for NCC_Mask in the third-eye approach. Scaled functions are shown in Fig. 7, together with NCC_Mask (for iSGM) which was kept constant for scaling. Sequences 'Midday' and 'Queen' have a very low variance, and 'Dusk' and 'Wiper' represent special challenges for stereo matching apparent by rapid drops in NCC-Mask values. Values of these curves show locally some kind of correspondence with NCC_Mask, and sometimes differences. Correspondences may explain a fail of the stereo matcher, and differences may provide hints how to improve the stereo matcher at this particular place of the stereo sequence.

A more refined approach is to use the local variance of data measures (e.g. for six frames backward, the current frame, and six frames forward). We demonstrate this for a case where iSGM failed, and discuss the data measures only in the local context of 13 frames. See Fig. 8.

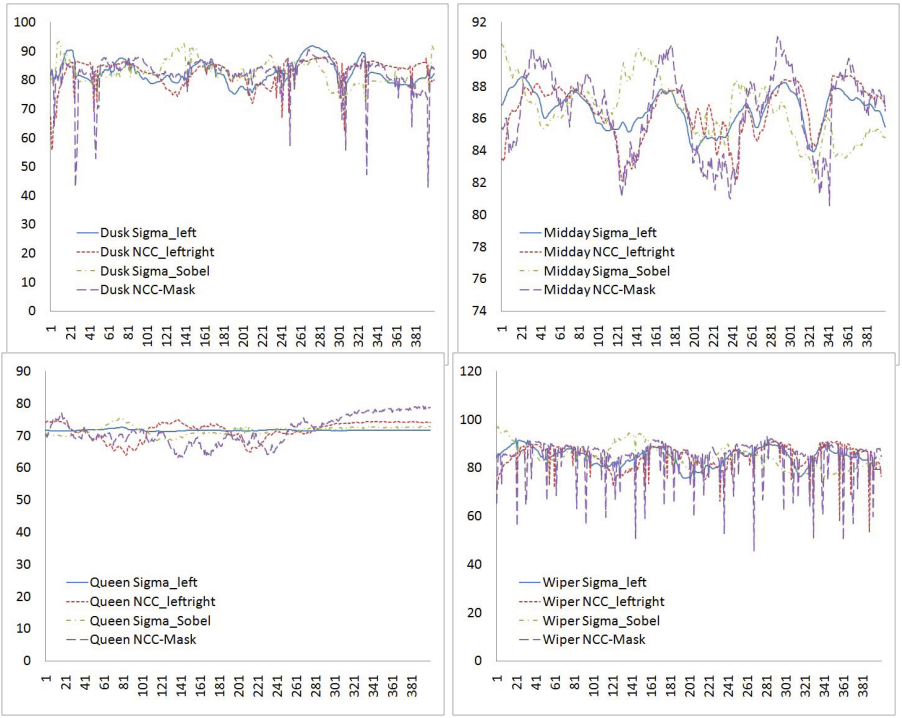


Fig. 7. Scaled curves for direct visual comparison

We analyze the situation around Frame 330 of the 'Dusk'. Not only the NCC_Mask values (see figure on the left), but also the appearance of iSGM disparity map for this frame indicates a fail in stereo matching. The figure shows on the right the local variances of the data measures and of NCC_Mask. They all go up around Frame 330, but are nearly constant before and after.

This illustrates a general observation: at places where a fail in stereo matching occurred, typically also one, two or all three of the data measures showed a large local variance.

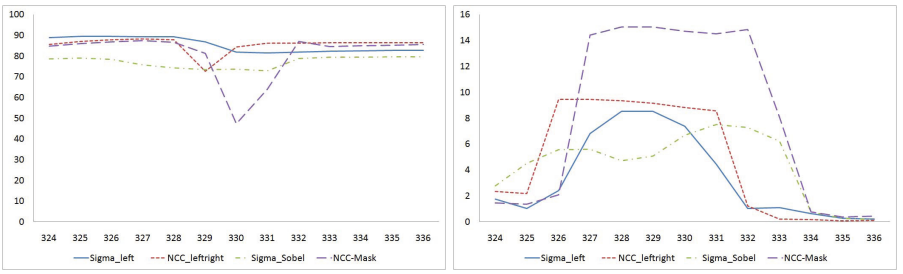


Fig. 8. Functions on the 'Dusk' sequence between Frames 324 to 336. Left: comparison between scaled data measures and iSGM NCC_Mask. Right: comparison of the local variance of data measures and iSGM's NCC_Mask.

5 Conclusions

The third-eye approach is a valuable tool for analyzing stereo matchers on long sequences or for continuous recording. Detections of a 'fail' are important for implemented systems. This paper provided a critical discussion how to detect such 'fails', and how to use data measures for a more detailed analysis at places where a 'fail' was detected. Such a data analysis might lead to ideas how to improve a given stereo matcher for a particular situation. Applied measures (here: NCC and NCC_Mask) should be studied further for comparative evaluations. For example, iSGM appears to provide better occlusion edges compared to linBP, but this effect is not yet measured by NCC_Mask.

Acknowledgement. The authors thank Simon Hermann and Waqar Khan for providing the executables for iSGM and linBP, respectively, and Sandino Morales for providing the sources for the third-eye approach and comments on the paper.

References

1. Baeza-Yates, R., Valiente, G.: An image similarity measure based on graph matching. In: Proc. String Processing Information Retrieval (2000)
2. Canny, J.: A computational approach to edge detection. IEEE Trans. PAMI 8, 679–698 (1986)
3. Drayer, B.: Image comparison on the base of a combinatorial matching algorithm. In: Mester, R., Felsberg, M. (eds.) DAGM 2011. LNCS, vol. 6835, pp. 426–431. Springer, Heidelberg (2011)
4. EISATS benchmark data base. The University of Auckland (2013), `www.mi.auckland.ac.nz/EISATS`
5. Felzenszwalb, P.F., Huttenlocher, D.P.: Efficient belief propagation for early vision. Int. J. Computer Vision 70, 41–54 (2006)
6. Hermann, S., Morales, S., Klette, R.: Half-resolution semi-global stereo matching. In: Proc. IEEE Intelligent Vehicles Symposium, pp. 201–206 (2011)
7. Hermann, S., Klette, R.: Iterative semi-global matching for robust driver assistance systems. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part III. LNCS, vol. 7726, pp. 465–478. Springer, Heidelberg (2013)
8. Kalinic, H., Loncaric, S., Bijnens, B.: A novel image similarity measure for image registration. In: Proc. Image Signal Processing Analysis, pp. 195–199 (2011)
9. Klette, R., Krüger, N., Vaudrey, T., Pauwels, K., Hulle, M., Morales, S., Kandil, F., Haeusler, R., Pugeault, N., Rabe, C., Lappe, M.: Performance of correspondence algorithms in vision-based driver assistance using an online image sequence database. IEEE Trans. Vehicular Technology 60, 2012–2026 (2011)
10. Khan, W., Suaste, V., Caudillo, D., Klette, R.: Belief propagation stereo matching compared to iSGM on binocular or trinocular video data. MItech-TR-82, The University of Auckland (2013)
11. Morales, S., Klette, R.: A third eye for performance evaluation in stereo sequence analysis. In: Proc. Computer Analysis Images Patterns, pp. 1078–1086 (2009)
12. Ranftl, R., Gehrig, S., Pock, T., Bischof, H.: Pushing the limits of stereo using variational stereo estimation. In: Proc. IEEE Intelligent Vehicles Symposium, pp. 401–407 (2012)
13. Rosin, P., Ellis, T.: Image difference threshold strategies and shadow detection. In: Proc. British Machine Vision Conf., pp. 347–356 (1995)
14. Thiyagarajan, M., Samundeeswari, S.: A new similarity measure for image segmentation. Int. J. Computer Science Engineering 02, 831–835 (2010)

Counting Pedestrians in Bidirectional Scenarios Using Zenithal Depth Images

Pablo Vera, Daniel Zenteno, and Joaquín Salas*

Instituto Politécnico Nacional
Cerro Blanco 141, Colinas del Cimatario, Querétaro, México, 76090
pvera@ipn.mx, ezentenoj1100@alumno.ipn.mx, jsalasr@ipn.mx

Abstract. In this document, we describe a people counting system that can precisely detect people as they are seen from a zenithal depth camera pointing at the floor. In particular, we are interested in scenarios where there are two preferred directions of motion. In our framework, we detect people using a Support Vector Machine classifier, follow their trajectory by modeling the problem of matching observations between frames as a bipartite graph, and determine the direction of their motion with a bi-directional classifier. We include experimental evidence, from four different scenarios, for each major stage of our method.

1 Introduction

Counting automatically the number of people passing a specific point is a function of paramount importance in applications such as surveillance, monitoring, and interaction between humans and machines. For instance, imagine a civil protection situation taking place in a building: People continuously enter and leave when suddenly an alarm sounds, indicating that the building must be evacuated. One can imagine how useful it is for people in charge of the evacuation procedure to query an automated monitoring system to figure out how many people are still in the building and in what areas.

In this document, we describe a people counting system that can precisely detect people as they are seen from zenithal depth cameras pointing downwards. There are a number of advantages of this configuration, including the fact that people are less likely to occlude one another and that privacy may be better protected. In particular, we are interested in scenarios where there are two preferred directions of motion, such as hallways, where people move primarily in two opposing directions, or entrances, where pedestrians pass in or out. In this paper, we introduce the detection of people in depth images using zenithal cameras. This allows us to develop a robust counting system where all entities except people are readily discarded.

In our method, we detect people, follow their movement, and determine their direction of motion. Our people detector is based on an application of Dalal and Triggs' method of the histograms of oriented gradients[5]. Then, we construct *tracklets* (chronological sequences of observations) of people by matching observations, which

* Thanks to Cole Shaw and Paul Riley for their thoughtful review of the manuscript. Also, thanks to Omar Rodríguez for installing the cameras used in this work. This research was partially supported by SIP-IPN through grant number 20121642.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 84–93, 2013.

include appearance and space-time information, in a bipartite graph. Finally, tracklets are classified by modeling their normalized destination and defining a quadratic classification surface[19].

The rest of the document is organized as follows (see Fig. 1). In §2 we present a survey of the reported research related to counting people, with particular emphasis on the use of zenithal cameras. Then, as the measurement of the height will be very important in later stages of our method, in §3, we describe how we obtain a geometric description of the floor underneath the camera. Next, we describe how the people detector is implemented in §4, and how the people tracklets are constructed in §5. The direction classifier is introduced in §6. In §7, we assess the performance and show some results from the implementation of the methods described. Finally, we conclude and outline possible future applications of this research.

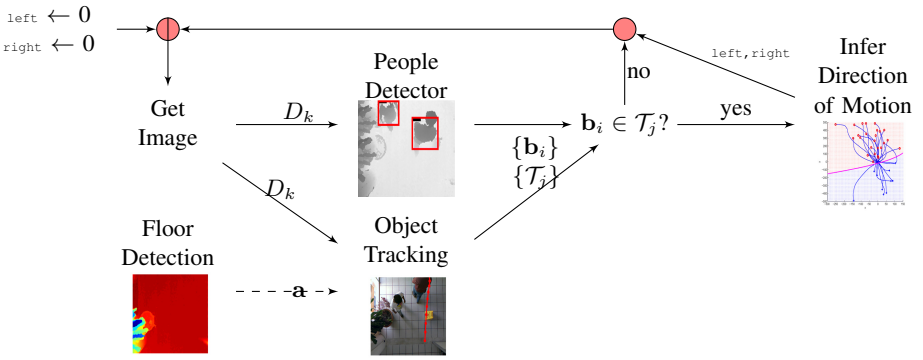


Fig. 1. Solution Global Scheme. By computing the floor parameters, $\mathbf{a}$, it is possible to normalize the measurements of height. Thus, each time a new image D_k is read, there are two tasks to solve. One is to track the moving objects and produce tracklets $\{\mathcal{T}_j\}$. The other is to detect people, which will eventually give a set of bounding boxes $\{\mathbf{b}_i\}$. Whenever the tracklet contains a person, the counters `left` and `right` are updated depending on the direction of motion.

2 Related Work

Due to the considerable number of applications, ranging from entertainment to security, the research efforts aiming to produce reliable and fast people counters has been extensive. Nowadays, there are two main directions of research[14]: Pedestrian detection and tracking-based methods, and feature regression-based methods. In the former, pedestrians are singled out [2,24,11] and the trajectory of each individual is known. In the latter, the problem is framed as a classification one. During learning, an observed set of feature descriptors, such as edges and texture, is correlated with the number of people present. During operation, the feature descriptors are classified and the estimated number of people in the scene is obtained as a result, preserving privacy to some extent [1,12].

In our case, we focus our attention on those applications where a top view image is considered better because there are either fewer occlusions or because there are some privacy concerns that are of prime importance. For other applications where the interest is primarily frontal view images, the interested reader is referred to the reviews of Enzweiler[6] and Gavrila or Geronimo *et al.*[9], where the former places more emphasis on monocular vision and the latter on driver assistance systems. Raheja *et al.*[16] review some of the techniques commonly used in the computer vision community to count people. As a general rule, background models[22] are constructed, and the objects of interest are found via background subtraction. One key component of our approach is that people detection is done directly from depth images, extending the method proposed by Dalal and Triggs[5] to detect people in frontal-view, intensity images. However, other classification-based methods may be suitable. For instance, Haubner *et al.*[10] propose a system to detect body parts above and around a tabletop setup using a depth camera. Their work is based on the frontal view people detection framework proposed by Shotton *et al.*[20]. Importantly, the subjects are dressed with special clothes where colors signal specific portions of the body. Thus the head is covered with a red mask, the neck is yellow, the shoulders are green, the main trunk is violet, and so on. Depth images have been used previously to process zenithal images, in the form of stereo pairs. For instance, Yahiaoui *et al.*[23] detect people by dividing the observed depth map into intervals and applying a morphological operator to seek circular shapes representing the head. To our knowledge, this document reports the first counting system based on top-view depth images that uses a classifier to detect people.

3 Floor Detection

To use the height as a descriptor of the objects we observe, we normalize our measurements and express them with respect to the scenario floor, which is assumed to be flat. To that end, we rotated the 3D points computed from the depth images in order that the z-axis would coincide with the floor's normal orientation. The floor's plane is defined by the equation: $ax + by + cz = d$, where $\mathbf{x} = (x, y, z)^T$ are the coordinates of a point belonging to the plane. An appropriate scale factor can be obtained to make $(a, b, c)^T$ a unitary vector normal to the plane. Then, the rotation matrix $\mathbf{R}$ for rotating the 3D point cloud will satisfy

$$\begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} = \mathbf{R} \begin{bmatrix} a \\ b \\ c \end{bmatrix}. \tag{1}$$

In our case, $\mathbf{R}$ is defined by two angles, α and β, corresponding to rotation over the y and x axes, as $\begin{bmatrix} \cos\alpha & \sin\alpha\sin\beta & -\sin\alpha\cos\beta \\ 0 & \cos\beta & \sin\beta \\ \sin\alpha & -\cos\alpha\sin\beta & \cos\alpha\cos\beta \end{bmatrix}$.

Assuming that the floor either is the only plane or the plane with the largest area visible in the image, α and β can be determined as

$$\arg\max_{\alpha,\beta} N_{\max}, \tag{2}$$

where $N_{\max}$ is the highest value of frequency on a histogram of the z value of the 3D points after being rotated by $\mathbf{R}$. A large value of $N_{\max}$ indicates that the points on the floor are well aligned to have roughly the same z value.

We solved (2) using the Downhill Simplex method[15]. To avoid local minima during the minimization process, we used an initial simplex with one of the vertices having values of α and β computed with the RANSAC method[8]. This computation is done by selecting sets of 3 points at random, computing the parameters of the plane which includes each set of points, and then selecting the plane for which the maximal $N_{\max}$ is obtained, where α and β are computed using

$$\alpha = \arcsin a, \text{ and } \beta = \arcsin \frac{b}{-\cos \alpha}. \tag{3}$$

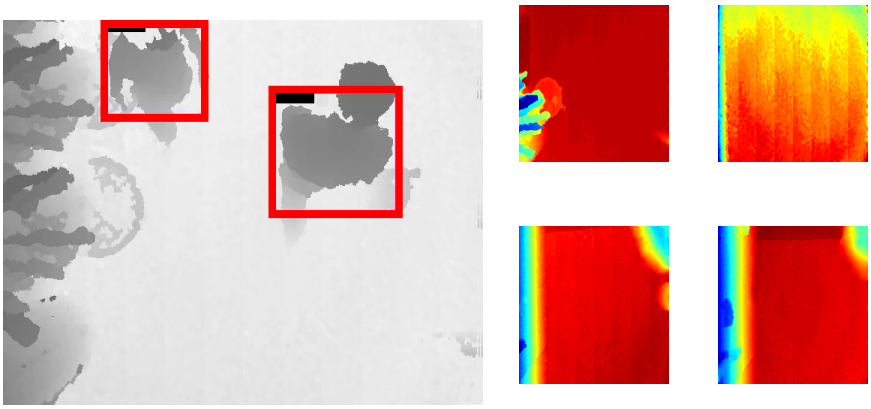
We added the Downhill Simplex optimization stage in conjunction with the RANSAC because it results in a better solution to (2). Besides, the computation complexity is low and the estimation is done offline, as the rotation matrix is computed only once after the camera is fixed.

4 People Detection

In [5], Dalal and Triggs introduced a people detector model for intensity-based images and people in upright positions. In their approach, a candidate was described in terms of features extracted from an image that is divided into same-size overlapping blocks. The blocks themselves are divided equally in same-size cells, and the amount of overlap between blocks corresponds to the size of the cell. The image features Dalal and Triggs use are based on histograms of oriented gradients extracted from the cells, weighted by the magnitude of the gradient in the block. For classification, they used a Support Vector Machine (SVM)[4]. In their method, the candidates in the image were sought using a pyramidal search.

As the intensity represents properties of the blending of many factors including the light sources, the materials of the surfaces and their geometrical structure, and the camera operating function, some researchers [21] have tested successfully a research hypothesis stating that depth information will reflect more clearly the structural properties of the objects being characterized. However, these experiments have been done on images of people taken from frontal view cameras. In the problem of counting the number of people that pass by, frontal view images may result in occlusions that could compromise the objective. In what follows, we describe how we have adapted the Dalal and Triggs methodology to detect people using zenithal depth cameras.

We define a standard size bounding box for people detection of 96×96 `pixels`. The size was chosen by calculating an average over 993 true positive samples and approaching the resulting size to suitable cell and block elements. The sample selection for training is illustrated in Figure 2. As in [5], the features were extracted by computing the gradient over cell structures, which corresponds to neighborhoods of 8×8 pixels. The orientation of the gradient was clustered into nine-bin histograms. The frequency was weighted using the magnitude of the gradient blocks, of 2×2 cells. That is, a person is described by a feature vector $\mathbf{x}$ of size 1089×1.



(a) Positive sample (b) Scenarios without people

Fig. 2. Creating samples for training the people detector. Illustration of some positive samples and scenarios from where the negative samples were obtained to train the classifier. Samples were created by selecting manually or randomly bounding boxes on scenarios with people and without people, respectively, by visual inspection of the images.

5 Tracking

As activity develops below the camera, some objects are detected in the scene. Let $\{O_{j,1}, \ldots, O_{j,m}\}$ correspond to the m objects detected during frame j. In our case, we are interested in inferring where the objects that were detected during frame i have moved in frame j. This tracking problem has been the subject of a large number of papers (for a recent review, please see[18]). We are particularly interested in the problem where the inference process occurs between consecutive frames, and where, as consequence of the observations made, we have appearance and temporal-location information. That is, let a particular observation $O_{j,a} = (h_{j,a}, \mathbf{x}_{j,a}, t_{j,a})$ consist of the location $\mathbf{x}_{j,a}$ and time $t_{j,a}$ where its maximum height $h_{j,a}$ was observed. Just as in a number of other articles[3,17], we model the problem of matching observations in frame j with observations in frame $j+1$ as a complete bipartite weighted graph $G = (O, E, P)$. The vertices $O = \{O_j, O_{j+1}\}$ are divided between the observations made in the frame j and the one that follows it. The set of edges $E = \{e_{j,a}^{j+1,b}\}$ represents the hypothesis that two particular observations correspond to the same person. And the weight $P = \{p_{j,a}^{j+1,b}\}$ represents the likelihood of a particular hypothesis about the correspondence of two observations.

In our approach, we express the weight between two observations as

$$p_{j,a}^{j+1,b} = 1 - p(h_{j,a}, h_{j+1,b}) \cdot p(\mathbf{v}_{j,a}^{j+1,b}) \cdot p(t_{j,a}, t_{j+1^b}), \tag{4}$$

where $p(h_{j,a}, h_{j+1,b})$ is defined as

$$p(h_{j,a}, h_{j+1,b}) = e^{-\alpha |h_{j,a} - h_{j+1,b}|}, \tag{5}$$

(a) (b) (c)

Fig. 3. Trackelet construction. The dots in the line correspond to the highest point positions. The illustration is done in color images but the process was done with the depth images. This simple strategy fails to count pedestrians when there are no people, as in (c). Therefore, there is a need to couple it with a people detector.

and α represents the weight for how likely it is to observe heights h_j^a and h_{j+1}^b; correspondingly $p(\mathbf{v}_{j,a}^{j+1,b})$ is defined as

$$p(\mathbf{v}_{j,a}^{j+1,b}) = e^{-\beta\left|\frac{\mathbf{x}_{j,a}-\mathbf{x}_{j+1,b}}{t_{j,a}-t_{j+1,b}}\right|^n}, \tag{6}$$

and β is the weight associated with observing a certain velocity $\mathbf{v}_j^{a,b}$ between two observations; then, $p(t_{j,a}, t_{j+1^b})$ is defined as

$$p(t_{j,a}, t_{j+1,b}) = e^{-\gamma|t_{j,a}-t_{j+1,b}|^n}. \tag{7}$$

Here, γ is the weight corresponding to observing the same object at times $t_{j,a}$ and t_{j+1}^b. Note that n in (6) and (7) has the purpose of establishing a threshold on the observable average velocity and time values. That is, the larger the value of n, the steeper it is the slope of (6) and (7) around the value of β and γ, respectively. In practice, this has the effect of accepting low values but being critical for values larger than β or γ. Also, note that the third term in (4) is introduced to avoid accepting observations that spread in time.

Now, the problem is to find a match of observations corresponding to consecutive frames at minimal cost. Although other more sophisticated methods could be used, given the usually reduced size of the graphs, we have used the Hungarian algorithm[13]. Illustrations of actual tracklets are presented in Fig. 3.

6 Bi-directional Classifier

In our method, a tracklet $\mathcal{T}^k = \{\mathbf{t}_{j+1}^k, \ldots, \mathbf{t}_{j+m}^k\}$ corresponds to a person when the detector responds positively to a certain number of observations within the tracklet. Given $\mathbf{b}$ as the center of the person's detected bounding box, and $\mathbf{t}_{j+n}^k$ as the position of the tracked object, the observation is said to correspond to the detection whenever $\left\|\mathbf{t}_{j+n}^k - \mathbf{b}\right\| < \tau$, for a predefined value of τ.

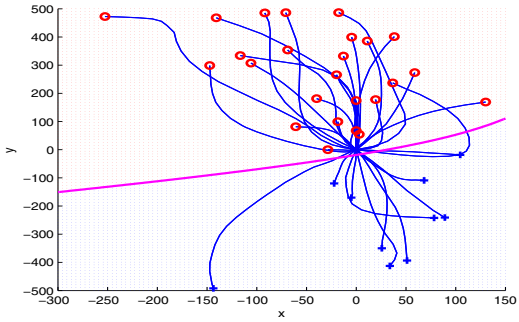


Fig. 4. Classification of the direction of motion. The trajectories are normalized to start at $(0, 0)$. The end point is circled or crossed for the trajectories going in opposite directions. The curved line crossing horizontally corresponds to the decision surface.

In our problem, we want to classify a person's trajectory as going in one of two opposite directions. Thus, given a trajectory $\mathcal{T}^k$, we want to classify it in either one of two disjoint sets $\mathcal{T}_l$ and $\mathcal{T}_r$, which represent opposite directions. To that end, we define centered trajectories as $\overline{\mathcal{T}}^k = \{\overline{\mathbf{t}}^k_{j+1}, \ldots, \overline{\mathbf{t}}^k_{j+m}\}$, where $\overline{\mathbf{t}}^k_{j+n} = \mathbf{t}^k_{j+n} - \mathbf{t}^k_{j+1}$, for $n = 1, \ldots, m$. The problem now is to define a surface to distinguish between classes. Although more sophisticated methods can be applied, in our approach we fitted multivariate normal densities with covariance estimates computed by class, resulting in a decision surface of the form $f(\mathbf{x}) = \mathbf{x}^T\mathbf{A}\mathbf{x} + \mathbf{x}^T\mathbf{b} + c$,

$$D(\overline{\mathcal{T}}^k) = \begin{cases} \mathcal{T}_l & f(\overline{\mathbf{t}}^k_{j+m}) < 0, \\ \mathcal{T}_r & \textit{otherwise}, \end{cases} \tag{8}$$

where $\mathbf{A}$, $\mathbf{b}$, and c correspond to the quadratic, linear, and constant coefficients computed as in [19].

7 Experimental Results

We implemented and tested the methods described in this article in four different scenarios. To obtain the depth images, we used four Microsoft Kinects. To grab our images and process the results we used a variety of software tools including OpenCV, OpenKinect, Linux, Matlab, and Eclipse.

For the people detector, we collected 993 positive images and 1,038 negative images, of which 20% were used for testing and the rest for training the SVM classifier (see Figure 2). To assess the performance of the classifier, we constructed a *precision-recall* curve where the SVM margin was varied between -1 and 1. Note that precision is related to the fraction of detections that are correct, whereas the recall is the fraction of detections out of the possible detections. The result is shown in Fig. 5. As customary, we follow the practice at the Pascal Challenge[7], where a detection is declared when the following condition is met

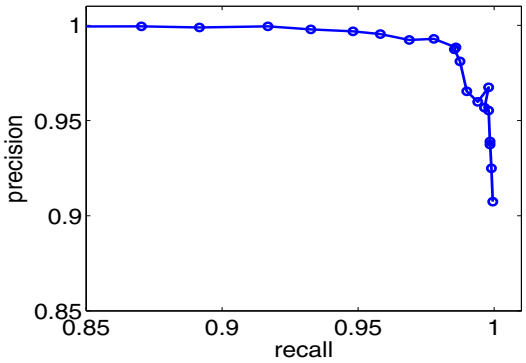


Fig. 5. Precision-recall curve for the thresholds between -1 and 1, in intervals of 0.1. The circles between the segments correspond to the evaluations.

$$t_p = \begin{cases} 1 \; \textit{if} \; \frac{area(b_d \cap b_g)}{area(b_d \cup b_g)} > 0.5, \\ 0 \qquad \textit{otherwise,} \end{cases} \tag{9}$$

where b_d and b_g correspond to the bounding box of the detection and the ground truth (i.e., test samples labeled as positives or negatives), respectively. This curve is very useful to find a suitable threshold for the SVM margin that satisfies a commitment between sensitivity and a true positive rate or specificity of detections.

During tracking, we extracted foreground objects by subtracting the current depth image from a depth image without objects. Then, we found the pixel corresponding to the highest point in the depth map, and discarded all the pixels corresponding to the connected component for that pixel. This procedure was repeated until there was no point that was 0.2 (`m`) above the floor. The values of the constants in (5) and (6), and the values of α, β and n are 1, 0.8, and 6 respectively.

To generate the bi-directional classifier we used a sequence with 1,000 (`frames`) for each scenario. After a set of tracklets was obtained, each tracklet was normalized to make it coincide with the origin. Each side was modeled with a multivariate normal density with covariance estimates computed by class. For instance, for the curve corresponding to the scenario illustrated in Fig. 4 the coefficients for $\mathbf{A}$, $\mathbf{b}$ and c correspond to $10^{-4} \begin{pmatrix} 0.2579 & -0.5389 \\ -0.5389 & 0.016 \end{pmatrix}$, $\begin{pmatrix} -0.0178 \\ 0.03 \end{pmatrix}$, and 0.5249, respectively.

To count the number of people, we select a one hour sequence of images for each scenario. In Fig. 6 we show the results organized as pedestrians going in one direction (*in*) or the opposite direction(*out*). We show two columns corresponding to the result of our system and ground truth obtained by visual inspection of the sequences. The results vary widely but are nevertheless encouraging. Comparing results with other methods is difficult as they may vary depending on the particular sequence being studied. From our work, our sequences are available for review at `http://imagenes.cicataqro.ipn.mx/CountingPedestrians/`.

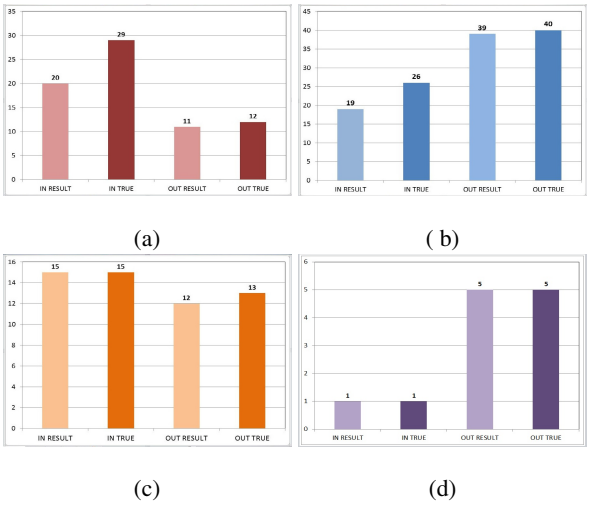


Fig. 6. Four different scenarios were tested collecting images for an hour. The bars show people going in one direction (*in*) or another (*out*). For each direction, we show the result of our method and ground truth.

8 Conclusion

In this document, we introduced a computer vision system to count the number of pedestrians in a bidirectional scenario. In our approach, we have introduced a people detector based on depth images (inspired on the scheme proposed by Dalal and TriggsTriggs[5]) that allows us to deal with complex scenarios in which what is moving in front of the camera may not necessarily be a person. We have included experimental evidence for each major stage of out method. In particular, as for the number of pedestrians that are being counted, the results, although varying with the scenario, are nevertheless encouraging.

In the future, we plan to streamline every stage of the processing sequence and experiment with more sophisticated scenes, such as those resulting from evacuations, where crowds of pedestrians move very close to each other but where the advantages of a zenithal camera to reduce the problems associated with occlusions can be demonstrated fully.

References

1. Chan, A., Liang, Z., Vasconcelos, N.: Privacy Preserving Crowd Monitoring: Counting People without People Models or Tracking. In: CVPR, pp. 1–7 (2008)
2. Chen, T., Chen, T., Chen, Z.: An Intelligent People-flow Counting Method for Passing through a Gate. In: Conference on Robotics, Automation and Mechatronics, pp. 1–6 (2006)
3. Chowdhury, A., Chatterjee, R., Ghosh, M., Ray, N.: Cell Tracking in Video Microscopy using Bipartite Graph Matching. In: ICPR, pp. 2456–2459. IEEE (2010)

4. Cristianini, N., Shawe-Taylor, J.: An Introduction to Support Vector Machines: And other Kernel-based Learning Methods. Cambridge University Press (2000)
5. Dalal, N., Triggs, B.: Histograms of Oriented Gradients for Human Detection. In: CVPR, pp. 886–893 (2005)
6. Enzweiler, M., Gavrila, D.: Monocular Pedestrian Detection: Survey and Experiments. IEEE Trans. on Pattern Anal. and Mach. Intell. 31(12), 2179–2195 (2009)
7. Everingham, M., Van Gool, L., Williams, C., Winn, J., Zisserman, A.: The PASCAL Visual Object Classes (VOC) Challenge. Int. J. Comput. Vision 88(2), 303–338 (2010)
8. Fischler, M., Bolles, R.: Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography. Communications of the ACM 24(6), 381–395 (1981)
9. Geronimo, D., Lopez, A., Sappa, A., Graf, T.: Survey of Pedestrian Detection for Advanced Driver Assistance Systems. IEEE Trans. on Pattern Anal. and Mach. Intell. 32(7), 1239–1258 (2010)
10. Haubner, N., Schwanecke, U., Dorner, R., Lehmann, S., Luderschmidt, J.: Towards a Top-View Detection of Body Parts in an Interactive Tabletop Environment. In: Architectures of Computing Systems (2012)
11. Kilambi, P., Ribnick, E., Joshi, A., Masoud, O., Papanikolopoulos, N.: Estimating Pedestrian Counts in Groups. Computer Vision and Image Understanding 110(1), 43–59 (2008)
12. Kong, D., Gray, D., Tao, H.: A Viewpoint Invariant Approach for Crowd Counting. In: ICPR, vol. 3, pp. 1187–1190 (2006)
13. Kuhn, H.: The Hungarian Method for the Assignment Problem. Naval Research Logistics Quarterly 2(1-2), 83–97 (2006)
14. Li, J., Huang, L., Liu, C.: Robust People Counting in Video Surveillance: Dataset and System. In: AVSS, pp. 54–59 (2011)
15. Press, W.H., Teukolsky, S.A., Vetterling, W.T., Flannery, B.P.: Numerical Recipes. Cambridge University Press (2007)
16. Raheja, J., Dutta, P., Kalita, S., Lovendra, S.: An Insight into the Algorithms on Real-Time People Tracking and Counting System. Int. J. of Comp. Appl. 46(5), 1–6 (2012)
17. Rowan, M., Maire, F.D.: An Efficient Multiple Object Vision Tracking System using Bipartite Graph Matching. In: FIRA Robot World Congress. FIRA Robot World Congress (2004)
18. Salti, S., Cavallaro, A., Di Stefano, L.: Adaptive Appearance Modeling for Video Tracking: Survey and Evaluation. IEEE Trans. on Image Process. (2012)
19. Seber, G.: Multivariate observations. Wiley (1984)
20. Shotton, J., Fitzgibbon, A., Cook, M., Sharp, T., Finocchio, M., Moore, R., Kipman, A., Blake, A.: Real-time human pose recognition in parts from single depth images. In: CVPR (2011)
21. Spinello, L., Arras, K.: People Detection in RGB-D Data. In: IROS, pp. 3838–3843 (2011)
22. Toyama, K., Krumm, J., Brumitt, B., Meyers, B.: Wallflower: Principles and Practice of Background Maintenance. In: ICCV, p. 255 (1999)
23. Yahiaoui, T., Meurie, C., Khoudour, L., Cabestaing, F.: A People Counting System based on Dense and Close Stereovision. In: Image and Signal Processing, pp. 59–66 (2008)
24. Zhang, E., Chen, F.: A Fast and Robust People Counting Method in Video Surveillance. In: International Conference on Computational Intelligence and Security, pp. 339–343 (2007)

Tracklet Reidentification in Crowded Scenes Using Bag of Spatio-temporal Histograms of Oriented Gradients

Michał Lewandowski[1], Damien Simonnet[1], Dimitrios Makris[1], Sergio A. Velastin[1,2], and James Orwell[1]

[1] Digital Imaging Research Centre, Kingston University, UK
[2] Department of Informatic Engineering, Universidad de Santiago de Chile, Chile
m.lewandowski@kingston.ac.uk

Abstract. A novel tracklet association framework is introduced to perform robust online re-identification of pedestrians in crowded scenes recorded by a single camera. Recent advances in multi-target tracking allow the generation of longer tracks, but problems of fragmentation and identity switching remain, due to occlusions and interactions between subjects. To address these issues, a discriminative and efficient descriptor is proposed to represent a tracklet as a bag of independent motion signatures using spatio-temporal histograms of oriented gradients. Due to the significant temporal variations of these features, they are generated only at automatically identified key poses that capture the essence of its appearance and motion. As a consequence, the re-identification involves only the most appropriate features in the bag at given time. The superiority of the methodology is demonstrated on two publicly available datasets achieving accuracy over 90% of the first rank tracklet associations.

Keywords: multi-target tracking, tracklet association, visual surveillance, histogram of oriented gradients, computer vision.

1 Introduction

Multi-target tracking in real crowded scenes is a fundamental problem for many vision applications, and especially for visual surveillance. The performance of pedestrian tracking algorithms has been improving to reach the point where generated tracks are increasingly reliable over longer period of time. As a consequence, the research community has devoted more attention to tracking in dense crowds.

For instance, Nevatia et al [15] propose an online learning framework to learn non-linear motion patterns as well as robust appearance models based on estimated entry/exit points. Alternatively, Kuo el al [6] suggest a robust association using appearance based affinity measure by jointly solving ranking and classification of associations using online weight learning for weak classifiers trained previously offline. In turn, Suna et al [12] formulate the problem as maximum weight matching in a bipartite graph, whose solution is learned through a structured support vector machine. In Song et al [11], a system which simultaneously performs the scene-semantic-learning and tracking is presented and makes them supplement to each other, whereas Bo Yang et al [16] propose to learn using Conditional Random Fields and subsequently progressively associating detection responses into long tracks.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 94–103, 2013.

Another line of work follows the standard tracking-by-detection procedure. While Benfold et al [2] combine detections with a Markov Chain Monte Carlo Data Association and a KLT tracker, Andriyenko et al [1] present multi-target tracking as the minimization of a continuous energy function. On the other hand, Simonnet et al [9] extend the standard framework with a hierarchy of Bayesian local predictions using a dynamic occlusion map to supplement the responses from detector. Finally, Song et al [10] introduce a stochastic graph with an evolution step that considers the statistical properties of individual tracklets, as well as the statistics of the targets along each proposed long-term track to perform association.

However, long and frequent occlusions as well as interactions between targets are still challenging, especially for tracking-by-detection approaches, and result in the generation of highly fragmented tracks, hence called 'tracklets', with frequent identity switches between them. To overcome this problem, we propose a novel tracklet association process which is formulated as a re-identification process based on the observation that, for any given pedestrian, the appearances of tracklets (e.g. before and after an occlusion) tends to be similar. As a consequence, an discriminative and efficient tracklet descriptor can be used to rank candidates and select those that correspond to the same pedestrian. In most of typical visual surveillance scenarios, even tracklets corresponding to the same individual still differ significantly, in terms of duration and orientation with respect to the camera. Moreover, they may consist of a number of walking cycle repetitions which in turn may performed at various speeds. Hence, it is extremely difficult to encapsulate all these variabilities in a single global tracklet descriptor. Therefore, we propose to represent a tracklet as a bag of local independent spatio-temporal descriptors which express the unique appearance properties of the motion, over relatively short periods of time around 'key poses'. This is achieved by first splitting trajectories of individuals into approximately linear segments based on motion direction and speed. Then, the motion signatures are extracted by analyzing periodicity of such segments. Finally, the obtained signatures are used to generate a representative set of spatio-temporal features for each tracklet. This allows a re-identification process to use only the most appropriate local descriptors at any given time, and hence substantially mitigates the impact of variability in tracklet duration, speed or orientation. As a result, the introduced tracklet re-identification framework produces not only less fragmented tracks, but also introduces less 'identity-switch' type errors.

2 Methodology

Fig. 1 presents an overview of the proposed approach. The whole framework consists of the following stages which are discussed in subsequent sections: tracklet generation, trajectory segmentation, periodicity analysis, local descriptors generation and tracklet association.

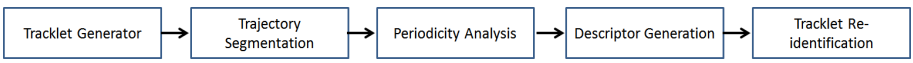


Fig. 1. An overview of tracking framework

Tracklet Generator. The tracking of multiple targets in sets of observations is performed by an online tracklet generator. It is based on the extended tracking-by-detection principle [9]. First, pedestrians are detected globally using the HoG detector [4] followed by a local detection process using a dynamic occlusion map when no global observation is available [9]. In turn the prediction, selection of the validated measurement and update of the tracker state is carried out using the Kalman Filter [13]. Since tracklets are very likely to be correct when maintaining a spatial proximity to a single pedestrian for their duration, a tracklet classifier is incorporated to provide an indication of tracklet usability for the next stage without needing to wait until the tracklet is completed [9].It allows classification of tracklets at a given frame into three states: *tentative*, and *reliable* (when the spatio-temporal reliability condition is satisfied at each time-step up to Δ frames or after Δ frames respectively) and *unreliable* (at given time-step the reliability condition is not satisfied up to Δ frames). As a consequence, a *tentative* state stands for a tracklet which is not yet long enough to extract meaningful features for the tracklet data association step and only becomes *reliable* or *unreliable* after Δ frames. In turn, *reliable* tracklets are sufficiently informative to be used in the tracklet association stage, while *unreliable* tracklets are considered as false alarms by the system. Finally, a generated tracklet T is a short sequence of appearance features with variable length L, i.e. $T = \{f_i\}_{i=1}^{L}$, which are extracted from the corresponding bounding boxes in which the target is not occluded for extended time periods or to a significant spatial extent.

Trajectory Segmentation. The extracted tracklets are split into subsequent mutually exclusive fragments (segments) so that $T = \{S_s\}, S_s = \{f_i\}_{i=1}^{L_s}, \prod_s S_s \equiv \emptyset, \sum_s L_s = L$ to separate significantly different appearance observations due to motion orientation with respect to the camera. As a consequence, the periodicity analysis can be performed for each segment as well as most of identity switches can be detected. This segmentation is achieved by localizing fragments within a trajectory with an approximately constant velocity.

Initially, the trajectory for a tracklet is assembled as a temporal sequence of coordinates of pedestrians' bounding boxes (e.g. the 'centre' or 'foot' of bounding box). When analysed in a temporal domain, the trajectory is decomposed to three different motion components along vertical/horizontal direction and speed (Fig. 2). Segmentation is performed in a three level hierarchical framework. In the highest level, changes in the direction along either horizontal or vertical axis are detected. Linear segments are fitted into curve trajectory in least-squares sense using Free-Knot B-Spline approximation [8] (Fig. 2). As a result, the trajectory curve is approximated by a set of linear segments which allows identification of unique peaks of sudden changes along chosen dimension (horizontal/vertical). These peaks segment the tracklet into fragments of constant direction. Afterwards this process is repeated for each segment independently in the other direction dimension (vertical or horizontal). Finally, on the lowest level each obtained segment is analysed again in terms of speed variation and when the noticable change is detected, the segment is further divided.

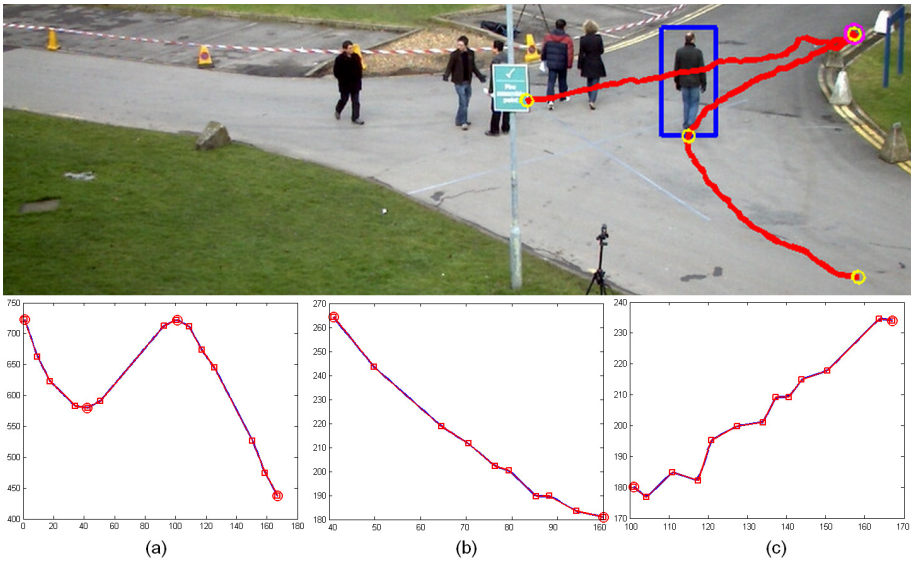


Fig. 2. Trajectory segmentation based on a horizontal or vertical direction of object position (y-axis) along the time (x-axis) where red squares correspond to borders of linear segments fitted into trajectory and red/yellow circles are peaks where the change is detected: (a) segmentation of horizontal direction into three segments, (b) and (c) segmentation of the obtained 2nd and 3rd segment respectively along vertical direction with no detected further changes

To address problem of identity switches, the tracklet $T = \{1, ..., S_s, ...\}$ is allowed to split into two different tracklets $T_1 = \{1, ..., S_s\}$ and $T_2 = \{S_{s+1}, ...\}$ when the estimated subsequent segments represented as vectors in the image space have opposite motion directions and an angle between them is less then a constant tolerance threshold:

$$\arccos\left(\frac{\overrightarrow{\boldsymbol{S_s}} \cdot \overrightarrow{\boldsymbol{S_{s+1}}}}{|\overrightarrow{\boldsymbol{S_s}}||\overrightarrow{\boldsymbol{S_{s+1}}}|}\right) < -A \tag{1}$$

Although, this procedure cannot handle identity switches between subjects moving into roughly the same direction, it is computationally efficient and accurate for dealing with crossing identity switches. Note that in very rare cases, the abrupt turnaround of a subject may result in division of such tracklet, but such a split can generally be recovered in the tracklet re-identification process.

Periodicity Detection. After segmentation, each trajectory fragment is analysed in terms of motion periodicity to extract the key poses f_k which repeat themselves with a constant period of $2\pi Q'$ where Q' is the total number of cycle repetitions within the segment.

Initially, a square temporal self-similarity matrix is constructed for each object as it evolves in time for a given trajectory fragment S_s [7] (Fig. 3a):

$$[d_{i,j}]_{i,j=1..L_s} = \begin{bmatrix} 0 & d_{1,2} & d_{2,2} & ... & d_{1,L_S} \\ d_{2,1} & 0 & d_{2,3} & ... & d_{2,L_S} \\ ... & ... & ... & ... & ... \\ d_{L_S,1} & d_{L_S,2} & d_{L_S,3} & ... & 0 \end{bmatrix}. \quad (2)$$

where $d_{i,j}$ is the Euclidean distance between low level features f at time instants i, j like histogram of oriented gradients [4]. The autocorrelation of such self-similarity matrix is a variant of the recurrence plot which encodes different spatio-temporal properties and behaviours of dynamic systems such as periodic patterns [3]. In particular, the periodicity of the gait creates parallel diagonals in this matrix with the main diagonal of 0 corresponding to comparing frame to itself (no dissimilarity) (Fig. 3b). In turn, the temporal symmetry of gait cycles are represented by cross diagonals which are orthogonal to the main diagonal (Fig. 3c). The period of motion, i.e. the time difference between periodic patterns, can be determined by analysing the intersections between diagonal and cross diagonal lines which corresponds to the change of motion phase.

First, the 2D autocorrelation of the similarity matrix is computed using Fourier transform based on Wiener-Khinchin Theorem [14] and then smoothed by applying the Gaussian filter. As a result, local peaks along cross diagonal of autocorrelation matrix are revealed (Fig. 3b)). Then for each local extreme a diagonal 'corridor' is created of a predefined size. All extremes within this corridor with coordinates (x, y) are used to fit the optimal crossed diagonal line $y = ax + b$ with parameters $\beta = [a, b]$, i.e. slope a and shift b. This is achieved by the standard linear least square solution of an over-determined system of equations:

$$argmin_\beta \|y - \beta[x, 1]^T\|^2 \quad (3)$$

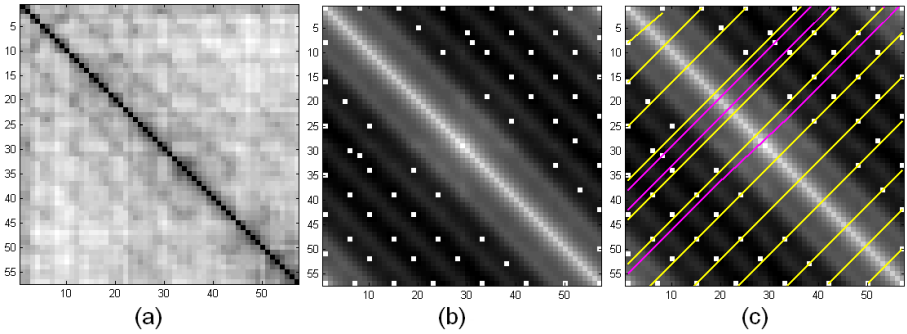


Fig. 3. (a) Temporal self-similarity matrix of a tracklet segment. (b) Autocorrelation of the self-similarity matrix with local peaks used to estimate parameters of cross diagonal lines. (c) Autocorrelation matrix with discarded three noisy cross diagonals (magenta) and an optimal set of cross diagonals used to determine the period of analysed motion segment (yellow).

which minimizes the sum of squared residuals, i.e. differences between observed values and their values estimated by a model given parameters β. The approximated solution is obtained by normal equations or more numerically stable orthogonal decomposition methods such as QR or SVD decompositions.

The obtained set of Q crossed diagonal lines share almost the same slope close to -1 due to temporal symmetry of the matrix and differs by shift parameters $b_q > b_{q-1}$ which express the intersection of the main diagonal of the autocorrelation matrix with corresponding cross diagonals. Based on recurrence plot patterns [3], each b_q should correspond to a phase change of periodic signal, i.e. $\sin(b_q) = 0$, so the objective is to find the best fit of periodic sine function at points of phase change, i.e. $r = [0 : \pi : \pi * Q']$ with obtained shift parameter space by minimizing following equation:

$$E_{Q'} = \sum_{q=1}^{Q} min(|b_q - r(Q')|), Q' = 2 : Q$$
$$argmin_{Q'}\ detrend(E) \tag{4}$$

Note that some of cross diagonals may not share the same periodicity pattern due to noisy observations (the magenta lines in Fig. 3c)), therefore the whole range of phases $\pi Q'$ needs to be analyzed to discard such noisy cross diagonals and find the best global fit. The final frame frequency H between phases is given by: $H = 2 * L/Q'$, while key pose f_k corresponds to location in time of phase change $b_{q'}$ so that $k = (q' - 1) * H$. Fig. 4a presents results of such minimization process with the corresponding optimal fit of sinus function into shift parameter space in Fig. 4b. In turn, Fig. 3c shows in yellow colour cross diagonals which satisfied the estimated periodicity pattern, whereas Fig. 5 illustrates some examples of detected key poses corresponding to the phase change of motion within the tracklet.

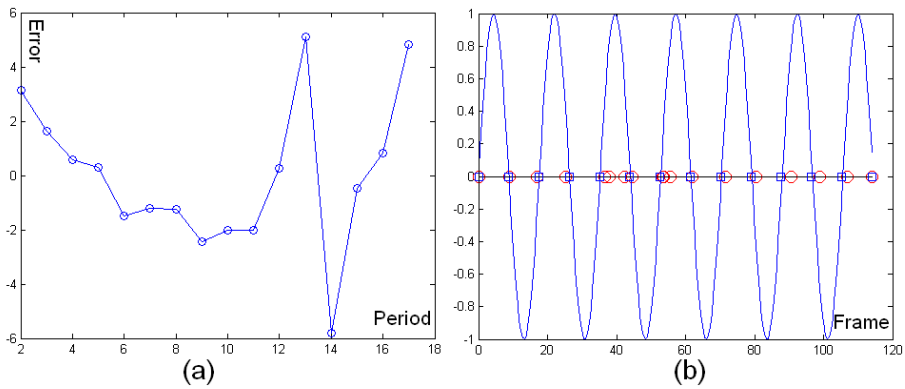


Fig. 4. (a) The detrend error of fitting periodic sine function into a sequence of shift parameters at points of phase change. (b) The best fit of periodic sine function (blue) into a sequence of shift parameters (red) at points of phase change.

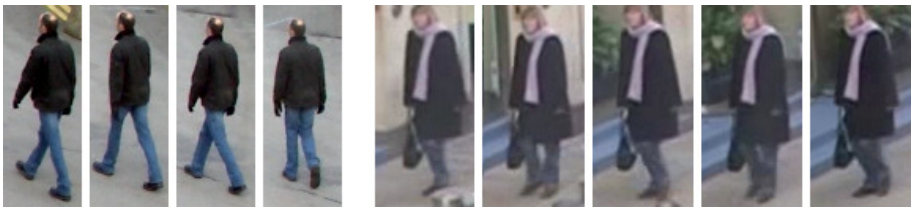


Fig. 5. Examples of detected key poses corresponding to the phase change of motion within the tracklet

Descriptor Generation. At this stage, a tracklet is represented as a bag of local and independent motion signatures $M = \{m_k\}, m_k = \{f_i\}_{i=k-cH/2}^{k+cH/2}$ of the length cH where parameter c controls overlapping of signatures. These signatures are generated at estimated key poses f_k and each represented by a spatio-temporal descriptor which capture a key appearance properties of the motion over corresponding short period of time around change of motion phase. Here, we propose to adopt the histogram of oriented 3D gradients [5] as a descriptor which proves to be very effective for action recognition but to the best of our knowledge has not been yet applied for tracking. Such a histogram is extracted densely inside a bounding box based video volume of given motion signature in a very efficient way due to integral video representation of 3D gradients. Such volume is first divided into cells at different spatio-temporal scales. Then, for each cell the mean 3D gradient vector is computed and its full orientation is quantized using 3D polar binning. Afterwards, the cells are normalized and concatenated to form the histogram vector which is finally normalized over all cells. For very short tracklets or tracklets without periodic pattern, only one motion signature is generated at its centre.

Tracklet Data Association. Given a *reliable* 'probe' tracklet T_z, the objective is to identify and then uniquely associate the most similar subsequent tracklet among a 'gallery' set of N_t predicted tracklets $(T_{z,v})_{v\in[\![1,N_t]\!]}$, generated during a time window t. This is achieved by first comparing all motion signatures of both tracklets and the best ones are ranked according to:

$$rank_{v=1..N_t}(W \min(\chi^2(M_z, M_{z,v}))) \tag{5}$$

where χ^2 is the chi-square distance between histograms and W is penalty weight expressed by: $W = exp(\|f_{z,L} - f_{z,v,1}\|$ which favours candidate tracklets which are spatially closer to the probe tracklet.

3 Evaluation

The proposed framework was validated using two public benchmark datasets, i.e. PETS2009 [12] (Scenario S2.L1) and Oxford [2], which have been commonly used in recent multi-target tracking research. Both datasets containing typical crowded pedestrian scenes. The evaluation follows methodologies of [1,6,15,12] for PETS2009

and [2,9] for Oxford to provide the fair comparison with the current state of the art approaches. The considered metrics include: recall and precision rate (showing detection performance after tracking), false alarms per frame (FA), mostly tracked (MT), mostly lost (ML) and partially tracked (PT) (the ratio of tracks with successfully tracked parts for more than 80% or less than 20% or between respectively), track fragmentation (Frag) (the number of times that a ground truth trajectory is interrupted) and id switches (IDS) (the number of times that a track changes its matched id). Note that novelty of this paper is focused on the tracklet re-identification, thus the most relevant metrics are track fragmentation and identity switches.

The tracklet generator was initialized using a few default parameters [9]. As a feature for pedestrian detection and periodicity estimation, the standard spatial histogram of oriented gradients was used [4] (8x8 pixel cells, 9 orientation bins). On the other hand, the motion signatures were represented as spatio-temporal histograms of oriented gradients [5] (7x7x2 cells, 15 orientation bins). Motion signatures did not overlap ($c = 1$), whereas the tolerance angle for splitting tracks was set to $A = 25°$.

Tables 1 and 2 present the performance of the proposed framework in comparison to other state-of-the-art approaches. The improved performance is clearly demonstrated in the *track fragmentation* and *identity switch* metrics, which are arguably the most relevant indicators. Other metrics are still competitive with other approaches.

This is supported by obtaining almost 96% and 90% of first rank tracklet association for Oxford and PETS2009 datasets respectively (Fig. 6). This accuracy difference was expected since in the Oxford dataset motion orientation of pedestrians with respect to

Table 1. Comparison of results on Oxford sequence

Method	FA	GT	MT	PT	ML	Frag	IDS
SWT [2]	0.190	201	0.42%	0.46%	12.0%	144	91
DTWAF [9]	0.060	201	0.40%	0.43%	17.0%	44	28
Our approach	0.067	201	0.44%	0.48%	8.0%	20	15

Table 2. Comparison of results on PETS2009 sequence

Method	Recall	Precission	FA	GT	MT	PT	ML	Frag	IDS
Energy Min. [1]	-	-	-	23	82.6%	17.4%	0.0%	21	15
PRIMPT [6]	89.5%	99.6%	0.020	19	78.9%	21.1%	0.0%	23	1
NLMPRAM [15]	91.8%	99.0%	0.053	19	89.5%	10.5%	0.0%	9	0
S-SVM MOT1 [12]	97.2%	93.7%	0.379	19	94.7%	5.3%	0.0%	19	4
S-SVM MOT2 [12]	96.6%	93.4%	0.396	19	94.7%	5.3%	0.0%	26	5
Our approach	93.3%	99.1%	0.025	19	89.5%	10.5%	0.0%	4	1

the camera is much more stable than in PETS2009, thus making the re-identification process more reliable. On the other hand, in the former dataset, recovery from identity switches is more of a challenge (than in the latter dataset).

Note that in contrast to our methodology, frameworks [15,12,6] require either some context information a priori or are learned online which in some scenarios may not be feasible or difficult to perform due to lack of appropriate training data. Finally, our results may be further improved when a more advanced tracker is used to generate initial tracklets.

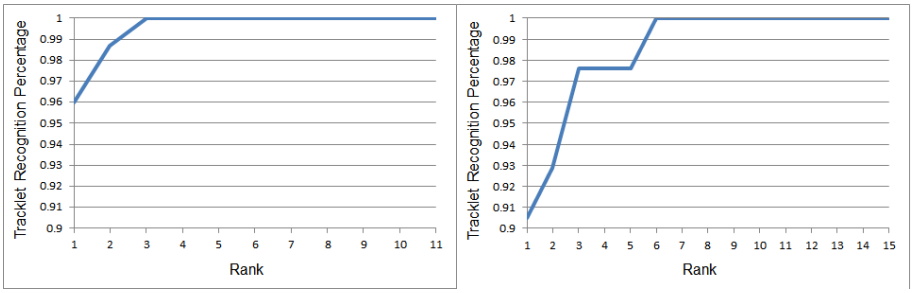


Fig. 6. Cumulative Matching Characteristic of ranking tracklets for Oxford (left) and PETS2009 (right) datasets

4 Conclusion

This paper has proposed a new descriptor for robust tracklet association using a bag of motion signatures expressed by spatio-temporal histograms of oriented gradients. To overcome problems of tracklet variability such as change of velocity or existence of cyclic repetitions, features in bag are generated at various time points to encapsulate key appearance and dynamic properties over short period of time. As a consequence, the ranking procedure for re-identification can take into account only the best features at given time as well as recover from badly assigned identities during tracking. The experiments have confirmed that our approach provides generally comparable performance to other methods, while being advantageous in the reduction of track fragmentation and number of identity switches, thus producing more reliable tracks.

References

1. Andriyenko, A., Schindler, K.: Multi-target tracking by continuous energy minimization. In: Proc. of CVPR (2011)
2. Benfold, B., Reid, I.: Stable Multi-Target tracking in Real-Time surveillance video. In: Proc. of CVPR (2011)
3. Cutler, R., Davis, L.S.: Robust Real-Time periodic motion detection, analysis, and applications. TPAMI (2000)
4. Dalal, N., Triggs, B.: Histograms of oriented gradients for human detection. In: Proc. of CVPR (2005)

5. Kläser, A., Marszałek, M., Schmid, C.: A Spatio-Temporal descriptor based on 3D-gradients. In: Proc. of BMVC (2008)
6. Kuo, C.-H., Nevatia, R.: How does person identity recognition help Multi-Person tracking? In: Proc. of CVPR (2011)
7. Lele, S.: Euclidean distance matrix analysis (edma): Estimation of mean form and mean form difference. In: Mathematical Geology (1993)
8. Schütze, T., Schwetlick, H.: Constrained approximation by splines with free knots. In: BIT Numerical Mathematics (1997)
9. Simonnet, D., Lewandowski, M., Velastin, S., Orwell, J., Turkbeyler, E.: Tracking pedestrians in crowded scenes using dynamic time-warped appearance features. In: Workshop PCRA (2012)
10. Song, B., Jeng, T.-Y., Staudt, E., Roy-Chowdhury, A.K.: A stochastic graph evolution framework for robust multi-target tracking. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 605–619. Springer, Heidelberg (2010)
11. Song, X., Shao, X., Zhao, H., Cui, J., Shibasaki, R., Zha, H.: An online approach: Learning-semantic-scene-by-tracking and tracking-by-learning-semantic-scene. In: Proc. of CVPR (2010)
12. Kim, S., Kwak, S., Feyereisl, J., Han, B.: Online multi-target tracking by large margin structured learning. In: Lee, K.M., Matsushita, Y., Rehg, J.M., Hu, Z. (eds.) ACCV 2012, Part III. LNCS, vol. 7726, pp. 98–111. Springer, Heidelberg (2013)
13. Welch, G., Bishop, G.: An introduction to the kalman filter (1995)
14. Wiener, N.: Generalized harmonic analysis. Acta Mathematica (1930)
15. Yang, B., Nevatia, R.: Multi-target tracking by online learning of non-linear motion patterns and robust appearance models. In: Proc. of CVPR (2012)
16. Yang, B., Huang, C., Nevatia, R.: Learning affinities and dependencies for multi-target tracking using a crf model. In: Proc. of CVPR (2011)

LIDAR and Panoramic Camera Extrinsic Calibration Approach Using a Pattern Plane

Angel-Iván García-Moreno, José-Joel Gonzalez-Barbosa, Francisco-Javier Ornelas-Rodriguez, Juan B. Hurtado-Ramos, and Marco-Neri Primo-Fuentes

CICATA, Instituto Politécnico Nacional
Cerro Blanco 141 Col. Colinas del Cimatario, Quéretaro
jgonzalezba@ipn.mx

Abstract. Mobile platforms typically combine several data acquisition systems such as lasers, cameras and inertial systems. However the geometrical combination of the different sensors requires their calibration, at least, through the definition of the extrinsic parameters, i.e., the transformation matrices that register all sensors in the same coordinate system. Our system generate an accurate association between platform sensors and the estimated parameters including rotation, translation, focal length, world and sensors reference frame. The extrinsic camera parameters are computed by Zhang's method using a pattern composed of white rhombus and rhombus holes, and the LIDAR with the results of previous work. Points acquired by the LIDAR are projected into images acquired by the Ladybug cameras. A new calibration pattern, visible to both sensors is used. Correspondence is obtained between each laser point and its position in the image, the texture and color of each point of LIDAR can be know.

Keywords: panoramic camera, LIDAR, sensor calibration, extrinsic calibration.

1 Introduction

Reconstruction of urban environments is a challenge given the variety of scenes that can be scanned and problems that arise when working with real data. Many techniques are used for three-dimensional reconstruction, each with their own limitations and none of them efficiently solves the problems of digital modeling of urban environments. Recently new approaches for this purpose have been developed with the idea of using the information from different sensors, depth laser sensors, CCD cameras and inertial systems primarily [9] [4].

Information fusion from multiple sources is an important issue when processing data as it must have an accurate calibration of all the instruments of the acquisition platform, a good real-time data fusion technique is proposed in [9], they fuse colour camera, range scanning anda navigation data to produce a three dimensional colour point clouds. For this article we will focus on characterizing

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 104–113, 2013.

the rigid transformation between the camera and laser sensor. Using the intrinsic parameters of the camera and LIDAR, and the transformation we project a 3D point from LIDAR to the image plane of the camera. The problem of rigid transformation is solved by matching features of the 3D calibration pattern acquired from LIDAR and camera. These features are usually the corners of planar checkboard pattern, in the image as an intersection of black and white square, in the LIDAR correspont the intersection to square plane and hole square.

In this work, two types of sensors are used, *Velodyne HDL-64E* laser scanner and *Point Grey Ladybug2* spherical digital camera. The laser scanner operates by pulsing a laser diode for a short duration (typically 4 nanoseconds) and precisely measuring the amount of time it takes for the pulse to travel to an object, reflect off and return to a highly sensitive photodetector and camera system is a high resolution omnidirectional sensor, it has six $0.8 - Megapixel$ cameras, with five CCDs positioned in a horizontal ring and one positioned vertically, that enable the system to collect video from more than 75% of the full sphere. Complementary benefits of both sensors, it's possible to acquire more reliable scenes have characteristics such as depth, color and orientation. Both sensors are georeferenced. Data fusion approaches between LIDAR and images can be divided in two categories [11]:

- Centralized: the data fusion process occurs at pixel or feature level, i.e., the LIDAR and camera characteristics are combined into a single vector for subsequent classification. One drawback to this approach is that only areas that are seen by both sensors can be processed.
- Decentralized: data processing from each sensor is made separately and then fusion occurs. These methods usually require training to determine the fusion model.

The estimation of LIDAR and camera intrinsic parameters is a nonlinear problem that can be solved in different ways. A novel algorithm is proposed [5] for joint estimation of both the intrinsic parameters of the laser sensor and the LIDAR-camera transformation. Specifically, they use measurements of a calibration plane at various configurations to establish geometric constraints between the LIDAR's intrinsic parameters and the LIDAR-camera 6 d.o.f. relative transformation. They process these measurement constraints to estimate the calibration parameters as follows: First, analytically compute an initial estimate for the intrinsic and extrinsic calibration parameters in two steps. Subsequently, they employ a batch iterative (nonlinear) least-squares method to refine the accuracy of the estimated parameters. Other method using a planar board checkerboard patterns is proposed in [6], the autors defines the rotation matrix between the sensor laser and camera as achieved moving they platform and observing the resulting motion of the sensors, this step attempts to solve the well-known homogeneous transform, translation is calculated using a commonly least-squares estimation algorithm according to the corners of the pattern, detected by both sensors. Besides the problem of calibration in [8] provide a solution for the occlusion problem that arises in conjuntion with different view points of the fusioned sensors, they approach first perfoms a synchronization of both sensors to allow

for moving objects which incorporates an inertial correction of the LIDAR data and an automatic trigger mechanism depending on the view direction of the camera, the occlusion detection algorithm uses 2D convex hulls derived from a 3D object segmentation to allow for a visibility check of projected 3D LIDAR data.

Not only square flat patterns are used for calibration between these two sensors, in [7] they use a circle-based calibration object because its geometry allows to obtain an accurate estimation pose in the camera frame and the camera intrisic parameters. The autors use a linear minimization of the Euclidian distance error between the 3D circle center point sets, then they first generate the 3D circles of the n poses estimated by the camera, it consists in computing m points of each estimated circle pose by using the 3D circle center and an orthonormal base lying in circle's plane. Other approach using a conic based geometry object to calibrate 2D/3D laser sensors is presented in [1].

2 Platform Projection Model

The goal of the method is to find homogeneous transformation between the pinhole camera and the LIDAR in order to fuse the measurements from both sensors in urban digitalize enviroments applications.

Data is collected by a mobil platform using the data collection vehicle shown in Figure 1. This mobile mapping system is composed of a LIDAR sensor, video camera, GPS, Inertial Measurement Unit (IMU).

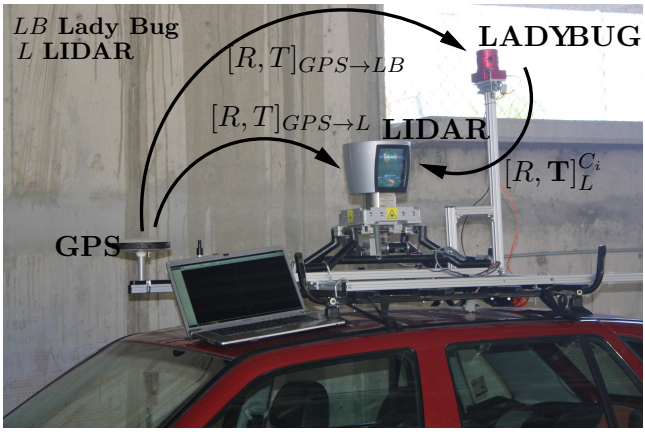


Fig. 1. Sensor platform composed of LIDAR Velodyne HDL-64E (L) Ladybug2 (LB) and GPS. Ladybug2® spherical digital video camera system has six cameras (C_i). L, C_i represent the LIDAR and six camera frames of the Ladybug.

This research work focuses the refinement of the registration of LIDAR and panoramic images.

2.1 Panoramic Camera and LIDAR Model

The six cameras frames of the *Ladybug2* are represented by C_i (where $i = 1, \ldots, 6$). The 3D points acquired by the LIDAR ($\mathbf{X}^L$) are transform from LIDAR frame to camera frames by $[R_L^{C_i}, \mathbf{T}_L^{C_i}]$, called the extrinsic parameters (see Figure 2.)

A 2D point in the camera C_i is denoted by $\mathbf{u}^{C_i} = \begin{bmatrix} u^{C_i} & v^{C_i} \end{bmatrix}^T$. A 3D point in LIDAR frame is denoted by $\mathbf{X}^L = \begin{bmatrix} X^L & Y^L & Z^L \end{bmatrix}^T$. We use $\hat{\mathbf{x}}$ to denote the augmented vector by adding 1 as the last element: $\hat{\mathbf{u}}^{C_i} = \begin{bmatrix} u^{C_i} & v^{C_i} & 1 \end{bmatrix}^T$ and $\hat{\mathbf{X}}^L = \begin{bmatrix} X^L & Y^L & Z^L & 1 \end{bmatrix}^T$. A camera is modeled by the usual pinhole: The image $\mathbf{u}^{C_i}$ of a 3D point $\mathbf{X}^L$ is formed by an optical ray from $\mathbf{X}^L$ passing through the optical center C_i and intersecting the image plane. The relationship between the 3D point $\mathbf{X}^L$ and its image projection $\mathbf{u}^{C_i}$ is given by

$$s\hat{\mathbf{u}}^{C_i} = A^{C_i}[R_L^{C_i}, \mathbf{T}_L^{C_i}]\hat{\mathbf{X}}^L = P_L^{C_i}\hat{\mathbf{X}}^L \tag{1}$$

with $A^{C_i} = \begin{bmatrix} -k_u f & 0 & u_0 & 0 \\ 0 & k_v f & v_0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}^{C_i}$

and $P = A^{C_i}[R_L^{C_i}, \mathbf{T}_L^{C_i}]$

where s is an arbitrary scale factor, $[R_L^{C_i}, \mathbf{T}_L^{C_i}]$, called the extrinsic parameters, is the rotation and translation which relates from LIDAR system L to camera system C_i, and A^{C_i} is called intrinsic matrix for the camera C_i, with (u_0, v_0) the coordinates of the principal point, fk_u and fk_v the scale factors in image u and v axes. The 3×4 matrix $P_L^{C_i}$ is called the camera projection matrix, which mixes both intrinsic and extrinsic parameters.

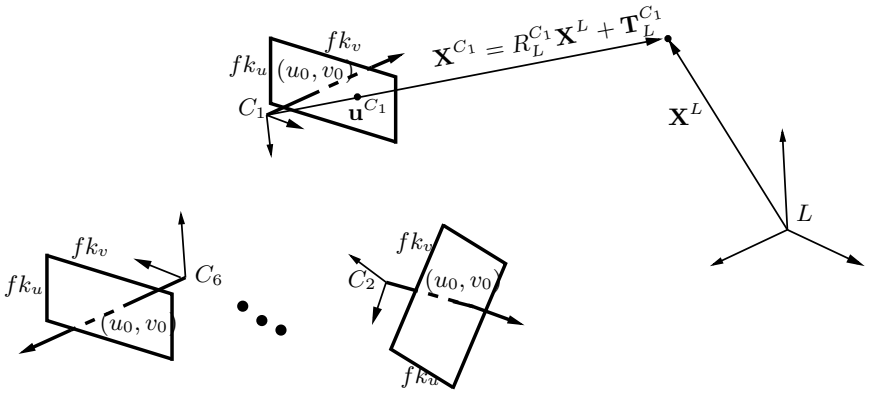


Fig. 2. LIDAR frame L and the six pinhole camera model

3 Panoramic Camera and LIDAR Calibration

The six cameras of the Ladybug2 are calibrated with the Zhang's method [10], in this the intrinsic parameters (A^{C_i}) of the cameras are computed. The LIDAR is calibrated using the method proposed in [2]. We use a pattern which facilitates the extraction of the same point in camera and LIDAR data. This pattern is shown in Figure 7 and it is composed of white rhombus and rhombus holes which take the black color by the background. The Figure 3 shows the relationship between the pattern W_i camera C_i and LIDAR L. The transformation between LIDAR and cameras shown in te Equation 1 is computed by

$$[R_L^{C_i}, \mathbf{T}_L^{C_i}] = [R, \mathbf{T}]_W^{C_i} * ([R, \mathbf{T}]_W^L)^{-1} \tag{2}$$

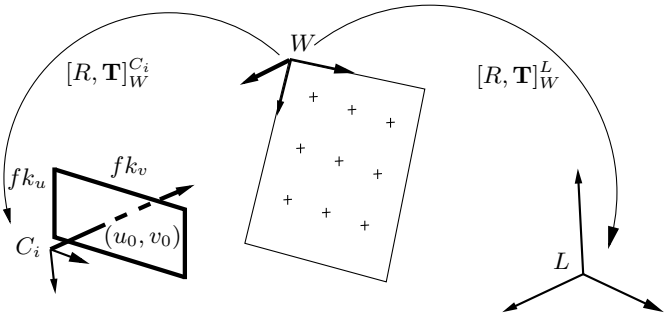


Fig. 3. The extrinsic parameters $[R, \mathbf{T}]_W^{C_i}$ are computed by Zhang's method and $[R, \mathbf{T}]_W^L$ are computed by Algorithm shows in Figure 4

In the algorithm shows in Figure 4, the function $transform\_to\_XYZ$ transform the LIDAR data to X, Y, Z points using the method proposed in [2]. RANdom SAmple Consensus ($RANSAC$) algorithm is widely used for plane detection in point cloud data. The principle of $RANSAC$ algorithm consists to search the best plane among a 3D point cloud. For this purpose, it selects randomly three points and it calculates the parameters of the corresponding plane. Then it detects all points ($nPatternPointCloud$) of the original cloud belonging to the calculated plane (Π), according to a given threshold. The $project\_to\_normal\_pattern$ algorithm projected the points $nPatternPointCloud$ to plane Π. In function $MaximizingFunction$, we build an artificial calibration pattern ($Pattern(radius, \theta)$) using the dimension of the pattern calibration. The artificial pattern can be moved on the plane Π rotating an angle θ and moving a distance $radius$ in the direction of rotation. The $nProjectedPattern$ points are comparing with the artificial pattern plane, the maximum comparison allows us to calculate the distance ($radius$) and angle (θ) in the plane where the reference pattern placed. Using the plane equation Π and $(radius, \theta)$ we computing the $[R, \mathbf{T}]_W^L$ parameters.

For each camera C_i do:

Algorithm for computing of extrinsic LIDAR parameters.

1: **for each** nRawPointCloud **do**
2: $nPointCloud \leftarrow transform\_to\_XYZ(nRawPointCloud)$
3: $(nPatternPointCloud, \Pi) \leftarrow RANSAC(nPointCloud)$ ▷ extract pattern points
4: $nProjectedPattern \leftarrow project\_to\_normal\_pattern(nPatternPointCloud)$
5: $[R, \mathbf{T}]_W^L \leftarrow MaximizingFunction(nProjectedPattern, \Pi)$
6: **end for**

Algorithm for computing extrinsic and intrinsic *Ladybug2* parameters.

1: **for each** nImage **do**
2: $[R, \mathbf{T}]_W^{C_i}, A^{C_i} \leftarrow$ `use Bouguet's camera calibration Toolbox`
3: **end for**

LIDAR and Ladybug calibration

1: **for each** nImage **do**
2: $[R_L^{C_i}, \mathbf{T}_L^{C_i}] = [R, \mathbf{T}]_W^{C_i} * ([R, \mathbf{T}]_W^L)^{-1}$
3: **end for**

Fig. 4. Algorithms used in this work

4 Results

The results are presented in three stages, the computation of extrinsic parameters of the LIDAR, the computation of extrinsic and intrinsic parameters of the cameras C_i of the Ladybug and extrinsic parameters between each camera and LIDAR. For practical purposes, we show the results by only one camera, however the Ladybug is formed with six cameras.

4.1 Algorithm for Computing of Extrinsic LIDAR Parameters

The Figure 5 shows the exracted points (blue points) using the $RANSAC$ algorithm and they proyected onto plane Π using the $project\_to\_normal\_pattern$ algorithm. The red circle in the Figure 5 represent the position of the artificial pattern plane. This position is compute by the algorithm $MaximizingFunction$. The $MaximizingFunction$ computes the matches between the real pattern data and the synthetic pattern, this is shown in Figure 6. The rotation and translation are performed on the plane Π, and the traslation is carried out in θ orientation.

The rigid transformation of a 3D point in the LIDAR frame, L, into the world frame (pattern frame) is defined by the rotation matrix and translation vector ($[R, \mathbf{T}]_W^L$):

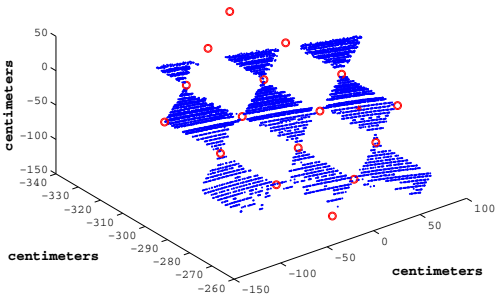


Fig. 5. Extraction of the calibration pattern using the *RANSAC* algorithm and projection onto the plane Π

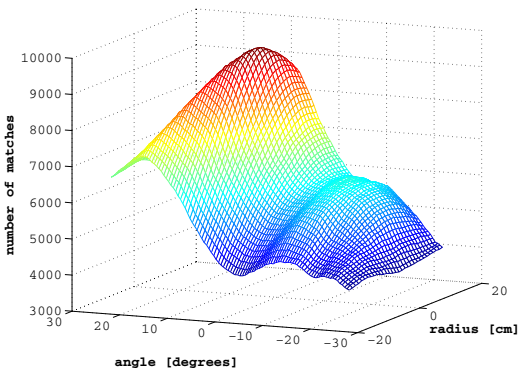


Fig. 6. The maximization function computes the number of points of the real pattern matches with the synthetic pattern

$$R_W^L = \begin{bmatrix} 0.1085 & -0.3006 & 0.9475 \\ -0.7800 & -0.6167 & -0.1063 \\ -0.6164 & 0.7275 & 0.3014 \end{bmatrix}; \mathbf{T}_W^L = \begin{bmatrix} -282.90 \\ 80.71 \\ -49.82 \end{bmatrix}$$

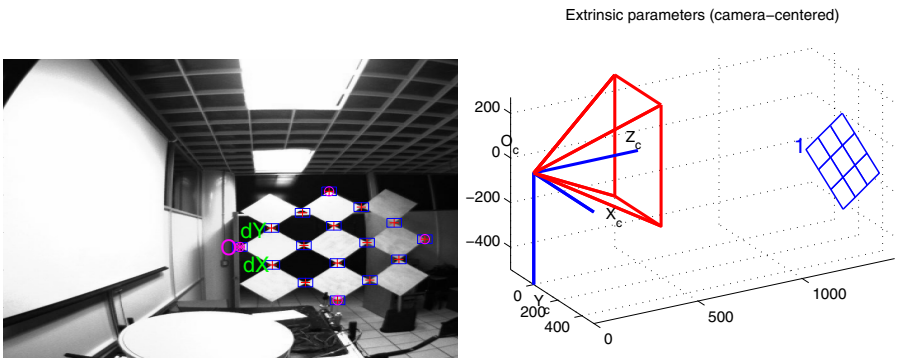
4.2 Algorithm for Computing Extrinsic and Intrinsic *Ladybug* Parameters

We use camera calibration toolbox for Matlab [3]. The intrinsic camera parameters are:

intrinsic parameters	value
Focal length	(628.4651, 622.5191) px
Principal point location	(348.0818, 507.8548) px
Distortion coefficients	(-0.3759,0.1139,0.0027, 0.0049,0) px^2
Skew coefficient	0

The Figure 7(a) shows the subpixelic extraction points. The pattern plane position is the same for the pattern plane acquired by the LIDAR in the Figure 5. The world frame is shown in the Figure 7(b) with respect to the camera frame. The $[R, \mathbf{T}]_W^{C_i}$ are:

$$R_W^{C_i} = \begin{bmatrix} 0.6588 & 0.6417 & -0.3926 \\ 0.6257 & -0.7572 & -0.1877 \\ -0.4178 & -0.1220 & -0.9003 \end{bmatrix}; \mathbf{T}_W^{C_i} = \begin{bmatrix} 35.0 \\ 76.35 \\ 365.36 \end{bmatrix}$$



(a) (b)

Fig. 7. (a) Subpixel corners extration. (b) World frame respect to the camera frame.

4.3 LIDAR and Ladybug Calibration

The rigid transformation between camera and LIDAR frame is computed using the Equation 2. The result is shown in Figure 8. In this figure, the pattern acquired by the LIDAR is transformed onto the image frame using the extrinsic parameters $[R_L^{C_i}, \mathbf{T}_L^{C_i}]$, this transformation allows us referenced in the camera the points acquired by the LIDAR. The projection is completed using the intrinsic camera parameters. The Figure 9 demostrate the calibration system. The Figure 9 (a) shows the LIDAR data that correspond a person, the 3D data are projected into the image. The projected points are show in Figure 9 (b) as red points.

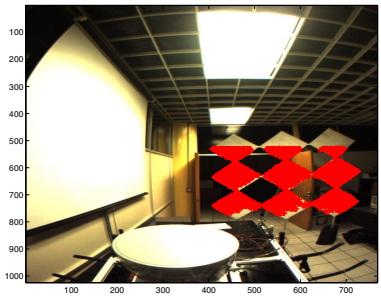
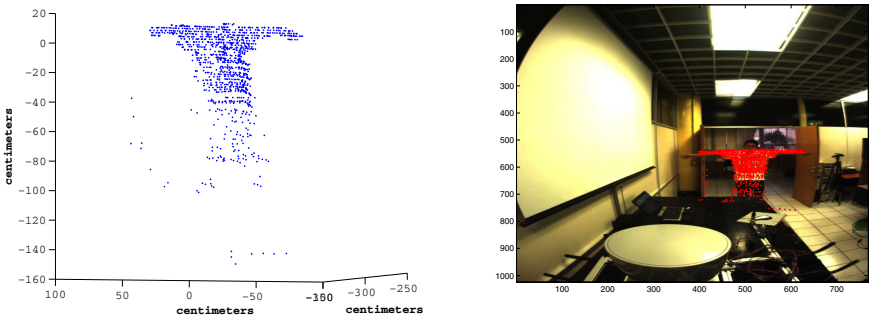


Fig. 8. The red points are acquired by the LIDAR and projected to image



(a) (b)

Fig. 9. (a) A person acquired by the LIDAR. (b) The red points correspond the 3D points projected into image.

5 Conclusions and Future Work

A new extrinsic calibration method for a multiple sensor has been proposed. By using a plane calibration target, extrinsic calibration between sensors and intrinsic camera calibration can be effectuated simultaneously. The results obtained in real data tests illustrate an appropriate and accurate projection of the lidar data. The future work are oriented to estimate the confidence intervals in the calibration method and the error propagation in data sensor fusion for texture 3D data.

References

1. Almeida, M., Dias, P., Oliveira, M., Santos, V.: 3d-2d laser range finder calibration using a conic based geometry shape. In: Image Analysis and Recognition, pp. 312–319 (2012)

2. Atanacio-Jiménez, G., González-Barbosa, J.J., Hurtado-Ramos, J.B., Ornelas-Rodríguez, F.J., Jiménez-Hernández, T., García-Ramirez, H., González-Barbosa, R.: Lidar velodyne hdl-64e calibration using pattern planes. International Journal of Advanced Robotic Systems 8(5), 70–82 (2011)
3. Bouguet, J.Y.: Camera calibration toolbox for Matlab (2008)
4. Mei, C., Rives, P.: Calibration between a central catadioptric camera and a laser range finder for robotic applications. In: Proceedings 2006 IEEE International Conference on Robotics and Automation, ICRA 2006, pp. 532–537. IEEE (2006)
5. Mirzaei, F.M., Kottas, D.G., Roumeliotis, S.I.: 3d lidar-camera intrinsic and extrinsic calibration: Observability analysis and analytical least squares-based initialization. International Journal of Robotics Research 31(4), 452–467 (2012)
6. Nunez, P., Drews Jr, P., Rocha, R., Dias, J.: Data fusion calibration for a 3d laser range finder and a camera using inertial data. In: European Conference on Mobile Robots, vol. 9, p. 9 (2009)
7. Rodriguez, F., Fremont, V., Bonnifait, P., et al.: Extrinsic calibration between a multi-layer lidar and a camera. In: IEEE International Conference on Multisensor Fusion and Integration for Intelligent Systems, MFI 2008, pp. 214–219. IEEE (2008)
8. Schneider, S., Himmelsbach, M., Luettel, T., Wuensche, H.J.: Fusing vision and lidar-synchronization, correction and occlusion reasoning. In: 2010 IEEE Intelligent Vehicles Symposium (IV), pp. 388–393 (2010)
9. Underwood, J., Scheding, S., Ramos, F.: Real-time map building with uncertainty using colour camera and scanning laser. In: Proceedings of the 2007 Australasian Conference on Robotics and Automation (2007)
10. Zhang, Z.: A flexible new technique for camera calibration. IEEE Transactions on Pattern Analysis and Machine Intelligence, 1330–1334 (2000)
11. Zhao, G., Xiao, X., Yuan, J.: Fusion of velodyne and camera data for scene parsing. In: 2012 15th International Conference on Information Fusion (FUSION), pp. 1172–1179 (2012)

Video Images Fusion to Improve Iris Recognition Accuracy in Unconstrained Environments

Juan M. Colores-Vargas[1], Mireya García-Vázquez[1], Alejandro Ramírez-Acosta[2], Héctor Pérez-Meana[3], and Mariko Nakano - Miyatake[3]

[1] Instituto Politécnico Nacional, CITEDI
Avenida del Parque 1310, Tijuana, B.C. México 22510
[2] MIRAL R&D, Palm Garden, Imperial Beach, USA 91932
[3] Instituto Politécnico Nacional, ESIME-Culhuacan, DF., México
colores@citedi.mx, msarai@ipn.mx, ramacos10@hotmail.com,
{mnakano,hmperezm}@ipn.mx

Abstract. To date, research on the iris recognition systems are focused on the optimization and proposals of new stages for uncontrolled environment systems to improve the recognition rate levels. In this paper we propose to exploit the biometric information from video-iris, creating a fusioned normalized template through an image fusion technique. Indeed, this method merges the biometric features of a group of video images getting an enhanced image which therefore improves the recognition rates iris, in terms of Hamming distance, in an uncontrolled environment system. We analyzed seven different methods based on pixel-level and multi-resolution fusion techniques on a subset of images from the MBGC.v2 database. The experimental results show that the PCA method presents the best performance to improve recognition values according to the Hamming distances in 83% of the experiments.

Keywords: Fusion, Iris, MBGC, PCA, Recognition.

1 Introduction

To date, the commercial iris recognition systems based on still images [1-2] are designed to work with special or restricted conditions. This means that they require an ideal environment and cooperative user's behavior during the iris image acquisition stage to obtain high quality images. Therefore, if any of these requirements are not met, it can cause a substantially increase of error rates, specially the false rejections. Many factors can affect the quality of an iris image, including defocus, motion blur, dilation and heavy occlusion. Naturally, poor image's quality cannot generate satisfactory recognition because they do not have enough feature information, in this regard; iris recognition is dependent on the amount of information available in two iris images being compared. A typical iris recognition system commonly consists of four main modules as shown in Figure 1:

- **Acquisition** the aim is to acquire a high quality image.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 114–125, 2013.

- **Preprocessing**, involves the segmentation and normalization processes. The segmentation consists in isolating the iris region from the eye image. The normalization is used to compensate the varying size of the pupil.
- **Feature encoding**, uses texture analysis method to extract features from the normalized iris image. The significant features of the iris are extracted as a series of binary codes known as digital biometric template.
- **Matching** compares the user digital biometric template with all the stored templates in the database. The matching metric will give a range of values of the compared templates from the same iris.

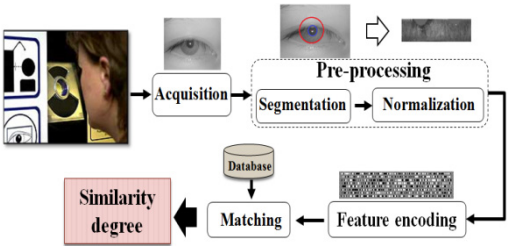


Fig. 1. Architecture for iris recognition system

In recent years, with an increasing of new massive biometric security demands around the world, it seems difficult to fulfill the conditions mentioned above in order to have a reliable iris recognition system [3-4]. Thus, with the aim of overcoming these drawbacks, news approaches are being developed in an attempt to improve iris recognition performance under non ideal situations. Among these approaches, the video-based eye image acquisition for iris recognition seems to be an interesting alternative [5-7] because it can provide more information through the capture of a video iris sequences. Besides that, it is a friendly system because it is not intrusive and requires few users' cooperation. In this paper, we propose to use the video-iris. It contains information related to the spatio-temporal activity of the iris and its neighbor region over a short period of time, such that, iris images can be selectively chosen for fusion while avoiding poor quality images. Therefore, the information from individual iris images can be fused into a single composite iris image with higher biometric information, resulting in better recognition performance and reducing the error rates.

The idea of fusing iris templates to perform biometric recognition has been recently described in the literature. Jillela and Ross [8] fused frames of iris videos using a variation of the PCA method with three data-level resolution, the performance of the image-level fusion was compared against that of score-level fusion, they observe significant improvements of the proposed technique, when compared to the use of any individual frame. Hollingsworth et al. [5] improve the matching performance using signal-level fusion. From multiple frames of iris video, they created a single average image, having observed that signal-level fusion performs comparably to state-of-the-art score-level fusion techniques. There are several methods of fusion, the main objective in this research is to experiment and analyze some fusion methods to determine the most suitable to be included as a step in the recognition system for unconstrained environments. This paper is organized as

follows: section 2 explains the basics of the evaluated image fusion methods. Section 3 presents the evaluation methodology and the evaluation results, finally in section 4 presents the conclusions and future work.

2 Image Fusion

The image fusion tries to solve the problem of combining information from several images taken the same object to get a new fusioned [9]. In this paper, each image of video is first pre-processed in order to obtain a single normalized template. Then, a fusion method is applied to provide a representative fusioned normalized template from these individual normalized templates. The resulting template should be contains more texture information as compared to individual templates. We analyzed the image fusion methods to determine the most suitable to achieve greater extraction of texture information. The image fusion methods can be categorized in two categories: pixel level and multi-resolution.

- **Pixel-level methods,** the input images are fused pixel by pixel followed by the information extraction. To implement the pixel level fusion, arithmetic operations are widely used, include basic arithmetic operations, logical operations and probabilistic operations.
- **Multi-resolution methods,** Multi-scale Transform (MST) is applied on the original images to construct a composite representation followed by down-sampling. Then an image fusion rule is applied to fuse the images in the MST format. After that, an Inverse Multi-scale Transform (IMST) is applied to create the fused image.

2.1 Image Fusion Using the Weighted Average

The fusion method based on the weighted average is a pixel level method that using weights assigned to each original image, the weight may be fixed or variable based on specific applications; commonly using the sharpness analysis to assigns a higher weight to the sharpest pixels in the input images [10]. The weights for each source image are defined by two arrays W_1 and W_2, where $0 \leq W_1, W_2 \leq 1$ and $W_1(x, y) + W_2(x, y) = 1$, a resulting image is given by the equation 1.

$$I(x, y) = W_1(x, y)I_1(x, y) + W_2(x, y)I_2(x, y) \tag{1}$$

To calculate the weight array is used the information from the image edges obtained by applying high-pass filters which reflect the abrupt changes in the intensities of the pixels with respect to its environment (edges).

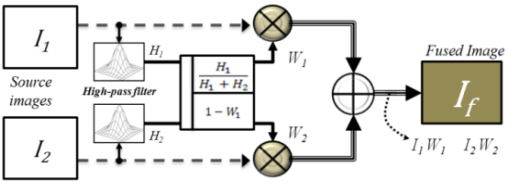


Fig. 2. Illustration of the weighted average fusion method

To obtain the related frequencies components of the edges in the images, was used a high-pass filter, is a Gaussian 3x3 convolution kernel that uses the concept of the second derivative defined by equation 2. Thus if H_1 contains the edges information of the picture I_1 and H_2 to the image I2, in both cases by the application of predefined kernel. The weights can be obtained as shown in Figure 2.

$$H(x,y) = |h * I(x,y)| \rightarrow h = \begin{bmatrix} -1 & -1 & -1 \\ -1 & 8 & -1 \\ -1 & -1 & -1 \end{bmatrix} \quad (2)$$

2.2 Image Fusion Using Principal Component Analysis

The fusion method based on the principal component analysis [11] is a straightforward way to build a fused image as a weighted superposition of several input images. The optimal weighting coefficients, with respect to information content, can be determined by a principal component analysis of all input intensities. By performing a PCA of the covariance matrix of input intensities, the weightings for each input image are obtained from the eigenvector corresponding to the largest eigen-value. Figure 3 shows the basic fusion scheme, where two images I_1 *and* I_2 are fused to obtain a resultant image I_f given by equation 3, W_1 *y* W_2 are the weights coefficients.

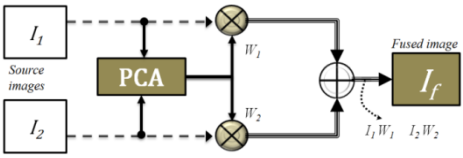


Fig. 3. PCA operation to fuse two images

$$I_f(x,y) = W_1(x,y)I_1(x,y) + W_2(x,y)I_2(x,y) \quad (3)$$

The weights for each source image are obtained from the eigenvector corresponding to the largest eigen-value of the covariance matrix of each source. Arrange source images in two-column vector.

- Organize the data, let S be the resulting column vector.
- Compute empirical mean (M_e) along each column.
- Subtract M_e from each column of S, the resulting is a matrix X.
- Find the covariance matrix C of matrix X.
- Compute the eigenvectors V and eigen-value and sort them by decreasing eigen-value.
- Consider first column which correspond to larger eigen-value to compute normalized component W_1 *and* W_2.

2.3 Laplacian Pyramid Image Fusion

Laplacian pyramid of an image is a set of band-pass images, in which each is a band-pass filtered copy of its predecessor [12]. Band-pass copies can be obtained by

calculating the difference between low-pass (Gaussian filter) images at successive levels of a Gaussian pyramid. In Laplacian fusion approach the Laplacian pyramids for input images are used. A strength measure is used to decide from which source what pixels contribute at each specific sample location. Given an image I, the first level of its Gaussian pyramid is defined as a copy of the original image and the k-th level is defined by the equation 4 and the k-th level of the Laplacian pyramid is defined by the equation 5, where $\downarrow 2$ and $\uparrow 2$ denotes a down-sampling and up-sampling at a factor of 2, w represent a 5x5 low-pass filter kernel

$$G_k(x,y) = [w * G_{k-1}(x,y)]_{\downarrow 2} \tag{4}$$

$$\tilde{L}_k(x,y) = G_k(x,y) - 4w * [G_{k-1}(x,y)]_{\uparrow 2} \rightarrow w = \frac{1}{256}\begin{bmatrix} 1 & 4 & 6 & 4 & 1 \\ 4 & 16 & 24 & 16 & 4 \\ 6 & 24 & 36 & 24 & 6 \\ 4 & 16 & 24 & 16 & 4 \\ 1 & 4 & 6 & 4 & 1 \end{bmatrix} \tag{5}$$

To reconstruct the original image I from both pyramids (Laplacian and Gaussian), is used equation 6. The fusion of the low-pass coefficients involves blending all the Gaussian pyramid coefficients; the fusion of the high-pass coefficients involves blending only the level of the Laplacian pyramid. Zhang and Blum [13] proposed a combination method, where the coefficients are considered separately; the low pass coefficients are fused using the arithmetic mean and the high pass coefficients according with the biggest absolute value (see equations 7 and 8).

$$\hat{G}_k(x,y) = \tilde{L}_k(x,y) + 4w * [\hat{G}_{k+1}(x,y)]_{\uparrow 2} \tag{6}$$

$$G_k^C(x,y) = \frac{G_k^A(x,y) + G_k^B(x,y)}{2} \tag{7}$$

$$\tilde{L}_k^C(x,y) = \begin{cases} \tilde{L}_k^A(x,y) & if \quad \left|\tilde{L}_k^A(x,y)\right| > \left|\tilde{L}_k^B(x,y)\right| \\ \tilde{L}_k^B(x,y) & otherwise \end{cases} \tag{8}$$

2.4 Contrast Pyramid Image Fusion

The composite images produced by this scheme preserve those details from the input images that are most relevant to visual perception [14].The essential problem in fusion images for visual display is pattern conservation: important details of the component images must be preserved in the resulting fused image, while the fusion process must not introduce spurious pattern elements that could interfere with succeeding analysis. Contrast itself is defined as the ratio of the difference between luminance at a certain location in the image plane and local background luminance to the local background luminance. Luminance is defined as the quantitative measure of brightness and is the amount of visible light energy leaving a point on a surface in a given direction. The construction of the Contrast pyramid is similar to the construction of the Laplacian pyramid. First a Gaussian pyramid is constructed by equation 9, which describes the k-th level of the pyramid and the reconstruction is defined by Equation 10. The coefficients fusion methodology employs a method similar to the Laplacian pyramid (see equation 7). However, to fuse the Laplacian coefficients are used the absolute maximum criterion described by Equation 11.

$$R_k(x,y) = G_k(x,y)/4w * [G_{k-1}(x,y)]_{\uparrow 2} \tag{9}$$

$$\hat{G}_k(x,y) = R_k(x,y) * 4w * [\hat{G}_{k+1}(x,y)]_{\uparrow 2} \tag{10}$$

$$R_k^C(x,y) = \begin{cases} R_k^A(x,y) & if \quad \left|R_k^A(x,y) - 1\right| > \left|R_k^B(x,y) - 1\right| \\ R_k^B(x,y) & otherwise \end{cases} \tag{11}$$

2.5 Gradient Pyramid Image Fusion

The gradient pyramid is emerging as a variation to the Laplacian pyramid. A gradient pyramid is obtained by applying a set of 4 directional gradient filters (horizontal, vertical and 2 diagonal) to the Gaussian pyramid at each level. At each level, these 4 directional gradient pyramids are combined together to obtain a combined gradient pyramid that is similar to a Laplacian pyramid. The gradient pyramid fusion is therefore the same as the fusion using Laplacian pyramid except replacing the Laplacian pyramid with the combined gradient. The k-th level with l orientation of the gradient pyramid is defined by equation 12. G_k is the k-th level of the Gaussian pyramid, d_l is the gradient filter for the l orientation and w is a Gaussian filter kernel described by equation 12, the gradient filters are given by equations (13-16).

$$D_{k,l}(x,y) = d_l * [G_k(x,y) + \dot{w}G_k(x,y)] \rightarrow \dot{w} = \frac{1}{16}\begin{bmatrix} 1 & 2 & 1 \\ 2 & 4 & 2 \\ 1 & 2 & 1 \end{bmatrix} \tag{12}$$

$$d_1 = [1 \quad -1], \qquad d_2 = \frac{1}{\sqrt{2}}\begin{bmatrix} 0 & -1 \\ 1 & 0 \end{bmatrix}, \qquad d_3 = \begin{bmatrix} -1 \\ 1 \end{bmatrix}, \qquad d_4 = \frac{1}{\sqrt{2}}\begin{bmatrix} -1 & 0 \\ 0 & 1 \end{bmatrix} \tag{13-16}$$

To reconstruct the image from the gradient pyramids is used equation 17. The fusion of coefficients involves the same methodology of the Laplacian pyramid.

$$\tilde{L}_k(x,y) = [1+w] * \sum_{l=1}^{4}(-\frac{1}{8}d_l * D_{k,l}(x,y)) \tag{17}$$

2.6 FSD Pyramid Image Fusion

FSD (filter, subtract, decimate) pyramid technique [15], is a variation of the Laplacian fusion. In Laplacian pyramid, the difference image L_k at level k is obtained by subtracting an image up-sampled and then low-pass filtered at level $k+1$ from the Gaussian image G_k at level k, while in FSD pyramid, this difference image is obtained directly from the Gaussian image G_k at level k subtracted by the low-pass filtered image of G_k as a result, FSD pyramid fusion method is computationally more efficient than the Laplacian pyramid method by skipping an up-sampling step.

$$L_k(x,y) = G_k(x,y) - w * G_k(x,y) \tag{18}$$

$$\hat{G}_k(x,y) = L_k(x,y) + w * (L_k(x,y) + \left[4\hat{G}_{k+1}(x,y)\right]_{\uparrow 2}) \tag{19}$$

The calculation of the k-th level of the Laplacian pyramid is computationally more efficient (see equation 18), however this variation restricts to use the equation 6. Thus it is necessary to define a different reconstruction formulation given by equation 19. The coefficients fusion uses the same methodology of the Laplacian pyramid.

2.7 Image Fusion Based on Wavelet Decomposition

The Wavelet transform decomposes an image into various sub images based on local frequency content [16]. The discrete Wavelet transform (DWT) coefficients are computed by using a series of low pass filter L[k], high pass filters H[k] and down samplers across both rows and columns. The results are the wavelet coefficient the next scale. The filter bank approach to calculate two dimensional dyadic DWT is shown in figure 4. The wavelet coefficients are of smaller spatial resolution as they go from finer scale to coarser scale. The wavelet representation has the advantage of not generating redundant information since functions are orthogonal, thus original image can be reconstructed from the wavelet decomposition of an inverse algorithm.

According with equations (20-23), we can define the k-th level of the pyramid of Wavelets, where $(1 \downarrow 2)$ is a down-sampling to remove half of the rows of the image and $(2 \downarrow 1)$ to remove half of the columns of the image. The pyramid is constructed by applying this decomposition recursively on approximation coefficients. To reconstruct the image, we must apply the inverse transformations.

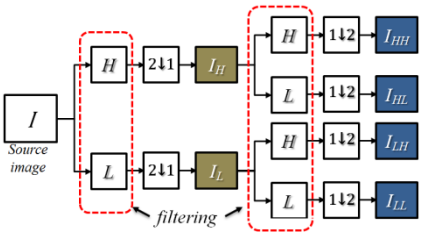


Fig. 4. Two-dimensional orthogonal Wavelet fusion

$$LL_k(x,y) = [w_L * [w_L * I_k(x,y)]_{2\downarrow1}]_{1\downarrow2} | W_L = \frac{1}{\sqrt{2}}[1 \quad 1] \tag{20}$$

$$LH_k(x,y) = [w_H * [w_L * I_k(x,y)]_{2\downarrow1}]_{1\downarrow2} | W_H = \frac{1}{\sqrt{2}}[-1 \quad 1] \tag{21}$$

$$HL_k(x,y) = [w_L * [w_H * I_k(x,y)]_{2\downarrow1}]_{1\downarrow2} | W_L = \frac{1}{\sqrt{2}}[1 \quad 1] \tag{22}$$

$$HH_k(x,y) = [w_L * [w_H * I_k(x,y)]_{2\downarrow1}]_{1\downarrow2} | W_H = \frac{1}{\sqrt{2}}[-1 \quad 1] \tag{23}$$

Finally, the reconstructed image is obtained from Equation 24.

$$\hat{I} = LL_k^{-1}(x,y) + LH_k^{-1}(x,y) + HL_k^{-1}(x,y) + HH_k^{-1}(x,y) \tag{24}$$

3 Experimental Results

3.1 Database

In order to evaluate the performance of the fusion methods, were handled a test set with 120 iris images from different videos of the Multiple Biometrics Grand Challenge “MBGC.v2” [17] database. All videos were acquired using an LG2200 camera with near-infrared illumination of the eye, some features: MPEG-4 format and the size for each frame in the video has 480 rows and 640 columns in 8 bits-grayscale space. The MBGC.v2 database presents noise factors, especially those relative to reflections, contrast, luminosity, eyelid and eyelash iris obstruction and focus characteristics. These facts make it the most appropriate to study the iris recognition system for uncontrolled environments. The test set is composed by 10 videos from 10 different people, for each video were randomly selected 10 images, analyzing and verifying that each image met the minimum quality parameters and segmentation rates exposed by Colores et al. [18]. In addition, we selected two reference images from each video for biometric comparison purposes, a reference image called "Reference 1" was chosen from the video based on the higher energy criterion [18], and the reference image called "Reference 2" was chosen based on the best subjective quality perceived. Each image in the set of test was segmented and normalized using the algorithms for iris recognition of Libor Masek [19], obtaining for each image a normalized template; this template contains only the texture information of the iris region. As mentioned in the first section, the recognition process is based on the value of the Hamming distance; this value reflects the correlation between two digital biometric templates. The digital biometric template is obtained from the normalized template by the encoding stage of an iris recognition system. In this sense, the Hamming distance will have a small value if digital biometric templates generated from the same iris (*comparison Intra-class*) are compared, or otherwise will have a value close to 1 (*comparison Inter-class*). In this paper, we are focused on improving the recognition by reducing the Hamming distance, this serves to reduce the chances of a false match. Hence, all possible Intra-class comparisons were performed on the test set, 100 comparisons for each reference image. In Figure 5, are illustrated Hamming distance values for different Intra-class comparisons, it can be seen that correlation values show a similar behavior in the two reference images, with average values of 0.2871 and 0.2863 for the image reference 1 and 2 respectively.

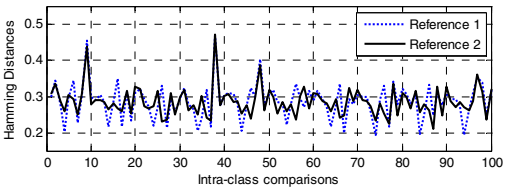


Fig. 5. Hamming distance values obtained from intra-class comparisons

3.2 Comparison of Image Fusion Schemes

The experimental analysis involves fusing pairs of normalized templates. Figure 6 illustrates the fusing process for normalized templates for each video. We calculated a total of 100 new fusioned normalized templates for each analyzed fusion method.

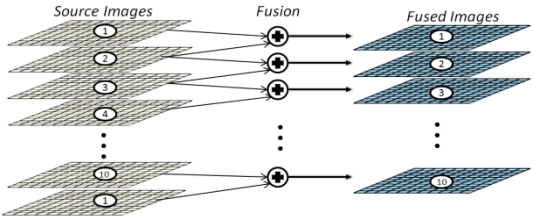


Fig. 6. Fusing process for normalized templates for each video

We implemented seven fusion methods described in section 2, intra-class comparisons were performed again with the 100 new digital biometric templates fused by each method, and the purpose was to determine the method that enhances the intra-class correlation values, as this could increase recognition rates by reducing the values of Hamming distances. Figure 7 shows the percentage of the experiments which had an improvement in the reduction of the Hamming distance for each method tested.

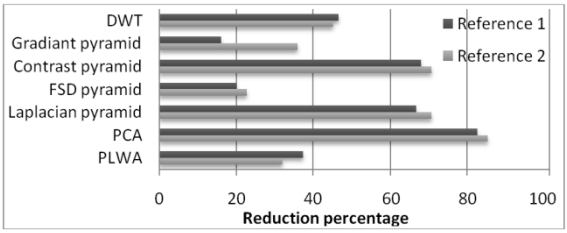


Fig. 7. Reduction percentage of Hamming distance for experiments with fusion methods

The fusion method based on principal component analysis PCA has the best performance compared to other methods; this does reduce the Hamming distance values in approximately 83% of the experiments. In Figures 8-9, are shown the variation between the Hamming distance values obtained from the intra-class comparison before and after to deploy the FSD and Gradient pyramid fusion methods. It can be appreciated that in these fusion methods, the majority of intra-class comparisons increased the Hamming distance values. Moreover, the fusion methods able to improve the correlation values in a high percentage of intra-class comparisons are performed by PCA, Laplacian pyramid and Gradient pyramid.

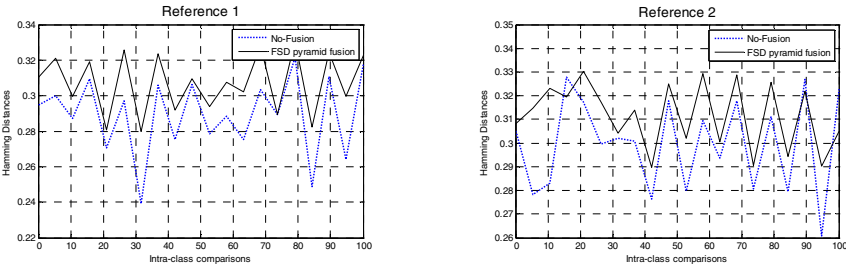


Fig. 8. Hamming distance values. After to deploy the FSD pyramid image fusion.

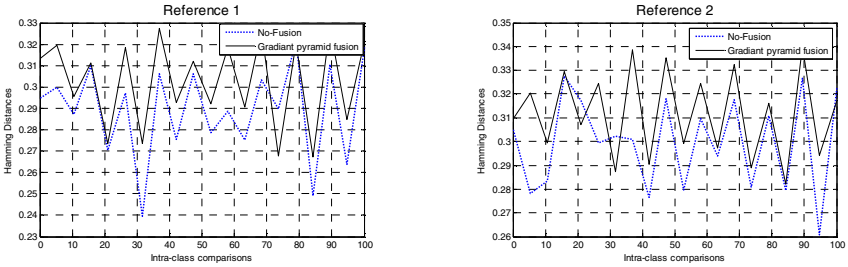


Fig. 9. Hamming distance values. After to deploy the Gradient pyramid image fusion.

Figure 10, shown the reduction in the Hamming distance values which were obtained by implementing the PCA fusion method. This method achieve the better percentage reduction in the experiments, reduces an average Hamming distance value in 0.015. Laplacian pyramid fusion methods and Contrast pyramid fusion method only reduce an average Hamming distance value at 0.0102. It is remarkable reduction in value of the Hamming distance at implement the PCA fusion method, which can project a possible reduction in the recognition error rates when implemented as a new module in a biometric system for uncontrolled environments.

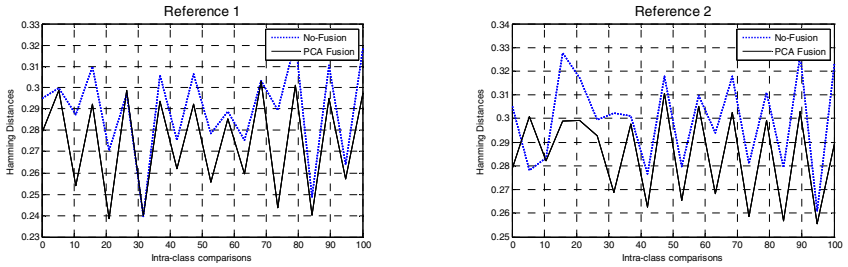


Fig. 10. Hamming distance values. After to deploy the PCA image fusion.

4 Conclusions

This paper proposes to exploit the biometric information from video-iris acquired under non-cooperative scheme, creating a fusioned normalized template through an

image fusion technique. The performances of seven image fusion methods are evaluated. The experimental results show that the PCA method presents the best performance to improve recognition values according to the Hamming distances in 83% of the experiments. In PCA, it is assumed that the information is carried in the variance of the features that is equivalent to walking around the data to see from which one gets the best information. In our opinion, the underlying reason is that PCA is more suitable to obtain the weights for iris images fusion, it's that analysis is based on small samples statistically independent (columns) of the source images data, resulting in a weights matrix more accurately. Thus, the results suggest that adding a fusion module to the architecture of the non-cooperative iris recognition, it could increase the system performance. Therefore, we can conclude that our proposal can be integrated as an optimization to the system developed by Colors et al [18], for an application of iris recognition in uncontrolled environments.

Acknowledgment. This research was supported by grant SIP 2013 from Instituto Politécnico Nacional.

References

1. Daugman, J.: How Iris Recognition Works. IEEE Transactions on Circuits and Systems for Video Technology 14, 21–30 (2004)
2. Wildes, R.: Iris Recognition: An Emerging Biometric Technology. Proceedings of the IEEE 85(9), 1348–1363 (1997)
3. Gamassi, M., Lazzaroni, M., Misino, M., Piuri, V.: Quality assessment of biometric systems: a comprehensive perspective based on accuracy and performance measurement. IEEE Trans. Instrum. Meas. 54, 1489–1496 (2005)
4. Vatsa, M., Singh, R., Gupta, P.: Comparison of Iris Recognition. In: International Conference on Intelligent Sensing and Information Processing, pp. 354–358 (2004)
5. Hollingsworth, K., Peters, T., Bowyer, K.: Iris recognition using signal-level fusion of frames from video. IEEE Transa. Inform. Forensics Secur. 4(4), 837–848 (2009)
6. Lee, Y., Phillips, P., Michaels, R.: An automated video-based system for iris recognition. In: Proc. Int. Conf. Biom., pp. 1–8 (2009)
7. Colores-Vargas, J., García, M., Ramírez, A., García, M., Nakano, M., Perez, H.: Iris recognition system based on video for unconstrained environments. Scientific Research and Essays 7(35), 3114–3127 (2012)
8. Jillela, R., Ross, A., Flynn, P.: Information Fusion in Low-resolution Iris Videos Using Principal Components Transform. In: Proceedings of IEEE Workshop on Applications of Computer Vision (WACV), Kona, USA (January 2011)
9. Mitchell, H., Singh, R., Gupta, P.: Multifocus Method for Controlling Depth of Field. Grafica Obscura (1994)
10. Haeberli, P., Singh, R., Gupta, P.: Image Fusion: Theories, Techniques and Applications. Springer, Heidelberg 2010 (2004)
11. Pajares, G., De la Cruz, J.: Visión por Computador: Imágenes Digitales y Aplicaciones. RA-MA, Madrid (2001)
12. Burt, P., Kolczynski, R.: Enhanced image capture through fusion. In: Proc. Fourth Int. Conf. on Computer Vision, pp. 173–182 (1993)

13. Zhang, Z., Blum, R.: A categorization of Multiscale-Decomposition-Based Image Fusion Schemes with a Performance Study for a Digital Camera Application. Proc. IEEE 87(8), 1315–1326 (1999)
14. Toet, A., van Ruyven, J., Valeton, J.: Merging termal and visual images by a contrast pyramid. Optical Engineering 28(7), 789–792 (1989)
15. Anderson, H.: A filter-subtract-decimate hierarchical pyramid signal analyzing and synthesizing technique. U.S. Patent 4.718 104 (1987)
16. Mallat, S.: A theory for multiresolution signal decomposition: the wavelet representation. IEEE Trans. Pattern Anal. Mach. Intell. 11, 674–693 (1989)
17. Multiple Biometric Grand Challenge, http://face.nist.gov/mbgc/
18. Colores-Vargas, J.M., García-Vázquez, M.S., Ramírez-Acosta, A.A.: Measurement of defocus level in iris images using convolution kernel method. In: Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Kittler, J. (eds.) MCPR 2010. LNCS, vol. 6256, pp. 125–133. Springer, Heidelberg (2010)
19. Masek, L.: Recognition of human iris patterns for biometric identification. Master's thesis, University of Western Australia (2003)

Perception Aptitude Improvement of an Odor Sensor: Model for a Biologically Inspired Nose

B. Lorena Villarreal[1] and J.L. Gordillo[2]

Center for Robotics and Intelligent Systems, Tecnológico de Monterrey
Monterrey, Nuevo León, México
lvillarreal.bg@gmail.com, jlgordillo@itesm.mx

Abstract. Biologically inspired systems are a common tendency in robotics. Nowadays the common robots use human-like behaving senses as capabilities as soon as they can see, hear and touch, but the senses of smell and taste, are starting to emerge. There are three main problems to solve when including a smell sensor into a robot: the environmental model or the way the odor molecules behave either in outdoors or indoors, the sensor model, and the algorithmic or process model. One of the difficulties of developing chemical sensors versus another sensor is that chemical reactions tend to change the sensor composition often in a way that is nonreversible. Also, the odor exposure quickly saturates the sensor which needs a lot of time to be ready for the next measure. This is why; the smell system design must be biologically inspired. In this paper we present the results of the sensor model including the biological inspired process of aspiration and the design of a smell system device.

Keywords: smell, bio-inspired nose model, nose system, nostrils, smell sense, chemical sensor model.

1 Introduction

Biologically inspired systems are a common tendency in robotics. Nature is a wide and wise teacher of how to reach specific goals. This is why; robotics and intelligent systems are trying to emulate behaviors and physical mechanisms out of it. For example there are artificial systems using algorithms based on animal behaviors to reach cooperation and coordination [1], odor path followers [2, 3], while there are also mechanical systems learning how to walk exactly as a human would do [4].

Nowadays there are a huge variety of robots emulating human senses as capabilities to reach its goals, as soon as they commonly can see, hear and touch, but the senses of smell and taste, are not commonly implemented. Although olfaction seems not to be essential for our survival, in fact, it is very important because if we lost this sense our lives becomes a little more dangerous, for example, we would not be alerted to spoiled foods, leaking gas or smoke from a fire [5].

In animal kingdom it is used for inspection, recognition, mating, hunting, and others. Due to this ability, dogs smell capability can be trained to search for drugs, explosives, some chemicals or hazardous substances or even lost people and are commonly

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 126–135, 2013.

used in search and rescue operations at disaster areas, at airports or border controls [6]. Smell sensors and robotics are starting to emerge as a unified area using emulations of the fast response of hound dogs in search and rescue operations [7-9]. At this task, the principal problem is the odor source localization. Nevertheless, to obtain the best algorithmic results, the perception of the environment is crucial.

There are three main problems to solve when including a smell sensor into a robot: the environmental model or the way the odor molecules behave either in outdoors or indoors, the sensor model, and the algorithmic or process model.

The main characteristics of commonly used smell sensors are the discrimination between odors and the sensitivity [10]. One of the difficulties of developing chemical sensors versus another sensor is that chemical reactions tend to change the sensor composition often in a way that is nonreversible. Also, the odor exposure quickly saturates the sensor which needs a lot of time to be restarted. There are some researchers trying to solve the saturation problem by other solutions [11].

Knowing that odor can be propagated without air flows present by diffusion, and by advection depending on the laminar air velocity, in a non-reversible process that increases the entropy, as the chemical reactions take place [12]; in this paper we are focusing on the perception problem, showing the results of the mathematical model for the sensor that includes as improvement the addition of the aspiration process biologically inspired, and the qualitative model for the sensorial system design.

In the following section we present the related work about smell and robotics applications, then on the third section the analysis of the biological nose, followed by the analysis of the bio-inspired sensorial model and the mathematical sensor model in section 4. In section 5 are presented the results. Finally on section 6 we present the conclusions and future work.

2 Related Background

2.1 Chemical Sensors Applications

Pashami et. al. in [13] propose a change point detection algorithm to find chances in intensity, compound, or mixture ratio of an odor. In this approach they evaluate it on individual gas sensors in an experimental setup where a gas source changes these variables.

But besides the discrimination of an odor, chemical sensors can be useful in a lot of more applications, for example: Using the concentration measurement at continuous time, the increment or decrement of certain odor could alert an oil plant about some leak of toxic or explosive gas and the corresponding time averaged behavior of it. This cannot be achieved if the sensor is saturated with a peak of that certain odor. This could lead to a false alarm and probably to a loss of time or money. In the other hand, the constant minimum increment that chemical sensors produce innately could lead to a constant replacement. In these applications it is important that odor sensors had the capability of desaturation.

The saturation of the sensor is a common problem in chemical sensors. Some research is focused to solve it [11]. They are using a multi-chamber system to decrease

the sampling time between sensors, but instead of it, we are trying to simplify the design by using only one chamber.

2.2 Smell and Robotics Applications

The principal task of a sniffing robot is the odor source localization. The algorithms used to solve this problem are commonly categorized by the terms of chemotaxis and anemotaxis. The first one refers to the orientation and movement of the agent in relation to the chemical gradient. In the other hand, anemotaxis, instead of following the gradient, considers the direction or current of a fluid [14].

Ishida and Miyatani in [15] are using pumps pulling air into a chamber with chemical sensors and creating an air curtain between two of them to produce two samples at different positions at the same time.

In [16] an odor grip map was presented, where the readings of the robot are convolved as soon as it moves using the radial symmetric two dimensional Gaussian function to create the next portion of the grid map. They used three different robot movement techniques: spiral, sweeping and instant gradient.

From this research, a hound robot was presented by Loutfi et. al. in [9]. This robot discriminates odors and creates grid maps of these using the sweeping movement technique. A major drawback of this approach is that the volatility of the odor particles causes them to be quickly distributed in the room so, by the time the robot has finished its sweeping routine, the distribution of the odor could be different.

Lochmatter and Martinoli [3, 17] developed three bio-inspired odor source localization algorithms, using a chemical sensor. The algorithms implemented in this research have the tendency of using the wind measurements to decide the best movement of the robot. They are also working on a simulation platform for the experimentation process [2].

At this point, the analysis of the sensor model and its efficient functionality is crucial before implement the odor source localization algorithms. This is why; this research is focused in the analysis, design and implementation of a biologically inspired sensor that generates a better measurement. The next section explains the human nose process to allow a complete understanding of the proposed design.

3 Biological Nose Functionality

According to [18], knowledge of the detailed air flow patterns in the human nasal cavity and the subsequent quantity of odorant transport to the olfactory receptor cites appear essential to a complete understanding of human olfaction. This is why the design of a biologically inspired smell sensor must considerate the air flow patterns to simulate complete olfaction.

The nose, in both, animals and humans, has two principal functions: breathing and odor perception [19]. Each nostril or nasal cavity is divided in three zones which are the nasal airway with bones into the lateral walls of the nasal cavity, forming a series of folds called turbinates; the epithelium, surrounded by an aqueous mucus layer; and

then the olfactory bulb is supported and protected by the cribriform plate (cartilage) which in mammals, separates it from the olfactory epithelium [20].

Odor perception starts with the transport of volatile chemical molecules (odor molecules) by air flow to the olfactory receptors inside the nose, specifically on the ciliated olfactory epithelium [21]. Then it is electrically transmitted to the olfactory bulb and ultimately to the brain [22].

The mucus helps to trap bacteria, particulate matter, and toxins. Conduction, filtration, heating and humidification are aided by mucosal and the turbinates [20].

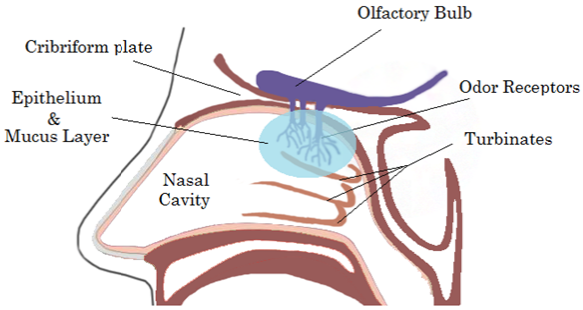


Fig. 1. Olfactory biological system of a human

There are three turbinates on each side of the nasal cavity and greatly increase the surface area of the nasal cavity. During inhalation air is directed over and under the turbinates. They are unconsciously controlled so that they swell on one side at a time and shrink on the other side. In humans, the turbinates divide the nasal airway and are responsible for forcing inhaled air to flow in a steady, regular pattern around the largest possible surface of cilia and climate-controlling tissue [20].

Upon entering the nasal cavity, inspiratory airflow is well mixed by turbulence, ensuring delivery of a representative odor sample to ephitelium [19]. It has been estimated that only 10% of inhaled air actually reaches the olfactory region during a normal resting breath [23].

This homogenized sample that reaches the odor sensors, send the electrical signal to the olfactory bulb, where data is pre-processed by glomeruli synapses [24] and the odors are discriminated. Finally the electrical information is send to the neocortex [22], where further processing takes place.

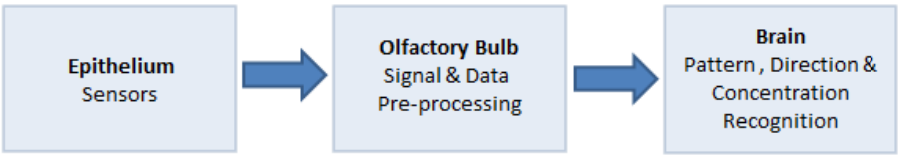


Fig. 2. Olfactory processes taking place in the biological nose

4 Perception System

As said before, the intention of this paper is to prove that the biological nose system can be emulated and implemented, improving the perception aptitude of chemical sensors thanks to the aspiration process generating a desaturation capability. At this point it is important to understand the implementation and the sensor behavior. This section describes the design and sensor model with and without the aspiration process.

4.1 Bio-inspired Design

Knowing that the smell sense in nature is divided in stages, we implemented a biologically inspired design of a nostril [25, 26]:

1. Aspiration process. At this stage the odor is carried by an air flow being absorbed by a nostril. In this process there must be a physical filter to protect the sensors from dust and other particles. The humidification of the sensor is also necessary to amplify the odor.
2. Conduction. At this stage the turbinates homogenize the air and mix the odor in such a way that the mix is prepared to be sensed. Then a good sample of the odor is conducted to the sensory array. The other part of the air is used in respiration.
3. Sensing. In this stage each array consists of three sensors separated by 2.5 cm from center to center in a linear position.
4. Processing. This is the algorithmic stage of the smell sense. It means that the data collected is sent to the brain which analyzes the information.
5. Transforming. The air is transformed to clean the sensory system so the saturation is not occurring.
6. Exhalation. The last stage is the expulsion of the air through the nose and the ventilation of all the system.

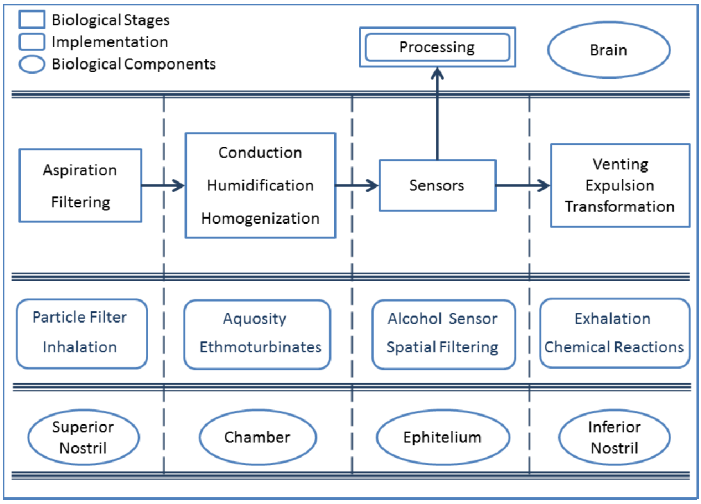


Fig. 3. Comparison between olfaction stages and processes, biological and implemented

All these stages are included in the smell system developed in this research to provide an efficient and optimized measurement. The complete system has the ability to inhale and exhale, using two ventilators in opposite direction, one for pull the air into the chamber, and one to send it out. After the measurement, the chamber is cleaned. It has different turbinates, that transport the odors at the same time that is being mixed, and then a sample of this mix is oriented to the sensory system. This system provides the homogenization and sampling of environment besides the results in section 5.

4.2 Sensor Model

The aim of a model is to represent the predicted behavior of a system in a logical, simplified and objective way depending on an input. In this research we are using an odor exposure as the input, and improving the model described in [27] by the inclusion of the aspiration process. A model is also useful to understand the behavior of the system by the analysis of the results obtained with the modification of common independent variables.

The gas sensor selected discriminates alcohol from any other odor. The chemical reaction changes the resistance between two terminals and because the supply voltage is maintained the current over the load changes and generates a different voltage on it. Then the results of the three sensors in the array for each nostril, are averaged and this average V_{in} [28] is the one that would be measured and analyzed. The concentration change (% C) was calculated as in [2]:

$$\%C = (\Delta R)/(V_{max} - R)\,100 \tag{1}$$

Where V_{max} is the maximum voltage ever measured, R or Reference is the lower voltage ever measured in absence of alcohol, and ΔR or differential range is the difference between averaged V_{in} and R.

$$r(t) = \begin{cases} r_1(t) & if\ t < t_s \\ r_2(t) & if\ t_s < t < t_s + \Delta t_r \\ r_3(t) & if\ t_s + \Delta t_r < t < t_s + \Delta t_a + \Delta t_r \\ r_4(t) & if\ t_s + \Delta t_r + \Delta t_a < t \end{cases} \tag{2}$$

$$r_1(t) = \mathrm{R}_0 \tag{3}$$

$$r_2(t) = R_0 + (R_{max} - R_0)(1 - \exp\left(\frac{-(t-t_s)}{\tau_r}\right)) \tag{4}$$

$$r_3(t) = R^*_{max} + \tau_a(t - t_s - \Delta t_r)(1 - \frac{(t-t_s-\Delta t_r)}{2}) \tag{5}$$

$$r_4(t) = R'_0 + (R^*_a - R'_0)(\exp\left(\frac{-(t-t_s-\Delta t_r-\Delta t_a)}{\tau_d}\right)) \tag{6}$$

$$R^*_{max} = R_0 + (R_{max} - R_0)(1 - \exp(-\Delta t_r/\tau_r)) \tag{7}$$

$$R^*_a = R^*_{max} + \tau_a(t_s + \Delta t_r + \Delta t_a)(1 - \frac{(t_s+\Delta t_r+\Delta t_a)}{2}) \tag{8}$$

In the complete model we have different adjustable variables: the response level before (R_0) and after (R'_0) the stimulus, the saturation level (R_{max}), the maximum value reached (R^*_{max}), the value before exhalation (R^*_a), the time constants of rise (τ_r), sampling the air (τ_a), and decay (τ_d), the time before the sensor started to respond (t_s), the duration of the rising period (Δt_r) and the duration of the sampling period (Δt_a).

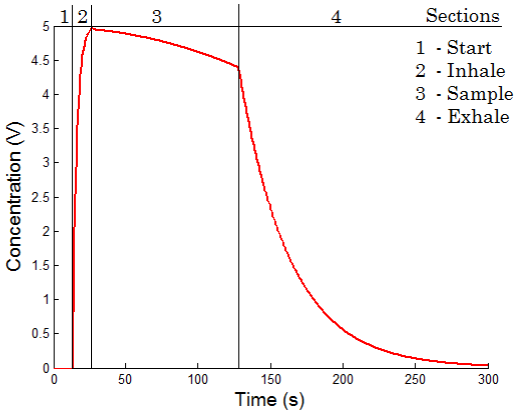


Fig. 4. Sensor model including inhalation, sampling and exhalation processes

Figure 4, displays the behavior of the sensor when an odor is exposed as a pulse, meaning that alcohol is exposed just an instant near the sensor. It is assumed an ideal first-order sensor and thus models the dynamic response to a pulse or step stimulus as an exponential rise and decay.

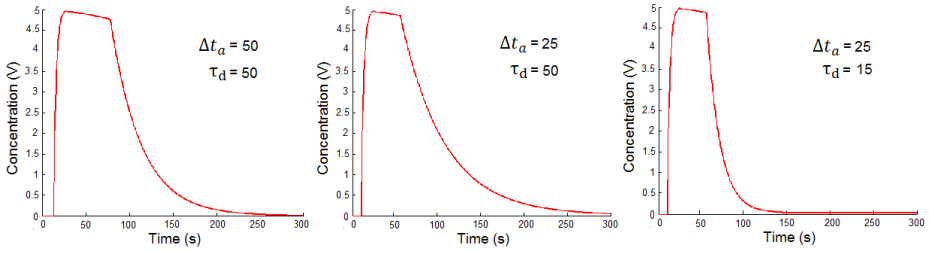


Fig. 5. Results with different sampling periods and decay time constants

It can be seen in Figure 5 that the decay time constant (τ_d) is not affected by decreasing the sampling period, instead, the system can sample in a short period, desaturating the sensor faster. To achieve this faster recovery of the sensors τ_d should be a lower value at exhalation.

5 Results

The experiments were realized in a robotics lab of 5x5 m2. The room was ambient temperature and flows of air were present in the room. The only restriction was to avoid transversal near floor currents through the path between the odor source and the sensors.

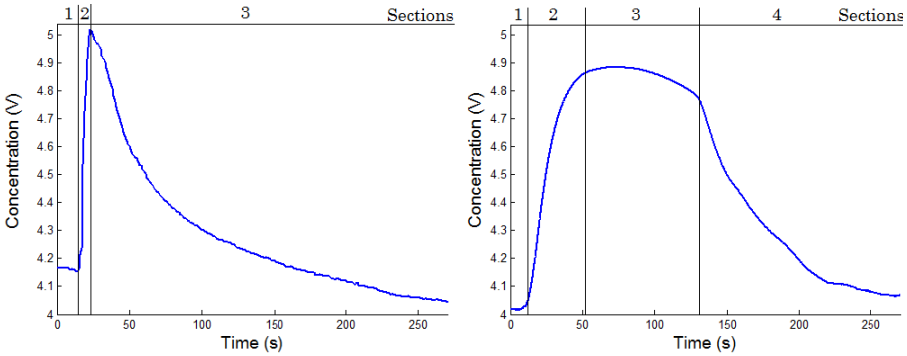


Fig. 6. Left: Sensor behavior without aspiration. Right: Sensor behavior with biologically inspired aspiration.

Three different processes were analyzed for aspiration including: inhalation, exhalation and the response when both ventilators are turned off.

The first set of experiments consisted in the comparison between the sensor with and without aspiration and smell system. Figure 6 shows the same pattern obtained by the mathematical model. It can also be noticed that desaturation process occurs faster with the aspiration model. It was obtained with the experimental data and the mathematical model a $\tau_d = 60.23$ when no aspiration is included and $\tau_d = 18.41$ otherwise, meaning that desaturation occurred around 70% faster with the biologically inspired nose system.

Table 1. Results of the rate of change of nostril measurements at different conditions

Aspiration Process	Environment	Rate	Behavior
Off	Odor Source	3 mV/s	Slow Rise
Exhale	Unsaturated	-15 mV/s	Fast Decay
Inhale	Odor Source	33 mV/s	Fast Rise
Off	Unsaturated	-0.3 mV/s	Static
Exhale	Odor Source	-5 mV/s	Slow Decay
Inhale	Unsaturated	1 mV/s	Very Slow Rise

Table 1 shows the comparative results of the rate of change in mV/s of the nostril measurements when an odor source is present and the ventilation system is off, inhaling or exhaling. And the same analysis when the odor source was turned off or closed.

These results show that aspiration is crucial when a sample of the environment is analyzed into the chamber; otherwise the sensors are saturated easily. Inhalation pull air into the nostril and the measure is reached faster, and exhalation desaturates the sensor even when an odor source is saturating the environment as a normal leak of gas.

6 Conclusions and Future Work

Thanks to the nostril design, we were able to concentrate the odor molecules near the sensors. And in the other hand, the aspiration system also helps in the desaturation of the sensors and makes them ready for the next measure.

Once we have a biologically inspired system that perceives the environment and obtain a homogenized and desaturated sample, the next goal is to use different sensors or array of sensors to discriminate more than one odor finding patterns in the environment of different mixtures.

In particular, we aim to apply this system on unmanned vehicles that can locate the source of an odor autonomously or with minimal human intervention, even at outdoors using time and space average, probably following the gradient of the odor trail.

References

1. Yasuda, G.: A Distributed Autonomous Control Architecture for Synchronization and Coordination of Multiple Robot Systems. In: 2012 Proceedings of Sice. Annual Conference (Sice), pp. 1864–1869 (2012)
2. Lochmatter, T., Martinoli, A., Wani, M.X.C., Casasent, D., Kurgan, L., Hu, T., Hafeez, K.: Simulation Experiments with Bio-Inspired Algorithms for Odor Source Localization in Laminar Wind Flow. In: Proceedings of the Seventh International Conference on Machine Learning and Applications, pp. 437–443 (2008)
3. Lochmatter, T., Martinoli, A., Khatib, O.: Tracking Odor Plumes in a Laminar Wind Field with Bio-inspired Algorithms. In: Khatib, O., Kumar, V., Pappas, G.J. (eds.) Experimental Robotics. STAR, vol. 54, pp. 473–482. Springer, Heidelberg (2009)
4. Huang, Q., Yu, Z., Zhang, W., Duan, X., Huang, Y., Li, K.: Generation of Humanoid Walking Pattern Based on Human Walking Measurement. In: 2008 8th IEEE-Ras International Conference on Humanoid Robots (Humanoids 2008), pp. 99–104 (2008)
5. Goldstein, E.B.: Sensation and Perception. Wadsworth, Cengage Learning, Canada (2010)
6. Browne, C., Stafford, K., Fordham, R.: The use of scent-detection dogs. Irish Veterinary Journal 59, 97–104 (2006)
7. Li, J., Meng, Q., Wang, Y., Zeng, M.: Odor source localization using a mobile robot in outdoor airflow environments with a particle filter algorithm. Autonomous Robots 30, 281–292 (2011)

8. Loutfi, A., Coradeschi, S., Karlsson, L., Broxvall, M.: Putting Olfaction into Action: Anchoring Symbols to Sensor Data Using Olfaction and Planning (2005)
9. Loutfi, A., Coradeschi, S., Lilienthal, A., Gonzalez, J.: Gas distribution mapping of multiple odour sources using a mobile robot. Robotica 27, 311–319 (2009)
10. Gardner, J., Bartlett, P.: A brief-history of electronic noses. Sensors and Actuators B-Chemical 18, 211–220 (1994)
11. Gonzalez-Jimenez, J., Monroy, J., Blanco, J.: The Multi-Chamber Electronic Nose-An Improved Olfaction Sensor for Mobile Robotics. Sensors 11, 6145–6164 (2011)
12. Crank, J.: The mathematics of diffusion. Oxford Univerity Press (1976)
13. Pashami, S., Lilienthal, A., Trincavelli, M.: Detecting Changes of a Distant Gas Source with an Array of MOX Gas Sensors. Sensors 12, 16404–16419 (2012)
14. Kowadlo, G., Russell, R.A.: Robot Odor Localization: A Taxonomy and Survey. The International Journal of Robotics Research 27, 869–894 (2008)
15. Miyatani, I., Ishida, H.: Active Stereo Nose: Using Air Curtain to Enhance the Directivity. 2010 IEEE Sensors, 1522–1525 (2010)
16. Lilienthal, A., Duckett, T.: Building gas concentration gridmaps with a mobile robot. Robotics and Autonomous Systems 48, 3–16 (2004)
17. Thomas, L., Xavier, R., Alcherio, M.: Odor Source Localization with Mobile Robots. Bulletin of the Swiss Society for Automatic Control 46, 11–14 (2007)
18. Zhao, K., Scherer, P., Hajiloo, S., Dalton, P.: Effect of anatomy on human nasal air flow and odorant transport patterns: Implications for olfaction. Chemical Senses 29, 365–379 (2004)
19. Craven, B., Paterson, E., Settles, G.: The fluid dynamics of canine olfaction: unique nasal airflow patterns as an explanation of macrosmia. Journal of the Royal Society Interface 7, 933–943 (2010)
20. Reddy, S.S., Ryan, M.W.: Turbinate Dysfunction: Focus on the role of the inferior turbinates in nasal airway obstruction. UTMB, Dept. of Otolaryngology (2003)
21. Hornung, D.E., Mozell, M.M. (eds.): Accessibility of odorant molecules to the receptors, New York (1981)
22. Pearce, T.C., Schiffman, S.S., Nagle, H.T., Gardner, J.W.: Handbook of Machine Olfaction Electronic Nose Technology (2003)
23. Hahn, I., Scherer, P.W., Mozell, M.M.: Velocity profiles measured for airflow through a large-scale model of the human nasal cavity. Modeling Physiol. 75, 2273–2287 (1993)
24. Ganong, W.F.: Review of medical physiology. Prentice Hall (1997)
25. Villarreal, B.L., Hassard, C., Gordillo, J.L.: Finding the Direction of an Odor Source by Using Biologically Inspired Smell System. In: Pavón, J., Duque-Méndez, N.D., Fuentes-Fernández, R. (eds.) IBERAMIA 2012. LNCS, vol. 7637, pp. 551–560. Springer, Heidelberg (2012)
26. Villarreal, B.L., Gordillo, J.L.: Method and artificial olfactive system. IMPI Instituto Mexicano de la Propiedad Intelectual, Mx/a/2012/014508, México (2012)
27. Lilienthal, A., Duckett, T.: A stereo electronic nose for a mobile inspection robot. In: Proceedings of the IEEE International Workshop on Robotic Sensing
28. Villarreal, B.L., Gordillo, J.L.: Directional Aptitude Analysis in Odor Source Localization Techniques for Rescue Robots Applications. In: 10th Mexican International Conference on Artificial Intelligence (MICAI), pp. 109–114 (2011)

A New Gaze Analysis Based Soft-Biometric

Chiara Galdi[1], Michele Nappi[1], Daniel Riccio[2], Virginio Cantoni[3], and Marco Porta[3]

[1] Università degli Studi di Salerno,
via Ponte Don Melillo, 84084 Fisciano (Salerno) Italy
[2] Università degli Studi di Napoli Federico II,
via Cintia 21, 80126, Napoli
[3] Università degli Studi di Pavia,
via Ferrata 1, 27100 Pavia
chiara.galdi@gmail.com, mnappi@unisa.it, daniel.riccio@unina.it,
{virginio.cantoni,marco.porta}@unipv.it

Abstract. Soft Biometric traits are physical or behavioral human characteristics like skin color, eye color, gait, used by humans to distinguish their peers. However soft biometric characteristics lack in distinctiveness and permanence to identify an individual uniquely and reliably. In this paper a new Gaze Analysis based Soft-biometric (GAS) is investigated. The way an observer looks at a particular subject, was recorded with a remote eye tracker. Feature vectors were built for each observation and used for testing the system as a recognition system. The accuracy of the GAS system was assessed in terms of Receiving Operating Characteristic curves (ROC), Equal Error Rate (EER) and Cumulative Match Curve (CMC), and provided encouraging results.

Keywords: gaze analysis, soft biometrics, eye tracking.

1 Introduction

Most studies related to soft biometrics have to date considered the iris as a potential predictor for ethnicity and gender [1]. For example, Qiu et al. have used specific features in the texture of the iris to determine ethnicity—Asian vs. Caucasian [2]. A similar approach has been used to determine gender by Thomas et al. [3]. More recently, the problem of predicting both gender and ethnicity using iris patterns has been addressed by Lagree et al. in 2011 [9]. While undoubtedly related to soft biometrics, however, these studies do not actually aim at identification, but rather at recognizing a single character. On the other hand, in [10], iris color is actually used as a soft biometry for indexing, in order to obtain a subset of identities where actual recognition will be performed.

While all the above cited works relate to static aspects of the human eye, eye movements can also be exploited to infer biometrically significant anatomical characteristics of the oculomotor plant [11]. This paper, too, examines a dynamic aspect, with the purpose to assess the relevance of eye movement patterns as a soft biometry [12].

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 136–144, 2013.

Reliable data about eye movements can be obtained through an eye tracker, that is a device (similar to an ordinary LCD monitor in its appearance) able to detect the user's gaze, usually when looking at a screen [4]. Eye movements occur as sudden, almost instantaneous, "saccades" (lasting less than 100 ms), followed by "fixation" periods of about 100-600 ms, during which the eye is almost still. According to the so-called "Eye-Mind Hypothesis" [5], there is a direct correspondence between the user's gaze and his or her point of attention. As demonstrated in some experiments [6], while it is possible to move one's attention without shifting the gaze, it is not possible to move one's gaze without shifting the attention, which is therefore strictly correlated to eye behavior. In particular, the vision process can occur both *overtly* and *covertly* [7], and it is just this last "vision modality" that is strictly related to a person's cognitive and psychological processes. Potentially, proper eye parameters can thus be exploited for a soft biometry which identifies, if not always the specific individual, at least groups of possible persons or classes of cognitive and emotional states.

The "scanpath" (i.e. the sequence of fixations acquired by an eye tracker during a vision process) obtained from the observation of images or other kinds of content in both overt and covert vision processes is one of the main sources of information for biometrics based on eye behavior. For example, simple forms of explicit authentication require the user to fixate some areas of the screen in sequence, thus creating a sort of visual "PIN". Implicit authentication, in which the user can watch wherever he or she wants on the screen (for example when observing a photograph), is much more complex. Although it is very difficult to find one-to-one correspondences between eye data and the subject, soft biometrics is however possible (see for instance [8]). The focus of this paper is specifically on verifying the conjecture that the way an individual looks at an image—specifically, at a head shot—might be a personal distinctive feature, albeit possibly a weak one. In particular, a new Gaze Analysis based Soft-biometric (GAS) is investigated.

2 Data Acquisition and Processing

A specific device, namely Tobii 1750 remote eye tracker, was used for data acquisition. It integrates all necessary components (camera, infrared lighting, etc.) into a 17'' LCD monitor (1280×1024 resolution). Data have been processed in order to have compact representations easier to compare with each other. Images employed in the experiment as stimuli, represent human faces. Each headshot has been segmented into significant areas of interest (AOIs). In order to attempt identification, a metric is needed to quantify distance between vectors. Even if several metrics have been investigated (Euclidean, Jaccard, Spearman, Cityblock, Cosine, and Correlation), we only report results for the Euclidean and Cosine distances, as they provided the best recognition accuracy.

2.1 Data Acquisition

In the Tobii device, five NIR-LED (Near Infra-Red Light Emitting Diodes) light eyes up producing reflection patterns. An image sensor records pupil position and

corneal-reflections to determine eyes position and the gaze point. With an accuracy of 0.5 degrees and relatively high freedom of movements, the system is ideal for real-use settings, where it would be intolerable to constrain users too much in their activities. For correct use of the system, at least one eye (better if both) must stay within the field of view of the infrared camera, which can be represented as a box with size 20×15×20 cm placed about 60 cm from the screen.

The sampling frequency of the device is 50 Hz (that is, the user's gaze coordinates are acquired 50 times a second). The Tobii ClearView gaze recording software was employed to define stimuli (still images, slideshows, videos, etc. to be presented to the subject), record and manipulate gaze data. The Tobii ClearView software acquires 50 raw gaze coordinates per second, from which fixations are then obtained (characterized by coordinates, duration and timestamps). For the purpose of our experiments, a fixation was considered a sequence of successive samplings detected within a circle with a 30 pixel radius, for a minimum duration of 100 ms. The ClearView software also allows to obtain two useful graphical depictions, namely *gazeplots* and *hotspots*. While a gazeplot displays the sequence of fixations of a user, in the form of circles with areas proportional to fixation times, a hotspot uses color codes to highlight those screen portions in which there are high concentrations of fixations – and consequently have been watched most. Gazeplot circles are numbered, thus clearly indicating the fixation order. A further output of the eye tracker is the *gaze replay*, which dynamically shows the evolution of fixations and *saccades* (fast eye movements) with time. A system output example is given in Figure 1.

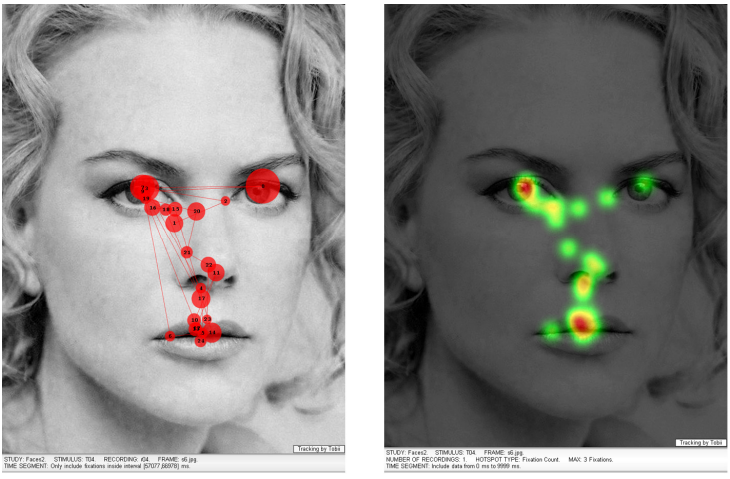


Fig. 1. A system output example: a gazeplot on the left and the corresponding hotspots on the right

2.2 Feature Extraction

As input images used as stimuli are all faces, it has been possible to define a limited set of AOIs which are always the same for all the observed photos, even if their

locations and dimensions can lightly change depending on the specific image. For this reason, GazePoint coordinates have been normalized and quantized according to the AOIs. Doing so, the size of the feature vector is fixed in advance and equals the number of AOIs. In particular, each AOI is assigned an element into the feature vector, which represents the total time spent looking at that AOI. It is worth noting that, at present, our feature vector does not take into account the number of visits for each AOI. If an observer spends some time on an AOI, quits and goes back on it later, only the total amount of time spent by the observer on that AOI is considered and computed by totaling up durations of both observations. Future enhancement of GAS strategy may involve the number of visits as an additional information to better characterize the observer behavior.

Figure 2 shows an example of a face sketch together with its AOIs marked with different colors and labeled with integer values.

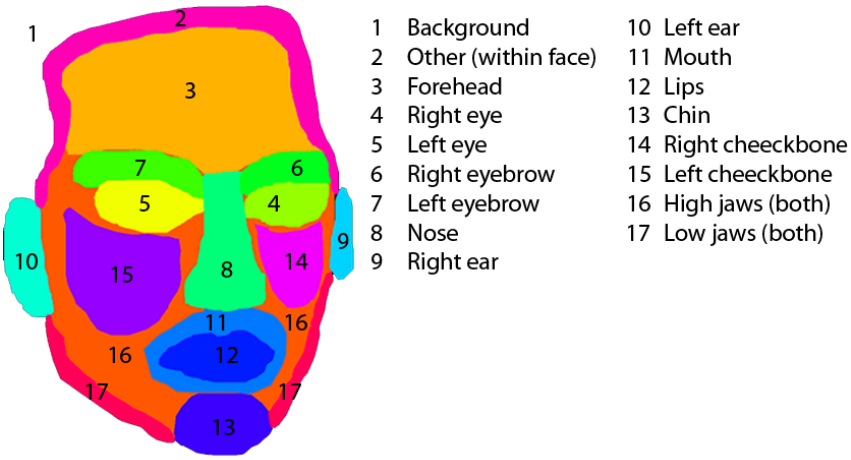


Fig. 2. A graphical sketch of the AOI mask

Doing so, feature vectors show a conveniently uniform representation that can be summed over observers as well as over subjects. That allows the fixation behavior of an observer to be modeled by averaging his or her feature vectors extracted from given set of face images used as training. In other words, the average feature vector over all training face images provides a description of the way that observer looks. In Figure 3, 16 feature vectors with a length of 17 AOIs are depicted for one of the observer involved in our experimentation (Observer 6). The bold line represents the average duration of fixations on each AOI computed over all 16 feature vectors. Different fusion strategies have been also investigated such as maximum, minimum and product rule; however the average value provided the best recognition accuracy in our experimental sessions.

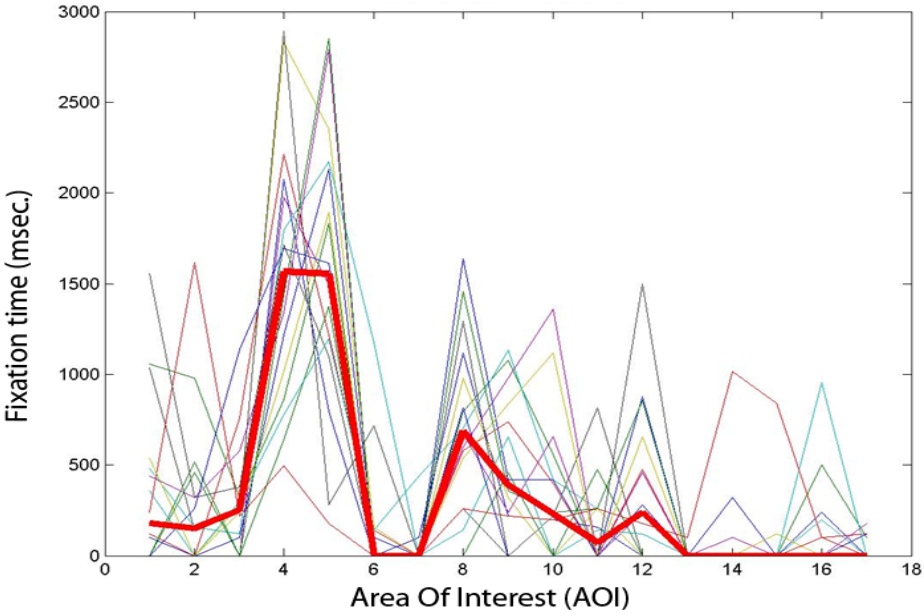


Fig. 3. Plots of feature vectors extracted from the gaze analysis of a given observer. Each vector refers to the observation of a single face image, while the (red) bold line represents the corresponding fixation model.

2.3 Matching

A small pool of face images that have been selected as training set is submitted to each observer and his/her fixation model is computed by averaging all corresponding feature vectors. A new observer is enrolled into the system by computing his/her fixation model, while the collection of all fixation models (one for each enrolled observer) constitutes the system gallery. When an observer asks for an authentication operation, a face image (different from that used for training) is submitted to him/her and the corresponding feature vector is then extracted. GAS computes the Euclidean distance between this vector and all fixation models into the system gallery. If the minimum distance is less than a fixed threshold δ, the system finds a match and the observer is associated with the identity corresponding to the fixation model that provided this minimum value. Otherwise the user is rejected by the system.

3 The Experimental Protocol

A total of 88 volunteer testers (59 males and 29 females) took part in the trials, subdivided into the following age groups: 17-18 (11 testers), 21-30 (39), 31-40 (7), 41-50 (15), 51-60 (7), 61-70 (8) and 71-80 (1). All participants reported normal or corrected-to-normal vision.

Prior to the beginning of the experiment, the observers were informed about the fact that some images, without specifying their kind, would appear on the eye trackers

display, interleaved with blank white screens with a small cross at their center. Each test was preceded by a short and simple calibration procedure, lasting about 10 seconds and consisting in following a moving circle on the screen. Participants were then instructed to look at the cross when the blank screen was displayed, and to freely watch wherever they wanted when the images were presented. The first blank screen was displayed for five seconds, while the others for three seconds. Each image was shown for ten seconds. 16 black-and-white pictures were used for the experiment, they contained close-up faces of 8 males and 8 females. Half of the faces (4 males and 4 females) were of famous persons (mostly actors and actresses), while the others were of people unknown to the testers.

The order of presentation of the 16 images was random. Behind the eye tracker there was a wall painted in neutral gray and the illumination of the room was uniform and constant. The tests were carried out in a quiet environment. On average, a single test session, including task explanation, device calibration etc., lasted a little more than five minutes.

36 of the 88 participants were involved in a second test session performed after the first one (with the same images), and 16 other participants were also involved in a third test session. The total number of tests carried out in the three sessions was therefore 140. Time intervals between the first and the second session, and between the second and the third session, ranged from a minimum of 5 days to a maximum of 9 days. As mentioned before, to compare observations Cosine and Euclidean metrics were used.

4 Experimental Results

The accuracy of the GAS system has been assessed in terms of Receiving Operating Characteristic curve (ROC), Equal Error Rate (EER) and Cumulative Match Curve (CMC). The ROC is a curve relating the Genuine Acceptance Rate and False Accepting Rate according to an acceptance threshold δ varying in the range [0,1]. The Equal Error Rate represents a sort of steady state for the system, as it corresponds to the point where False Acceptance Rate equals False Recognition Rate. The Cumulative Match Score at a rank n of a biometric identification system represents the likelihood that the correct identity is returned by the system among its top n answers. Thus the CMC is a curve representing the CMS with the rank ranging from 1 to N, where N is the number of enrolled subjects into the system gallery.

In all the following experiments, data acquired during the first session were used to enroll the observers, while observations captured during second and third sessions served as testing set.

In the first experiment only one feature vector has been used to build the fixation model of each observer, that is the fixation model coincides exactly with that feature vector. In particular, for each of the 16 face images in turn, the observation coming from the first session was considered as fixation model, while those acquired during second and third sessions were used as testing query. For each observer, results have been averaged over all 16 face images. Figure 4, shows the CMCs and ROCs obtained by averaging results over all the observers.

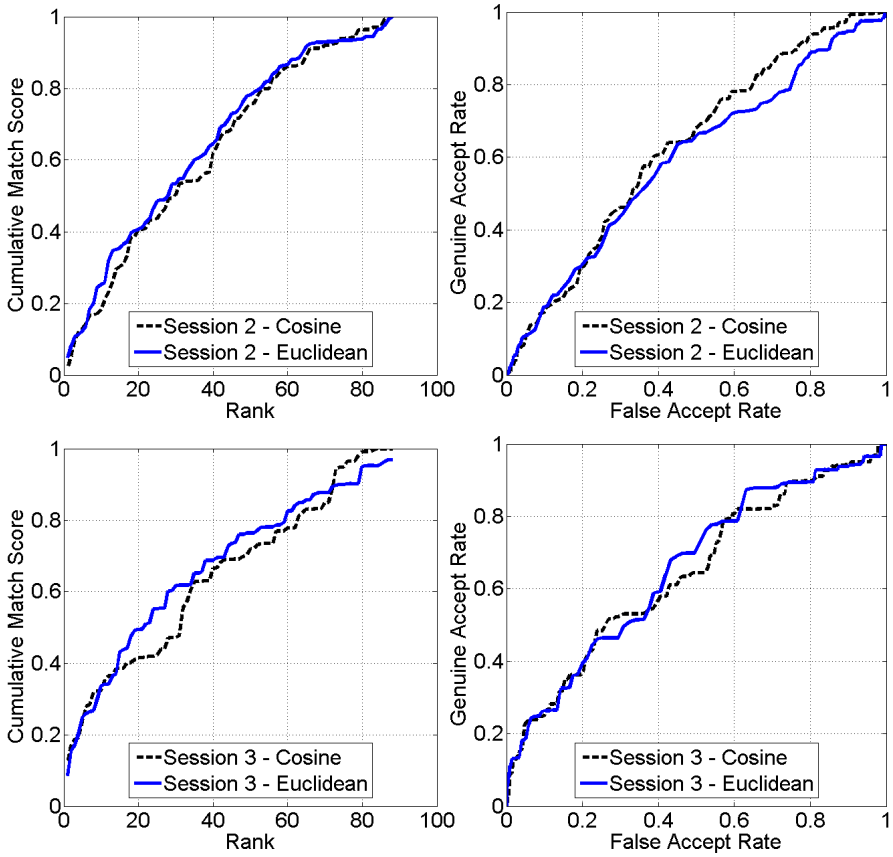


Fig. 4. CMC and ROC curves of the GAS system when only one feature vector is used to build the fixation model and the Euclidean/Cosine distance is used for matching

Looking at Curves in Figure 4 (upper left) and (upper right), it seems that Euclidean and Cosine distances reach comparable results. On the contrary, Figure 4 (lower left) and (lower right) show a small improvement in both CMC and ROC when the Euclidean distance is used. Session 2 and 3 mainly differ in the time gap from session 1 which was used to enroll the observers, thus pointing out the Euclidean distance to be more robust with respect to changes of feature vectors over time. This is also confirmed when looking at Equal Error Rates of the GAS system that are 0.404 and 0.415 for the Cosine and Euclidean distance respectively, when session 2 is used as probe and 0.432 and 0.355 when testing GAS with session 3. EERs also underline an improvement when using the Euclidean distance on session 3. For this reason, only the Euclidean distance is considered for next experiments.

In the second experiment, the fixation model of each user was built by using all 16 observations coming from the first session, while testing was performed by considering in turn one of the observations acquired during sessions 2 and 3

separately. Even in this case, results have been averaged over all face images belonging to the same session. Figure 5 reports the performance of GAS on session 2 and 3 when the Euclidean distance is used for matching.

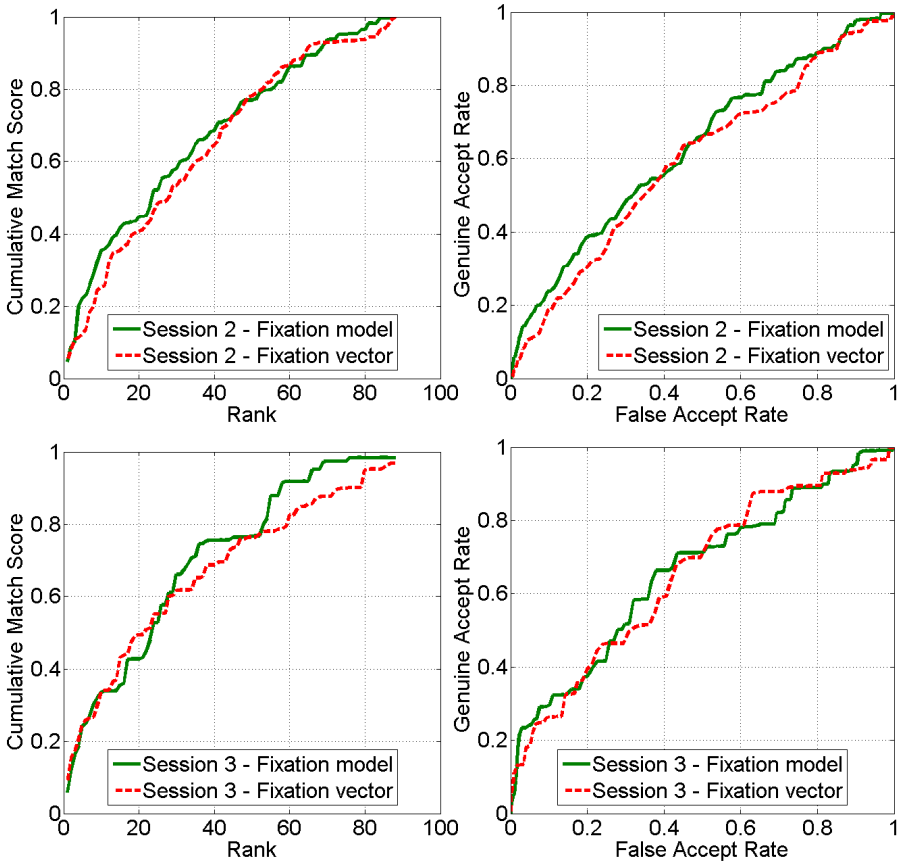


Fig. 5. CMC and ROC curves of the GAS system on session 2 and 3 when sixteen feature vectors (session 1) were used to build the fixation model and the Euclidean distance was used for matching

Figure 5 underlines that curves obtained exploiting the fixation model overcome those provided by a single feature vector in all cases. This is particularly noticeable for session 3 for which EER drops down from 0.408 (single feature vector) to 0.361 (fixation model). In other words, the fixation model better describes the way an observer looks a face image over time, since it is computed as the average of a pool of feature vectors that were used for training.

5 Conclusions

In this paper, the way an observer looks at a face image has been investigated as a potential new soft biometric trait by designing a new Gaze Analysis based

Soft-biometric system, namely GAS. Results obtained in terms of recognition accuracy confirm the feasibility of this line of research, even if many ways of improving the GAS architecture still remain to be tread. Increasing the number of AOIs by including some kind of "extra" points of interest (birthmarks, accessories, wrinkles) can take into account for further regions that are occasionally relevant in attracting the observer attention. The process of creation a mask of AOIs may also be automated by using a Facial Feature Point Detection algorithm, so that each fixation will be assigned with the closest Facial Feature Point. Last but not least, information coming from the temporal sequence of scanpaths and recurrent points of eye rest can be considered as additional features.

References

1. Porta, M., Ricotti, S., Jimenez Perez, C.: Emotional E-Learning through Eye Tracking. In: Proc. of the 2012 IEEE International Conference on Collaborative Learning, Marrakesh, Morocco, pp. 1–6 (2012)
2. Cantoni, V., Jimenez Perez, C., Porta, M., Ricotti, S.: Exploiting Eye Tracking in Advanced E-Learning Systems. In: Proceedings of the 13th International Conference on Computer Systems and Technologies (CompSysTech 2012), Rousse, Bulgaria, pp. 376–383 (2012)
3. Perego, E., Del Missier, F., Porta, M., Mosconi, M.: The Cognitive Effectiveness of Subtitle Processing. Media Psychology 13(3), 243–272 (2010)
4. Duchowski, A.T.: Eye Tracking Methodology – Theory and Practice, 2nd edn. Springer, London (2007)
5. Just, M.A., Carpenter, P.A.: Eye Fixations and Cognitive Processes. Cognitive Psychology 8, 441–480 (1976)
6. Poole, A., Ball, L.J.: Eye Tracking in Human-Computer Interaction and Usability Research: Current Status and Future Prospects. In: Gahoui, C. (ed.) Encyclopedia of Human-Computer Interaction, pp. 211–219. Idea Group (2006)
7. Itti, L., Koch, C.: A saliency-based search mechanism for overt and covert shifts of visual attention. Vision Research 40, 1489–1506 (2000)
8. Fookes, C., Maeder, A., Sridharan, S., Mamic, G.: Gaze Based Personal Identification. In: Wang, L., Geng, X. (eds.) Behavioral Biometrics for Human Identification: Intelligent Applications. IGI Global, Hershey (2010)
9. Lagree, S., Bowyer, K.: Predicting ethnicity and gender from iris texture. In: 2011 IEEE International Conference on Technologies for Homeland Security (HST), pp. 440–445 (November 2011)
10. Dantcheva, A., Erdogmus, N., Dugelay, J.-L.: On the reliability of eye color as a soft biometric trait. In: 2011 IEEE Workshop on Applications of Computer Vision (WACV), pp. 227–231 (January 2011)
11. Komogortsev, O.V., Karpov, A., Price, L., Aragon, C.: Biometric authentication via oculomotor plant characteristic. In: Proceedings of the IEEE/IARP International Conference on Biometrics (ICB), pp. 1–8 (2012)
12. Jain, A.K., Dass, S.C., Nandakumar, K.: Soft biometric traits for personal recognition systems. In: Proceedings of International Conference on Biometric Authentication, Hong Kong, pp. 731–738 (2004)

Evaluating Shape Descriptors for Detection of Maya Hieroglyphs

Edgar Roman-Rangel[1,⋆], Jean-Marc Odobez[2,3], and Daniel Gatica-Perez[2,3]

[1] University of Geneva, Switzerland
[2] Idiap Research Institute, Switzerland
[3] École Polytechnique Fédérale de Lausanne (EPFL), Switzerland
edgar.romanrangel@unige.ch,
{odobez,gatica}@idiap.ch

Abstract. In this work we address the problem of detecting instances of complex shapes in binary images. We investigated the effects of combining DoG and Harris-Laplace interest points with SIFT and HOOSC descriptors. Also, we propose the use of a retrieval-based detection framework suitable to deal with images that are sparsely annotated, and where the objects of interest are very small in proportion to the total size of the image. Our initial results suggest that corner structures are suitable points to compute local descriptors for binary images, although there is the need for better methods to estimate their appropriate characteristic scale when used on binary images.

Keywords: Shape detection, image retrieval, Maya hieroglyphs.

1 Introduction

The interpretation of ancient Maya inscriptions requires the identification of the basic individual components (glyphs) of the Maya writing system. Currently, this identification process is performed manually by experts, who often need to consult printed catalogs [1], [2]. However, often the size of the individual glyphs is considerably small in proportion to the size of a complete inscription, thus making laborious the manual detection process.

The complexity of the manual detection process increases if we take into consideration that, it is a common feature of the Maya writing system to arrange glyphs at arbitrary position within the inscriptions. Therefore, the implementation of techniques for automatic detection of these complex glyphs requires special attention. Fig. 1 shows and example of a Maya inscription.

One issue related to the automatic detection of Maya hieroglyphs, is that currently, the amount of annotated data that is available remains limited, thus making difficult the implementation of supervised learning methods.

In this paper we present the results of an evaluation of shape descriptors for the automatic detection of Maya glyphs, using weakly annotated binary

⋆ This work was conducted at Idiap as part of the first author's doctoral research.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 145–154, 2013.

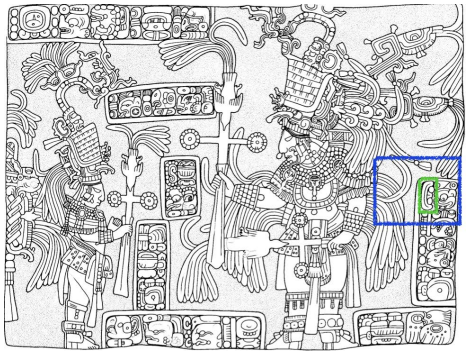


Fig. 1. Maya inscription from the archaeological site of Yaxchilan, Mexico. In a green rectangle, there is an instance of syllable u used as ground-truth for the random block bounded by the blue rectangle. © Image provided by archaeologist Carlos Pallan (University of Bonn), we use this image with his consent.

images. We believe this work will have a positive impact on the daily work of archaeologist diminishing the time required to decipher Maya inscriptions. Namely, the contributions of this paper are:

1. The generation of a synthetic dataset of Maya syllables, which was gathered to overcome the issue of only having few available instances in each visual class. Because of the nature and visual complexity of its instances, this is a unique and highly valuable dataset.
2. The approximation of the sliding-window detection approach by a retrieval-based detection scenario. Our approach overcomes the issue of having only few annotated data, which constrains the use of supervised learning methods.
3. The evaluation of two popular interest point detectors, and their combination with two state-of-the-art image descriptors on the task of shape detection. More precisely, the DoG [3] and Harris-Laplace [4] interest point detectors, and the SIFT [3] and HOOSC [5] descriptors.

Note that the HOOSC descriptor builds on top of Shape Context (SC) [10], and that was proposed to overcome some of its limitations when dealing with shapes that are more complex than the brand logos SC was evaluated on.

The rest of this paper is organized as follows. Section 2 discusses the related work in shape description and image detection. Section 3 introduces the dataset we used in this work. Section 4 explains our experimental protocol. Section 5 discusses the results obtained with our approach. Finally, in section 6 we present our conclusions and a discussion on the open issues.

2 Related Work

The representation of shapes is a research topic with long tradition [6], [7], [8], [9]. In a nutshell, shape descriptors differ according to whether they are applied

to contours or regions, and whether they describe global or local patterns of the shapes. For instance, descriptors based on moments are relatively easy to compute, and they are robust against location, scale, and rotation variations [6]. And Fourier descriptors work well for simple shapes of convex contours [7]. However, both of them perform poorly with affine transformations, and for complex shapes whose instances have many local variations. Also, they need efficient approaches to normalize descriptors derived from different shape signatures [8].

Shape context descriptors [10] incorporate robustness against affine variations, and are able to deal with shapes of high visual complexity [5]. However, the size of the bounding box containing the shape of interest is of high relevance for the normalization, which in principle is unknown on a detection setup. Therefore, they are not suitable for detection purposes.

Several approaches have shown success in the task of detecting objects on gray-scale images [11]. The common framework for image detection implements a sliding-window, in which a classifier is used to evaluate sub-windows and decides whether or not they contain the element of interest. However, such methods require having enough amount of data to train the classifier. Another limitation for using traditional gray-scale oriented approaches [3], is that they rely on local regions of interest whose size is estimated using the information provided by local intensity changes [4], and this information is absent in binary images.

Common approaches to deal with the problem of detecting shapes address these issues by relying on shape information estimated upon gray-scale images, i.e., by extracting contours and local orientations based on the local gradients of intensity images rather than using binary images [12]. For instance, using a networks of local segment as descriptors, and performing detection of shapes belonging to classes that are relatively easy to differentiate in visual terms [13].

In contrast, in this work we address the problem of detection of complex shapes that exist as binary images. These shapes belong to visual classes that exhibit high levels of both inter-class similarity and intra-class variability, thus making the problem more challenging. Also, we implement an ad-hoc approach to address the issue of having limited amount of data to train a classifier.

3 Dataset

We use blocks randomly segmented from very large inscriptions to have a better control over the experimental setup. The reason for this is that the inscriptions are very sparsely annotated relatively to their size and content, such that there is a high probability of detecting non-annotated true-positive instances. With this purpose, we generated three set of images: ground-truth, positive, and negative instances.

More specifically, we followed a five-steps process for data generation: (1) First, we chose 24 visual classes of syllabic Maya hieroglyphs, and for each of them, we manually located 10 different instances on a large collection of inscriptions (thus, 240 instances). We labeled this set as *ground-truth* instances. The reason to choose only 24 visual classes is because they correspond to the hieroglyphs that were most commonly used, thus facilitating their manual location

and segmentation. (2) Then, we generated a random block for each ground-truth instance. This generation of random blocks consisted in segmenting a sub-window containing the ground-truth itself plus a surrounding area, with the restriction that the left and right margins surrounding the ground-truth had a random size between one and four times the width of the respective ground-truth, while the top and bottom margins have random sizes between one and four times the height of it. The decision to use such values is to generate random blocks that contained enough visual information around the ground-truth, such that the challenge of a realistic detection setup is kept. Fig. 2 shows the details of the random block highlighted in Fig. 1. (3) The next step consisted in annotating the random blocks, such that the bounding box of each ground-truth is known, relative to the random block and not to the original large inscription, i.e., once a random block was segmented, we annotated the coordinates x and y where the ground-truth bounding box starts, and its corresponding width (w) and height (h). (4) Later, we generated 20 variants of each ground-truth by randomly shifting the position of its bounding box up to 0.2 times its width and height respectively, and we annotated the location (x, y, w, h) of these variants. This resulted in 200 instances per syllabic class that we labeled as *positive*. (5) Finally, for each segmented random block, we annotated the location (x, y, w, h) of all the existing bounding boxes that are of the same size as its respective ground-truth, but that do not overlap with it. This last part resulted in 6000+ bounding boxes that we labeled as *negative* instances. On average, each block contributed with 26.1 ± 16.0 negative instances.

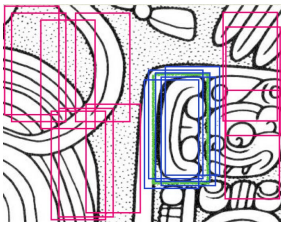


Fig. 2. Random block extracted from the inscription shown in Fig. 1. The ground-truth corresponds to an instance of syllable *u*, and it is inside the green rectangle, the positive instances are marked with blue rectangles, and the red rectangles indicate some of the negative instances.

In summary, the dataset is composed by 24 syllabic classes, in total containing 240 ground-truth instances (10 for each syllabic class), 4800 positive instances (200 for each syllabic class), and 6000+ negative instances that do not belong to any of the positive classes. By annotating the images in this way, we turned the traditional detection approach based on sliding-windows into a retrieval-based approach. This change avoids the risk of detecting non-annotated true-positive instances, and resulted in fast detection experiments, although at the price of non-exhaustive scanning of the large inscriptions.

4 Experimental Protocol

This section explains the experimental protocol followed to evaluate the detection performance achieved by using DoG and Harris-Laplace interest point, and combining them with the SIFT and HOOSC descriptors. Table 1 summarizes the combinations we evaluated.

Table 1. Tested combinations of interest points and local descriptors for detection of Maya syllables

Name	Interest points	Descriptor	Input format
DoG-SIFT	DoG	SIFT	shapes with thick contours
DoG-SIFT-thin	DoG	SIFT	shapes with thinned contours
DoG-HOOSC	DoG	HOOSC	shapes with thinned contours
HarrLapl-HOOSC	Harris-Laplace	HOOSC	shapes with thinned contours

For the DoG and SIFT implementations we used the OpenCV libraries, and we implemented the Harris-Laplace and HOOSC methods in Matlab. Since the HOOSC descriptor was developed to deal with medial axes of shapes, and with the purpose of comparing the two descriptors, we also computed DoG points and SIFT descriptors for the thinned versions of the shapes, as shown in Table 1. Namely, we performed our experiments under the following six-steps protocol:

1. **Interest point detection:** First, we detected points of interest (DoG or Harris-Laplace), along with their characteristic scales and local orientations on the random blocks. For the computations of interest points we considered each random block as a whole (i.e., the points of interest were not computed individually per each bounding box), thus avoiding potential boundary effects as in a common detection setup.
2. **Description:** Second, we computed the local descriptors (SIFT or HOOSC) using the point's characteristic scales and local orientations. This computations were also performed over each complete random block.
3. **Estimating visual vocabularies:** After computing the sets of descriptors for all the random blocks, we randomly drew 1000 descriptors (SIFT or HOOSC) from each visual class and clustered them into 1000 "words". To do so we used the k-means clustering algorithm.
4. **Indexing:** Then, we constructed bag-of-visual-words (*bov*) representations individually for each bounding box. The *bov* were constructed taking into account only those points whose characteristic scale was relevant within the current bounding box, thus excluding points that might contain more information about the exterior than about the interior of the bounding box. Therefore, we excluded: (1) points whose scale is much larger than the bounding box, and (2) points near the the edge of the bounding box and whose scale only intersects a small proportion of it. More specifically, we excluded

all those points whose ratio of intersection $r = A/(2s)^2$ was below 0.5, where s is the characteristic scale of the point, and where A is the intersection area between the characteristic scale and the current bounding box.

5. **Detection:** After computing the *bov* representation of each bounding box, we computed the euclidean distance from each ground-truth's *bov* against the *bov*s of all the positive and negative bounding boxes extracted from the random blocks of the same class as the current ground-truth, i.e., we performed detection on weakly annotated random blocks, looking for instances for which we know they are present inside a given random block. Note that we excluded the random block that contains the current ground-truth, as itself and all its positive variants are easily detected. In practice, each ground-truth is expected to have smaller distances to 189 bounding boxes (the other 9 ground-truth instances plus their 180 positive instances) than to the negative instances (on average, 234.8 negative bounding boxes per class). Thus our detection method is not a classical exhaustive sliding-window but an approximation based on a retrieval approach.
6. **Evaluation:** Finally, we ranked all the bounding boxes based on the computed distances, and evaluated the detection performance in terms of,
   - ROC curves. Comparing the mean average detection-rate (mA-DR) versus the mean average false-positive-rate-per-window (mA-FPPW) at various threshold values.
   - Curves showing the average-precision achieved at different top-N positions of the ranked subwindows.
   - The mean Average Precision (mAP).

Note that the HOOSC descriptor, as described in [5], has five main characteristics. Namely, (1) it uses thinned versions of the shapes, (2) estimates local descriptors only for certain locations (termed pivots) with respect to whole set of points in the thinned shapes, (3) computes an histogram of local orientations in each of the regions of a polar grid around each pivot, (4) in turn, the spatial scope of the polar grid is defined as a function of the average pair-wise distance between all the point in the thinned shape, and (5) the explicit relative position of the pivot to be described may be used as a part of its own descriptor.

These characteristics of the HOOSC descriptor work well in tasks such as classification and retrieval of shapes that have been previously segmented and where the instances are not rotated or reflected. However, such assumptions are not true in the case of a detection setup. For our experiments, it was not possible to compute the spatial scope of the polar grid as a function of the pair-wise distances of the contour points, as the correct size of the bounding box is unknown a priori, and evaluating all possible sizes would result impractical. Therefore, we made use of the characteristic scale of the interest point (DoG or Harris-Laplace) at which the descriptor is computed. More precisely, the polar grid we implemented has two local rings with boundaries at 0.5 and 1.0 times the characteristic scale of the interest point. Also note that we did not use the explicit relative position of the pivots in their description, as the size of the

candidate bounding boxes is assumed to be unknown, and also because some elements might be rotated within the inscriptions.

5 Results

The ROC curves in Fig. 3 show that the use of DoG points with thinned contours gives detection rates close to chance, both with SIFT and HOOSC descriptors (green and red curves, respectively). This observation is not especially surprising as binary images lack of intensity information which is the main clue to localize DoG interest points and to estimate their characteristic scale. The motivation to use DoG points in thinned shapes was based on the high frequency of blob structures present in the Maya syllables. However, some times of the DoG interest points correspond to large blob structures that encompass visual information beyond the locality of the glyph of interest, which in practice, were excluded as explained in section 4. This in turn, resulted in poor shape representations.

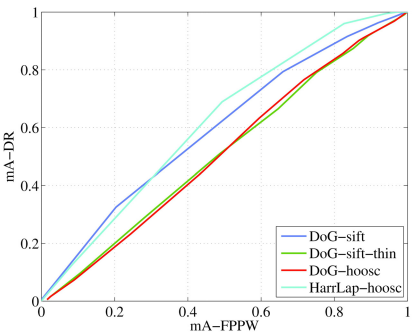
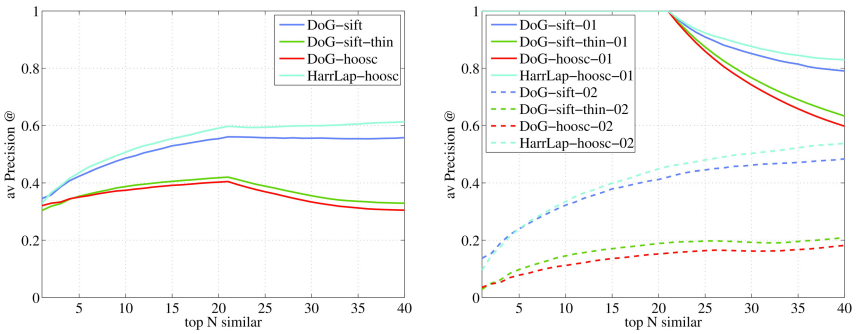


Fig. 3. ROC curves showing the detection performance of different combinations of interest point detectors and shape descriptors

Note that the detection rate is relatively increased with the estimation of DoG points on the original shapes that have thick contours (blue curve in Fig. 3), this is mainly explained by the used of the Gaussian convolutions that smooth the thick contours and approximate intensity values on the resulting image. Moreover, the use of Harris-Laplace interest points resulted in a slightly increased detection rate when used on thinned shapes (see cyan curve Fig. 3).

In terms of retrieval precision, the relative difference among the four methods remains proportional to their ROC curves, as shown in Fig. 4a. The slight peak in the retrieval precision at position 21 results because some classes have instances very similar to one another, such that for a given query (ground-truth instance), the 21 bounding boxes (ground-truth + positive instances) of (at least) one

relevant random block are well ranked at the top of the retrieved vector. To better illustrate this, we recomputed the average precision regrouping the ranked vectors into two sub-groups: one with the queries whose precision curves remain equal to 1 at the 21-st position, and the other with the remaining queries. These results are shown in Fig. 4b, where the solid curves (named XXX-01) show the average precision for the first set, and the dashed curves (named XXX-02) show the average precision of the second set. This said, some visual classes are very easy to retrieve, whereas some others are quite hard.



(a) Average precision at top N **(b)** Split average precision at top N

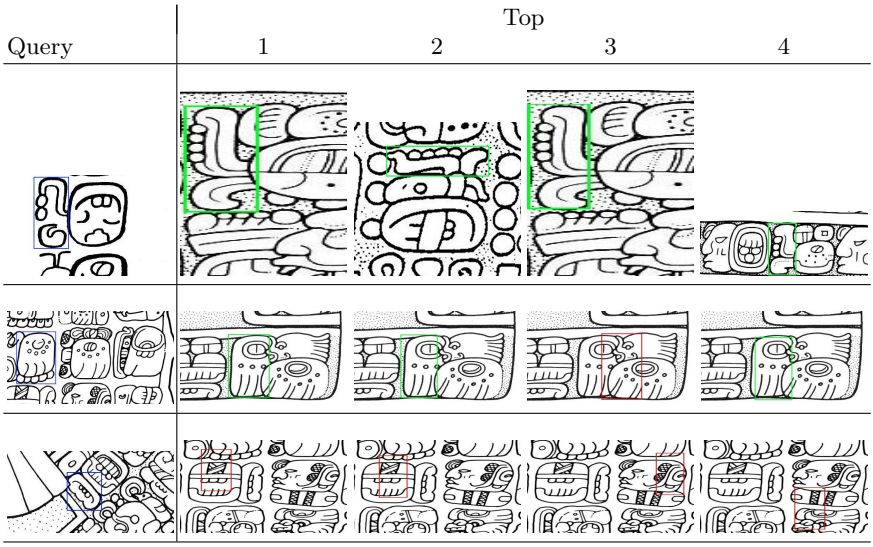
Fig. 4. Average retrieval precision achieved by the different combinations of interest point detectors and shape descriptors. (a) ROC curves, and (b) mean Average Precision curves at top N, plotted for different combinations of interest points and images descriptors evaluated in detection experiments.

The cyan solid curve in Fig. 4b corresponds to HOOSC descriptors computed at Harris-Laplace interest points. Note that this curve remains with good precision values at the 40-th position of the top N vector. Thus indicating that this combination of interest points and descriptor works well in general terms. To summarize the retrieval performance of the tested combinations, we present their mean Average Precision (mAP) in Table 2. Note that the use of corners as interest points achieves better performance than blob structures. Finally, Table 3 shows visual examples of the detection obtained using Harris-Laplace points with HOOSC descriptors.

Table 2. Mean average precision (mAP) for the combinations of interest points and local descriptors tested for detection of Maya syllables

Method	DoG-SIFT	DoG-SIFT-thin	DoG-HOOSC	HarrLapl-HOOSC
mAP	0.614	0.449	0.440	**0.646**

Table 3. Visual examples of detection with Harris-Laplace interest points and HOOSC descriptors. The first random block in each row contains a query inside a blue rectangle (ground-truth). The next four random block correspond to the four most similar bounding boxes according to the method, where green rectangles indicate correct detection, and red rectangles indicate erroneous detection.



6 Conclusions

In this work we explored an initial approach for detection of complex binary images (syllabic Maya hieroglyphs), evaluating the performance of DoG and Harris-Laplace interest points combined with SIFT and HOOSC descriptors.

We presented a controlled retrieval-based framework for detection that can be used as an alternative resource when the data is sparsely annotated, thus avoiding the risk of detecting non-annotated true-positive instances. This setup also avoids the exhaustive scanning of the traditional sliding-window approach.

Our results show that regardless of the local image descriptor, the use of DoG points with thinned contours gives detection rates close to chance as a consequence of the lack of intensity information in binary images. A slightly better performance is achieved by using thicker contours since the Gaussian smoothing approximates some sort of intensity information. Moreover, the use of corner detectors seems suitable for local description of complex binary images, as shown by the detection rates obtained by the Harris-Laplace interest points. In terms of retrieval performance, the HOOSC descriptor achieves competitive results, specially when it is combined with Harris-Laplace interest points.

It is important to remark, that this initial stage suggests the need for interest point detectors specially tailored for binary images, such that regions of interest are located within the shape along with their characteristic scales, and therefore, shape descriptors that have proven successful with segmented shapes can be used also for detection.

Acknowledgments. This research was supported by the Swiss NSF CODICES project (grant 200021-116702). We thank the Swiss NSF through the NCCR IM2 for providing travel funds, and Prof. Stéphane Marchand-Maillet for comments.

References

1. Thompson, J.E.S.: A Catalog of Maya Hieroglyphs. University of Oklahoma Press, Norman (1962)
2. Macri, M., Looper, M.: The New Catalog of Maya Hieroglyphs. The Classic Period Inscriptions, vol. 1. University of Oklahoma Press, Norman (2003)
3. Lowe, D.G.: Distinctive Image Features from Scale-Invariant Keypoints. International Journal of Computer Vision 60(2), 91–110 (2004)
4. Mikolajczyk, K., Schmid, C.: Scale and Affine Interest Point Detectors. International Journal of Computer Vision 60(1), 63–86 (2004)
5. Roman-Rangel, E., Pallan, C., Odobez, J.-M., Gatica-Perez, D.: Analyzing Ancient Maya Glyph Collections with Contextual Shape Descriptors. International Journal in Computer Vision, Special Issue in Cultural Heritage and Art Preservation 94(1), 101–117 (2011)
6. Hu, M.-K.: Visual Pattern Recognition by Moment Invariants. IEEE Transactions on Information Theory 8(2), 179–187 (1962)
7. Zahn, C.T., Roskies, R.Z.: Fourier Descriptors for Plane Close Curves. IEEE Transactions on Computers 21(3), 269–281 (1972)
8. Zhang, D., Lu, G.: Review of Shape Representation and Description Techniques. Pattern Recognition 37(1), 1–19 (2004)
9. Yang, M., Kpalma, K., Ronsin, J.: A Survey of Shape Feature Extraction Techniques. In: Pattern Recognition, pp. 43–90 (2008)
10. Belongie, S., Malik, J., Puzicha, J.: Shape Matching and Object Recognition Using Shape Contexts. IEEE Transactions on Pattern Analysis and Machine Intelligence 24(4), 509–522 (2002)
11. Viola, P.A., Jones, M.J.: Rapid Object Detection using a Boosted Cascade of Simple Features. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (June 2001)
12. Payet, N., Todorovic, S.: From Contours to 3D Object Detection and Pose Estimation. In: IEEE International Conference on Computer Vision (November 2011)
13. Ferrari, V., Fevrier, L., Jurie, F., Schmid, C.: Groups of Adjacent Contours for Object Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(1), 36–51 (2008)

Person Re-identification Based on Enriched Symmetry Salient Features and Graph Matching

Sara Iodice and Alfredo Petrosino

Department of Applied Science University of Naples Parthenope, Italy

Abstract. We propose a person re-identification non-learning based approach that uses symmetry principles, as well as structural relations among salient features. The idea comes from the consideration that local symmetries, at different scales, also enforced by texture features are potentially more invariant to large appearance changes than lower-level features such as SIFT, ASIFT. Finally, we formulate the re-identification problem as a graph matching problem, where each person is represented by a graph aimed not only at rejecting erroneous matches but also at selecting additional useful ones.

Experimental results on public dataset i-LIDS provide good performance compared to state-of-the-art results.

Keywords: Person Re-identification, Graph Matching, Symmetry.

1 Introduction

Symmetry detection is highly relevant in pattern recognition. Indeed, the description of a figure may be different when it is embedded in a context with horizontal or vertical symmetry [13]. Besides, in tasks requiring the completion of partially occluded visual stimuli, subjects tend to produce systematically symmetrical figures [11]. The concept of symmetry is not univocal: various kinds of properties of an image are defined as symmetry [24] [22]. As instance, a figure has *rotational symmetry* when it can be rotated less than 360° around its central point, or axis, and still matches the original figure.

This cue is peculiar in person re-identification where the problem consists in recognizing people in different poses from images coming from distinct cameras. This is an important task in the video surveillance, where large and structured environments must be supervised (such as airport, metro, station or shopping centers) and it becomes more critical when the cardinality of gallery set increases.

Like [5] we use symmetry principles, but not in a global exception, that is to say symmetry detected at high value of scale. In person re-identification context, global symmetry is not a good descriptor for pedestrian images, where there is often high variation of pose. On the contrary, we use local symmetry as descriptor like [2], based on the consideration that, at different scales, it is potentially more invariant to large appearance changes than lower-level features such as SIFT. Indeed, the symmetry feature combined with a texton-based feature is high discriminative [8].

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 155–164, 2013.

According to the idea that an image can be described at the higher level in terms of a nested hierarchy of local symmetries, in this paper we present a novel graph matching approach to the problem aimed at evolving an initial set of correspondences computed with the local features, as a kind of compromise between the constraints imposed by both the local features and the structural relations. The vertices are the image key points detected by the SIFT (specifically we adopt ASIFT [21]), enriched by features based on color, texture, and local symmetry. The cost of an edge joining two vertices represents a measure of their dissimilarity. Therefore, the problem of Person Re-Identification is formulated as a graph matching problem. Our approach is an appearance-based method which differs from the state of art: *i*) unlike [3] [15] we give great weight to local features; *ii*) we do not adopt spatio-temporal information such as [12] [10]; *iii*) our method is non-learning based one, unlike [26] [1] [16].

The paper is organized as follows: the Sec. 2 deals with local features; in the Sec. 3 we provide details about how graphs are obtained from images and how they are compared. In Sec. 4 we show testing results on public dataset ILIDS and finally in Section 5, some conclusions are drawn.

2 Feature Extraction

First of all, we start with the phase of enhancement applying a Gaussian filter on the image. Then, we transform each image I in the HSV space. In order to make the feature based on color invariant to lighting changes, we equalize the histogram of the V component.

Later, for each pedestrian we detect image asymmetry axes. We obtain the axes of asymmetry combining opportunely two operators: chromatic bilateral operator C and spatial covering operator S. Given two regions, the former C calculates the color distance, the latter S calculates the difference of FG (foreground). The asymmetry axes separate regions corresponding to human body's parts (head, trunk, and legs). The first axis, which is located at height iTL, separates regions with different chromatic contents but similar areas. These regions correspond to t-shirt/pants or suit/legs, for example. The second axis is calculated taking into account only the areas. This axis separates regions corresponding to head/body and iis located at height iHT.

This step assumes that the background has been subtracted from each pedestrian image. We use for each image I a mask Z generated through the STEL model [7].

We have chosen to detect features separately because each region is characterized by contrasting aspects. Specifically, features based on color, symmetry, and texture are extracted. They can discriminate the trunk part from the legs part, as shown in Fig. 2, where the first three PCA components of feature vectors derived from the same region are plotted: the head components are red; the trunk components are green; the legs components are blue. It clearly appears

(a) (b) (c) (d)

Fig. 1. Fig(a) Image 5003 in ILIDS dataset, Fig(b) Equalized Image, Fig(c) Background Subtraction, Fig(d) Asymmetry Axes at height iHT and iTL

that blue and green components are concentrated in clusters. To resume, this experiment shows that the features are dicriminative; they can distinguish the trunk region from the legs region.

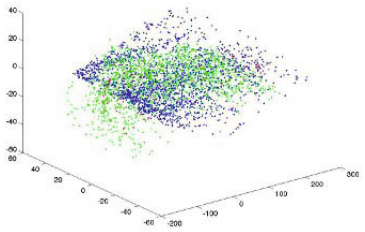


Fig. 2. First three PCs from head (in red), trunk (in green) and legs (in blue)

2.1 Local and Multiscale Symmetry

The property of being symmetrical finds correspondence in size, shape, and relative position of parts on opposite sides of a dividing line or median plane or about a center or axis. In particular, we deal with bilateral symmetry and the symmetry detection algorithm that has been considered in the present paper, named Kondra&Petrosino algorithm, was one of the top winners of the "Symmetry Detection from Real World Images" competition at IEEE CVPR2011[17]. We specifically take advantage of a measure obtained by using correlation with the flipped image around a particular axis. Indeed, in any direction, the optimal symmetry axis corresponds to the maximal correlation of a pattern with its symmetric version [4].

Instead of taking every point in the image, we downsample original images to increase speed by filtering with circular steerable filters [19]. Also, reflecting the patch around both x-axis and y-axis will save half the rotations of the patch. For color images, the patch is reflected with respect to the three bands before doing the correlation.

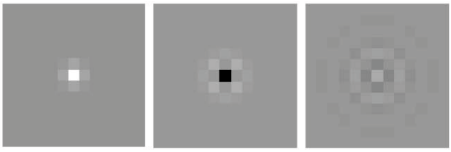


Fig. 3. Filters

2.2 Texture

Texture features are based on the technique proposed in [9]. Specifically, a two-dimensional patch is "seen" for each interest point. The method uses 3 spatial filters of dimension 13×13 as shown in Figure 7. The first is a Gaussian filter with $\sigma = 0.5$, the second is a LoG filter with $\sigma = 0.7$. Finally, the third filter is of the form

$$F(r, \sigma, \tau) = \cos\left(\frac{\pi \tau r}{\sigma}\right) e^{\frac{r^2}{2\sigma^2}} \tag{1}$$

In this case σ and τ are equal to 2. Each two-dimensional patch is convoluted with these filters, later thresholded so as to keep only the positive responses. The obtained binary image is denominated the "on" response. Thus, three binary images were obtained, and then one for each filter. The "on" responses are further convoluted with average filters, respectively of size 3×3 and 6×6 so as to obtain a multi-scale representation. At the next step, images are converted to grayscale and the intensity is normalized to have zero mean and variance equal to one. All filters have been normalized using a $L1$ normalization, so that the responses of each filter are in the same range, i.e. each filter F_i is divided by the $\|F_i\|$.

3 From Person Re-identification to Graph Matching

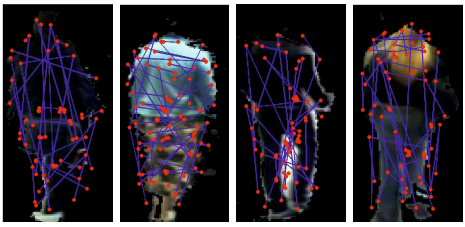
Each salient location detected by ASIFT [21] is enriched by the features described above. Each pedestrian image is rapresented by a graph, in order to consider structural relations. Furthermore, a graph can take into account the dissimilarity between keypoints. So the problem of person re-identification becomes a problem of graph matching.

3.1 Graph Construction

Usually a Graph G is a 3-tuple $G = (V, E, Z)$ where V is a set of vertices, $E \subseteq V \times V$ is a set of edges, where $e \in E$, $e = (v_i, v_j)$ is an edge joining nodes v_i,v_j $\in V$, and Z is a set of vectors, where $z_i \in Z$ is the vector of attributes associated to node $v_i \in V$.

We build the graph for each pedestrian image as follows: vertices are the image key points detected by the ASIFT algorithm. We associate to each vertex a feature vector that contains features based on color (R, G, B), texture, local symmetry and ASIFT descriptor $P_i = (X_i, R_i, U_i)$, where $X_i = (x_i, y_i)$ is its $2D$ location, $R_i = (r_i, \alpha_i)$ its scale and orientation (in radians from $-\pi$ to π) and a vector of length 128, i.e. $U_i = (U_{i,1}, .., U_{i,128})$, including information about the local texture on image.

The edge cost represents a measure of the joined vertex dissimilarity. We build an edge from vertex v_i to vertex v_j if $dist(Z_i, Z_j) = \min(dist(Z_i, Z_k))$, $k = 1, \ldots, m$ (m is the number of vertices) where $dist$ is the Euclidean distance.



(a) (b) (c) (d)

Fig. 4. Graphs of some ILIDS images: Fig(a) 3001; Fig(b) 5003; Fig(c) 13001 and Fig(d) 28005

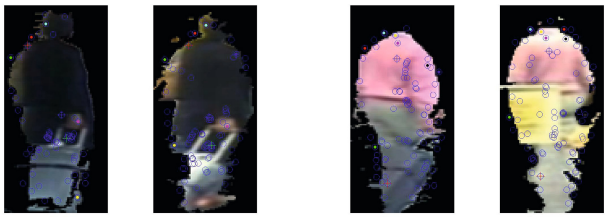
3.2 Graph Matching

One main point is how to measure the contribution of matching one node to another with regards to the structural relations.

We adopt a revised version of the algorithm reported in [18]. The distance between the vertices is obtained through the combination of two measures: the measure of similarity and the measure of consistency. The former is the Euclidean distance between the feature vectors; the latter takes into account the links between the vertices. The algorithm calculates the matching matrix. The matching matrix S is a binary matrix defining an injective mapping between two graphs $G_D = (V_D, E_D, Z_D,)$ and $G_M = (V_M, E_M, Z_M,)$. An element $S_{ij} \in S$ is set to 1 if node $v_i \in V_D$ is matched to node $v_j \in V_M$ and 0 otherwise. A measure of structural consistency for the match $v_i \rightarrow v_j$ is given by:

$$Q_{ij} = \exp\left[\mu \sum_{k \in V_d} \sum_{l \in V_m} D_{ik} M_{jl} S_{kl}\right]$$

where μ is a control parameter to avoid local minima. A cleaning heuristic is needed to obtain a binary matching matrix SS that selects the matches corresponding to the highest coefficients according to a parameter ρ. We set the value of $\rho = 0.8$ (to control the acceptance rate) and the valure of control parameter $\mu = 0.15$. In fig. 4 we show the correspondences detected by the algorithm.



(a) (b)

Fig. 5. Results achieved by the graph matching algorithm: Fig. (a) 0001001 vs 0001003; Fig. (b) 0002002 vs 0002004

4 Validation and Testing

We evaluate performance of our method on the i-LIDS MCTS dataset [20] depicting a real scenario observed by a multi-camera CCTV network monitoring the arrival hall of an airport. i-LIDS MCTS contains 479 images of variable size, for a total of 119 pedestrians. Each pedestrian is characterized at least by 2 images representing the same person wearing the same clothes. Such scenario is affected by several events such as illumination changes, occlusions, shape deformation and image blurring that make challenging the reidentification process.

4.1 Experimental Setup

We focus our attention on the challenging case MvsS, comparing our method with methods at state-of-the.art. [5] proposes SDALF, an appearence method that weights appearence informations extracted from different body parts, in agreement with their distance from symmetry axis calculated over the entire pedestrian image.

Zheng[25] tackles the MvsS case analyzing different types of visual features: Concatenated Histogram, Concatenated Histogram SIFT, and CRRRO-BRO.

In this setup, images are scaled to $128x64$ pixels and the probe set P is built by selecting one image for each pedestrian randomly, while the gallery set G is

composed by the remaining images. Each image $p_i \in P$ is compared with each image $g_i \in G$ in order to evaluate the matching or score distance:

$$\begin{aligned} d(p_i, g_j) = &\alpha \cdot d_{WH}(WH(p_i), WH(s_j)) + \\ &+ \beta \cdot d_{GRAPH}(GRAPH(p_i), GRAPH(s_j)) \end{aligned} \tag{2}$$

where $WH(\cdot)$ (Weighted Color Histograms) [5] is the concatenated HSV histogram extracted by splitting the image I into regions (head, trunk, legs) through horizontal asymmetry axes as described in sec. 2. $GRAPH(\cdot)$ is the graph representation of the pedestrian image as described in sec. 3.1 Vertices of the graph are characterized by concatenation of visual features (R, G, B, KIMA, TEXTONS1, TEXTONS2, ASIFT) and spatial information (x, y), resulting in a final feature vector $F \in R^{138}$. Furthermore, KIMA is a measure of symmetry obtained by selecting maxima correlation between image patches $N \times N$ defined by flipping an image region over an axes having M possible orientation and centered in (x,y). In our experiment we set $N = 8$ and $M = 12$. TEXTONS1 and TEXTONS2 are textural filters response evaluated in (x, y) as described in sec. 2.2. d_{GRAPH} is the graph similarity distance described in sec. 3.1, α and β are weighting values used to give different emphasis to d_{WH} and d_{GRAPH}.

4.2 Experimental Results

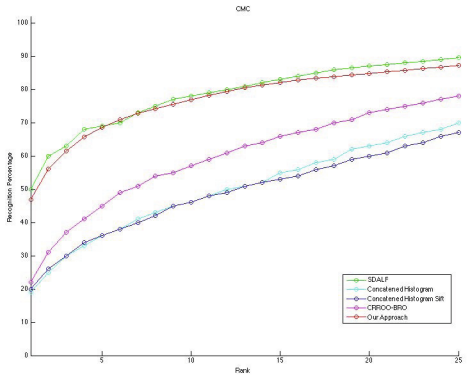
Once all the matches have been evaluated, for each $p_i \in P$, we generate the ranking list composed by elements of G in ascending order, whose first element rapresents the more similar pedestrian to p_i. In our experiment we obtain the best performances with these weighting values: $\alpha = 0.4$, $\beta = 0.6$. In fig. 6, CMC and SRR curves are reported. CMC [6] represents the re-identification rate of the system in function of rank position. Such measure is the analogous to the Roc Curve for detection problem [23]. The SRR or SDR (Synthetic Disambiguation or Reacquisition Rate) converts a performance metric of size N to one of size M, showing the probability that any of the M best matches is correct [6]. SRR and CMC are related by:

$$SDR(M) = SRR(M) = CMC(N/M) \tag{3}$$

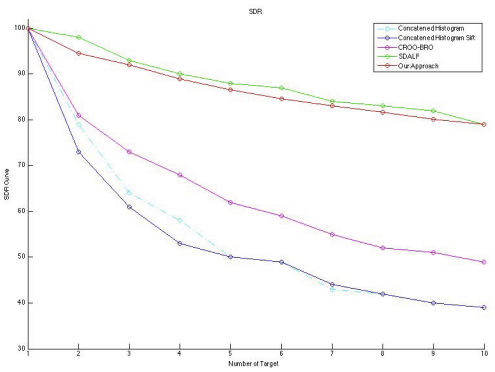
We repeat the experiment MvsS described above 100 times in order to build the CMC curve. As depicted in fig. 6, we achieve a percentage of 48% for rank 1 re-identification, producing results quite similiar to state-of-the-art method SDALF and outperforming Concatenated Histogram SIFT, Concatenated Histogram and CRRRO-BRO.

Such results are very promising since we are enriching ASIFT feature with a descriptor of simmetry that should be very discriminative also in condition of strong occlusions, where the SDALF method clearly should fail since it is based on global simmetry analysis. Unfortunally, the dataset we have utilized to evaluate re-identification methods, is not designed specifically on strong occlusions. State-of-the-art results are relative to the generic reidentification problem, where

the number of occlusion is relatively low compared with the number of images. Our method works correctly on a strong occlusions, such as shown in figure 6 where it obtains rank 1 re-identification .



(a)



(b)

Fig. 6. Comparison in term of CMC (a) and SDR (b) on iLIDS MCTS dataset

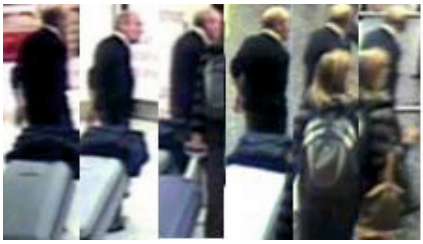


Fig. 7. Example of occlusion

5 Conclusions

We report a new feature detector and descriptor based on ASIFT enriched by local symmetries detected densely across image and scale space, collected together in a graph representation. The basic features are designed for finding correspondence between difficult image pairs rich in symmetries, whilst the graph representation should be able to catch structural relations.

A graph matching method aimed at solving the point-set correspondence problem, takes into account relative structural and geometrical measurements.

The descriptor and results on the i-LIDS MCTS dataset demonstrates that the presence ASIFT feature enriched by local symmetries and structure of an individual is very informative for re-identification. The idea appears completely new, promising and opening a new perspective to the investigation of the occlusion problem, that should be furtherly analyzed. There is room for further improvements in accuracy, mainly for what concerns the tuning of parameter values and a sensitivity analysis.

References

1. Bauml, M., Bernardin, K., Fischer, M., Ekenel, H.K.: Multi-pose Face Recognition for Person Retrieval in Camera Networks. In: IEEE Int. Conf. on Advanced Video and Signal Based Surveillance (AVSS), Boston, MA, USA (2010)
2. Cabrini Hauagge, D., Snavely, N.: Image Matching using Local Symmetry Features. In: Proc. of IEEE CVPR 2012 (2012)
3. Cai, Y., Pietikäinen, M.: Person Re-identification Based on Global Context. In: Koch, R., Huang, F. (eds.) ACCV 2010 Workshops, Part I. LNCS, vol. 6468, pp. 205–215. Springer, Heidelberg (2011)
4. Di Gesú, V., Zavidovique, B.: The S-kernel: A Measure of Symmetry of Objects. Pattern Recognition 40(3), 839–852 (2007)
5. Farenzena, M., Bazzani, L., Perina, A., Murino, V., Cristani, M.: Person Re-identification by Symmetry-Driven Accumulation of Local Features. In: Proc. of IEEE CVPR 2010 (2010)
6. Gray, D., Brennan, S., Tao, H.: Evaluating Appearance Models for Recognition, Reacquisition, and Tracking. In: Gray, D., Brennan, S., Tao, H. (eds.) IEEE International Workshop on Performance Evaluation of Tracking and Surveillance, PETS (2007)
7. Jojic, N., Perina, A., Cristani, M., Murino, V., Frey, B.: Stel Component Analysis: Modelling spatial correlations in image class structure. In: Proc. of IEEE CVPR 2009 (2009)
8. Kondra, S., Petrosino, A.: Self–Similarity and Points of Interest in Textured Images. In: Kundu, M.K., Mitra, S., Mazumdar, D., Pal, S.K. (eds.) PerMIn 2012. LNCS, vol. 7143, pp. 306–313. Springer, Heidelberg (2012)
9. Kondra, S., Torre, V.: Texture Classification Using Three Circular Filters. In: 6th Indian Conference on Computer Vision, Graphics and Image Processing (ICVGIP), Bhubaneshwar, India (2008)
10. Kuo, C.-H., Huang, C., Nevatia, R.: Inter-Camera Association of Multi-Target Tracks by On-Line Learned Appearance Affinity Models. In: Daniilidis, K., Maragos, P., Paragios, N. (eds.) ECCV 2010, Part I. LNCS, vol. 6311, pp. 383–396. Springer, Heidelberg (2010)

11. van Lier, R., Wagemans, J.S.: From Images to Objects: Global and Local Completion of Self-Occluded Parts. Journal of Experimental Psychology: Human Perception and Performance 25, 1721–1741 (1999)
12. Mazzon, R., Fahad Tahir, S., Cavallaro, A.: Person Re-identification in Crowd. Pattern Recognition Letters (2012) (in press)
13. Palmer, S.R.: The Role of Symmetry in Shape Perception. Acta Psychologica 59, 67–90 (1985)
14. Palmer, S.E., Hemenway, K.: Orientation and Symmetry: Effects of Multiple, Rotational, and Near Symmetries. J. Exp. Psychol. Hum. Percept. Perform. 4, 691–702 (1978)
15. Prosser, B., Gong, S., Xiang, T.: Multi-Camera Matching using Bi-directional Cumulative Brightness Transfer Functions. In: Proc. of the British Machine Vision Conf., Leeds, UK (2008)
16. Prosser, B., Zheng, W.S., Gong, S., Xiang, T.: Person Re-identification by Support Vector Ranking. In: Proc. of the British Machine Vision Conf., Aberystwyth, UK (2010)
17. Rauschert, I., Brocklehurst, K., Kashyap, S., Liu, J., Liu, Y.: First Symmetry Detection Competition: Summary and Results, CSE Dept Technical Report No. CSE11-012 (2011)
18. Sanroma, G., Alquzar Mancho, R., Serratosa i Casanelles, F.: A New Graph Matching Method for Point-Set Correspondence using the EM Algorithm and Softassig. Computer Vision and Image Understanding 116(2), 292–304 (2012)
19. Simoncelli, E.P., Freeman, W.T., Adelson, E.H., Heeger, D.J.: Shiftable Multi-Scale transforms. IEEE Trans Information Theory 38(2), 587–607 (1992)
20. U. H. Office : i-LIDS multiple camera tracking scenario definition (2008)
21. Yu, G., Morel, J.-M.: ASIFT: An Algorithm for Fully Affine Invariant Comparison, Image Processing On Line (2011), doi: http://dx.doi.org/10.5201/ipol.2011.my-asift
22. Wenderoth, P.: The Salience of Vertical Symmetry. Perception 23, 221–236 (1994)
23. Wilson, C., Hicklin, A.R., Bone, M., Korves, H., Grother, P., Ulery, B., Micheals, R., Zoep, M., Otto, S., Watson, C.: Fingerprint Vendor Technology Evaluation 2003: Summary of Results and Analysis Report. Technical Report NISTIR 7123, NIST (2004)
24. Zabrodsky, H.: Simmetry - A Review, Tech. Rep. 90–16, CS Dept., The Hebrew University of Jerusalem (1990)
25. Zheng, W.-S., Gong, S., Xiang, T.: Associating Groups of People. In: British Machine Vision Conference, BMVC (2009)
26. Zheng, W.S., Gong, S., Xiang, T.: Person Re-identification by probabilistic relative distance comparison. In: Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition. Colorado Springs, CO (2011)

A Video-Based Spatio-temporal Biometric Template Representation of the Spontaneous Pupillary Oscillations: A Pilot Experiment

Fabiola M. Villalobos-Castaldi and Ernesto Suaste-Gómez

Center of Research and Advanced Studies of the National Polytechnic Institute
Av. IPN # 2508 Col. Sn. Pedro Zacatenco, Del. Gustavo A. Madero, Mexico City, D.F., 07360
{mvillalobos,esuaste}@cinvestav.mx

Abstract. In this paper, we introduced a new video-based spatio-temporal identification system and we also presented our initial identity authentication results based on the spontaneous pupillary oscillation features. We demonstrated that this biometric trait has the capability to provide enough discriminative information to authenticate the identity of a subject. We described the methodology to compute a spatio-temporal biometric template recording the pupil area changes from a video sequence acquired at constant light conditions. To our knowledge, no attempts were made in order to distinguish individuals based on the spatio-temporal representations computed from the normal dilation/contraction condition of the pupil. In preliminary experiments, for the privately collected database, we observe that Equal Error occurs at a threshold of 5.812 and the error is roughly 0.4356%.

Keywords: Spatio-temporal biometric template, spontaneous pupillary oscillations, hippus.

1 Introduction

In humans, the iris around the pupil is one of the few autonomic physiological effectors that are immediately accessible to the eye. Eye-based biometric systems have some modalities: iris, retina, sclera veins, etc. [1, 6, 7, 8, 12, 16, 17, 18, 19,21], which sometimes are combined or fused, for example, using iris/retina features. This work introduced a spatio-temporal representation of the pupil dynamic features extracted from a video sequence acquired at constant external light conditions. This new approach employs the spontaneous pupillary oscillations as a liveness indicator by recording the pupil area fluctuations from the video record (Fig. 1). The spatio-temporal biometric template that has been proposed to extract efficiently the uniquely identifiable local dynamic features of the pupil is described in this work.

Since the eye-based biometric systems have unique features in their templates, so far in the existing literature there is not any identification method that completely relies on the pupil itself. This may be perhaps due to the difficulty in extracting and representing the “exact” features of the pupil.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 165–174, 2013.

The goal of this study is to develop a spatio-temporal pupillary representation through the use of the spontaneous pupillary oscillations signals (extracted from a video sequence) that are specific to the pupil of different people, and acquired at constant light conditions (without external stimuli). This natural dynamic behavior of the pupil will be the warranty that this template belongs to a living subject. The spatio-temporal representation of the spontaneous pupillary oscillations satisfies the following common biometric requirements: universality, distinctiveness, permanence and collectability.

Fig. 1. Spontaneous pupillary oscillations as a biometric trait

The major contributions of this work are:
The acquisition of the video sequence of 50 subjects and the creation of our own pupil video sequence set, the design of the experiment and the algorithms to extract and represent the pupil spatio-temporal features, the evaluation and demonstration of the potential discrimination among subjects based on the spatio-temporal representation of the spontaneous pupillary oscillations, the proposal of a mechanism of liveness detection based on the natural spontaneous pupillary oscillations induced at constant light conditions.

2 Background

Biometric traits are the human characteristics (physiological or behavioral) used by biometric-based application like identity access management, access control, surveillance, etc. [5]. The pupil is the circular aperture at the center of the iris which allows light from our environment to pass freely to the light-sensitive sensory cells of the retina. The size of the pupil is determined by the iris muscles, the radial dilator pupillae and the concentric smooth muscle circles of the sphincter pupillae, which work in opposition to dilate (mydriasis) and constrict (miosis) the pupil. [3,4]

The sphincter and dilator muscles are innervated by the Autonomic Nervous System (ANS) parasympathetic and the sympathetic pupillomotor fibers of the third cranial nerve, respectively. The sympathetic and parasympathetic systems work in opposition; a tonic activation in both systems balances each other to produce an average waking pupil-size in ambient illumination of 2-6mm, with an average of 5mm and a range of 1-9mm (Fig. 2).

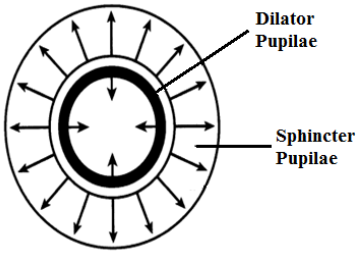


Fig. 2. Muscles of the iris. Two opposite muscle groups within the iris of the human eye determine the aperture of the pupil.

It has been hypothesized, that the muscles controlling the dilation and contraction of the pupil are anatomically proper of the individual. In other words, such a complex system could exhibit high degree of uniqueness. As the iris has a trabecular structure capable of altering its size according to luminous stimuli, it is assumed that this feature can be altered differently in each subject, or that the evaluation of the pupillary fluctuations pattern can contribute to the recognition of a particular subject [18]. It has been demonstrated that the pupil/iris dynamic feature (DF) is discriminating, reason by which may be employed for personal identification. In addition, the possibility to extract this DF from living irises could avoid attempts of fraud in personal identification. [4,16,18].

As it was demonstrated by John Daugman in 2004 [21], the pupil, in most cases, is not a perfect circle; it is better approximated by an ellipse. Because the shape of the pupil is controlled by muscles (trabeculae), the contraction and dilation can change the size of the pupil, however the shape of the pupil remains same. [6,18].

Infrared video pupillography (IVP) combined with a special image processing and analyzing software offers the possibility of noninvasive stable long time record of pupil area and detailed analysis of changes of spontaneous pupillary oscillations. Person's eyes provide several useful biometric features both for the high security and user convenient applications. [11, 15, 17, 19]. Arguably, the most important feature in facial expression and animation are the eyes. [19]. The size of the pupil is determined by the iris muscles, the radial dilator pupillae and the concentric smooth muscle circles of the sphincter pupillae, which work in opposition to dilate (mydriasis) and constrict (miosis) the pupil. [4,17,18, 19, 21]

2.1 Spontaneous Pupillary Oscillations (HIPPUS)

The pupil area of the human eye continuously undergoes to small temporal fluctuations even in dark or under conditions of constant illumination, ranging in magnitude from 0 to 20%. This pupil activity, also called "hippus", is a normal condition, is independent of illumination, convergence, or psychic stimuli (Fig.3). Hippus appeared to be spontaneous and seems most likely to occur when the subject is passive and left to himself. There is no doubt that hippus occurs in the absence of any light stimulus and in the pupils of both eyes, simultaneously and in phase. [2, 9 10, 13, 14].

As we had no preconceived hypothesis about whether and how the spatio-temporal features of pupillary signals are discriminative or not, in this work we present the preliminary verification results based on the spontaneous pupillary oscillations computing a spatio-temporal template recording the pupil area changes from a video sequence acquired at constant light conditions. A prototype image acquisition system was designed and developed to capture the pupillary hippus. In this work, we could detect liveness using as involuntary information mechanism the spontaneous pupillary oscillations. Our preliminary experimental results on a private database containing 50 subjects are presented.

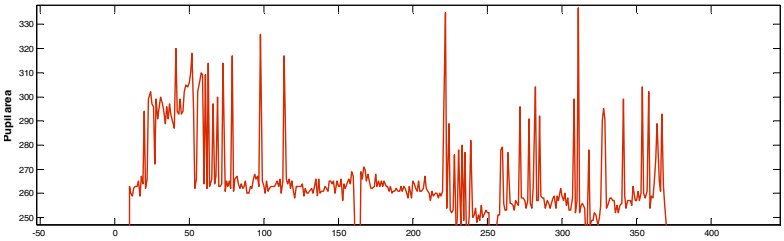


Fig. 3. Typical example of spontaneous Hippus. At constant illumination, the average diameter of the pupil decreases as slow oscillations appear. When oscillations cease the pupil diameter returns to its initial value.

3 The Proposed Video-Based Method

Still-image recognition is routinely covered by existing research studies. However, video-based recognition approach is a relatively new research topic. The pupil video-based recognition method has superior advantages over the static image-based recognition [20]. Firstly, the temporal information of pupil images can be utilized to facilitate the recognition task (the person-specific dynamic pupil characteristics can help the recognition). Secondly, more effective representations can be obtained from the video sequence and used to improve recognition results. Thirdly, the involuntary characteristic of the pupil will be employed as a liveness detector. Finally, video-based recognition allows learning or updating the subject model over time.

Our proposed algorithm can represent the dynamic information of the spontaneous pupillary oscillations by a spatio-temporal biometric template and hence improve the recognition performance of the system. The proposed video-based biometric approach integrates a number of steps, each of which are explained in detail in the following sub-sections.

Pupil video acquisition, extraction of the Region of Interest, pupil region segmentation and representation, template matching.

3.1 Pupil Video Acquisition

A problem associated with the video-based technique is primarily that of getting a good view of the pupil. Head and eye movements can set up the image of the pupil

out of focus or out from the view field of the camera. So, if we want to obtain accurate measurements of a human subject´s pupil, it is necessary that the subject remains absolutely still, thus, we employed a combined head/chin stand where the subjects rested their chin in such a way that the head position remains adjusted (Fig. 4). The study followed the tenets of the declaration of Helsinki.

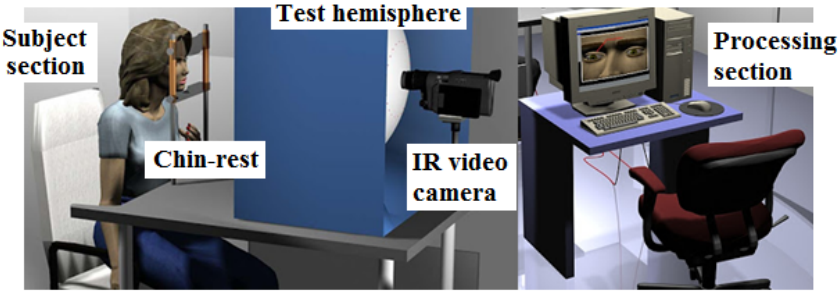


Fig. 4. Experimental setup used for the proposed system [22]

The head/chin rest was mounted in a small table, approximately 20 centimeters in front of the infrared (IR) camera. In this way, the subject was able to feel comfortable and eventual eye movements would not appear. The alignment of the infrared camera was adjusted depending of the central position of the pupil in the screen. The accuracy of the IR video camera was greatly improved by fixing the head and eye position. [13,14]. Subjects were seated on a comfortable chair.

When (infrared) light arrives to the user's eye, several reflections occur on the boundary of the pupil, the so-called *Purkinje images*. This reflex can compromise the image quality of the pupil to be analyzed and segmented, so we positioned the IR video camera on a fixed place and instructed the subjects to look at the IR LED (2 mm in diameter) of the camera such that the reflex falls at the center of the pupil.

To reduce fatigue, and the potential effects of head or eye movements, each stage lasted 10 seconds approximately. The recordings started when the pupil image is located at the center of the screen camera and when the IR LED reflex appears at the center of the pupil. We acquired 20 infrared frames per video sequence per subject. Data were recorded from one eye only, typically the left eye. All the gray images were stored in *.jpeg format.

Another challenge in this approach is related to the detection and definition of the dynamic features of the pupil related to the light conditions, with the best discrimination potential among the subjects. The light luminance level was controlled in a quantitative manner using the visible light LEDs placed on the back of the campimeter, at a number of discrete stimulus locations out of the visual field.

We performed an exhaustive analysis on the spontaneous pupillary oscillations related to the ambient light conditions. After this, the selected luminance of the stimulus was 40 cd/m^2. This was the highest luminance available on the experiment. As we were interested in evaluating the inter-person variations of the spontaneous pupillary fluctuations, we should factor out the possible influence of the light reflex in the pupil

(change in pupil size from the dark to light condition); therefore, complete dark adaptation was not necessary. In constant light condition, the amount of information obtained from subject's individual images is almost the same, so the data will be compatible.

Pupil images were recorded and transferred frame by frame by means of a video capture board providing real-time digitizing along the video sequences. The video-capture board was fitted by USB on a Pentium ® Dual-Core T4200 @ 2GHz and 4 GB of internal memory, and with a MATLAB 7.10 (R2010a) implementation to obtain real time recordings of the movie. The image size is 480 x 640 pixels.

Fig. 5. Samples of IR pupil images of 4 different persons; the bright spot in the middle of the pupil is the reflex of the infrared illumination LED used as the fixation point

We also developed a Graphical User Interface (GUI) which provides a real time visual feedback of the pupil images that is actually acquired. This GUI has the following functionality:

Assist the user in correct positioning of the pupil at the center of the screen and of the light reflex at the center of the pupil before the video recording starts, acquires the video sequence of the user's pupil, displays video sequences that were recorded previously, extracts the spatio-temporal representations from a given pupil video sequence, registers the user in the database along with the extracted spatio-temporal representation, checks whether a given video sequence of the pupil match any of the subject in the database, and updates a particular user's entry in the database by re-computing the spatio-temporal representation.

To preprocess the data, erroneous measurements were previously removed. These problems were caused mostly by involuntary blinking and in few cases from unintentional head and eye movements. Fig. 5 shows some examples pupil images of our database.

3.2 Extraction of the Region of Interest

Commonly, eye images without preprocessing are used to detect pupil region. This takes more time to process the image and in biometric applications the time of analysis plays an important role in the performance of the system. Therefore, before the feature extraction step, it is necessary to extract from the original images a specific portion to work with. This step is known as extraction of the Region of Interest (ROI) and let us to restrict the spatial image processing to the central region. This step reduces a lot the data amount without losing much useful information. This will speed up the following feature extraction and matching processes.

The proposed approach is based on the fact that all eyes' pupils are black round shaped, are far less covered by the eyelids than the iris and are located mostly at the center of the image. For simplify the process, we excluded the external regions, because they contain information that can compromise the image evaluation, such as eyelashes, eyelids, and sclera. The proposed algorithm scans only on the expected pupil's position. For finding the pupil region, the original image is reduced. For example, given a grey level image, we will divide the image by four equidistant horizontally and then by four equidistant vertically fringes, the first and last horizontal and vertical quarts are eliminated and we focus only on the central region. After this, a new image is provided with a clear pupil region that can be easily and quickly segmented and processed.

3.3 Pupil Region Segmentation and Representation

The low contrast between pupil and iris, usually, will harden the pupil boundary detection process and decrease the accuracy of its detection. The textural contrast between sclera and iris is high; conversely the texture contrast between iris and pupil is low. The problem is further aggravated with the presence of light reflection in the pupil [15].

The proposed pupil segmentation scheme relies on the fact that the pupil is round and is typically much darker than its surroundings. This is done using some morphological operations with a suitable thresholding detection procedure. Thresholding pupil detection procedures refer to an image segmentation method manipulating the statistical properties of the image brightness probability distribution (histogram). Hence, the reduced pupil image is preprocessed through the successive application of two morphological operations.

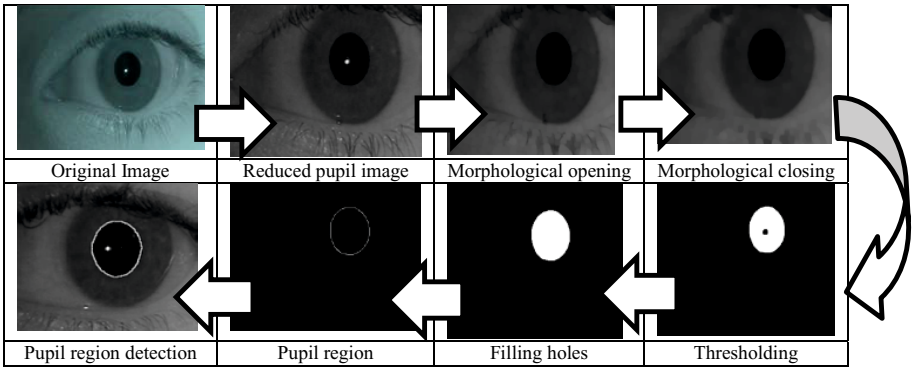


Fig. 6. Example of the image thresholding and pupil region segmentation method

Fig. 6 illustrates the outcome of each step of the pupil region segmentation stage. The measurement of the pupil area frame by frame is saved as the spatio-temporal representation of the pupillary oscillations. By taking into account all the frames in the video sequence, we initially proposed a feature vector with 20 elements.

3.4 Matching

After the extraction of features from pupil images, a matching metric is required to find the similarity between the probe and the gallery template. This metric should give one range of values when comparing templates generated from the same pupil (known as intra-class comparisons) and another range of values when comparing templates created from different pupils (known as inter-class comparisons). These two cases should give distinct and separate values so that a decision can be made with high confidence related to whether two templates are from the same pupil or from different pupils. The metric used for the proposed system is the Euclidean distance.

4 Experimental Results

4.1 Database Establishment

A particular own database was created for this study. We studied 50 healthy young subjects between 24 and 45 years old (20 females and 30 males); most of the subjects were university students and employers. The consent to participate in this study was obtained from each subject after the nature of the procedure was completely explained to them. We collected 1000 grayscale left eye images from 50 persons, 20 from each person. The size of all eye images in the database is 480×640 pixels. Experiments were conducted in a quiet usability laboratory. The video sequences were collected in a resting condition with subjects seated on a comfortable chair.

4.2 Evaluation of the Segmentation Method

We visually evaluated the success of the proposed pupil thresholding and segmentation method for all the 1000 images. The pupil region was correctly detected in all the images getting a success rate of 100%. The proposed method is robust enough because: a) the head/chin rest avoid the involuntary head movements, b) the alignment of the IR camera guaranty that the pupil region is almost always at the center of the video frame and c) the pupil region is not affected by the eyelid and eyelashes noise. Evaluation of the inter-individual differences

The spontaneous pupillary oscillation-based verification system was tested using our own database of 50 users. As was mentioned previously, the measurement of the pupil area frame by frame is saved as the spatio-temporal representation of the pupillary oscillations. 20 images of each user's pupil were used to compute the feature vector fusing it with the user's abbreviation name and birthday and then were stored in the database.

To better evaluate the inter-subject variability in spatio-temporal pupillary oscillations, it seemed useful to plot the individual feature template of 20 of the users using the Chernoff faces mapping the spatio-temporal dynamic features of the pupil to the attributes of the cartoon faces (area and shape of the face, length of the nose, location of the mouth, curve of the smile, width of the mouth, location, separation, angle, shape and width of the eyes, location and width of the pupil, and location,

angle and width of eyebrow). We found great inter-individual differences with regard to the extent as reflected in these cartoon faces.

4.3 Experimental Results on Spontaneous Pupillary Oscillation Verification

To obtain the verification accuracy for the spatio-temporal pupillary template, we computed the false acceptance rate (FAR) and false rejection rate (FRR) at different match thresholds. For this, each pupil video sequence is matched with the feature vector stored in the database associated with the claimed identity. The verification is positive if the distance between the query and the stored feature vector is lower than the threshold value, otherwise is considered as a non-matched vector. The FRR decreases from 0.102 to 0 whereas FAR increases from 0 to 0.9463 as the threshold value increases from 1 to 7. In a pilot experiment, for the privately collected database we noticed that the FRR and FAR intersect at the point [5.813, 0.004356] which represent the EER.

5 Conclusions

This work introduced a new pupillary video-based spatio-temporal verification system. We also presented the initial identity authentication results based on the pupil dynamic features extracted from a video sequence which was acquired at constant light conditions. This paper reports the preliminary results conducted by using an IR-camera, a head/chin rest and a campimeter in laboratory conditions. We demonstrated that this biometric characteristic provide enough discriminative information to verify the identity of a subject. We described the methodology to compute a spatio-temporal feature vector recording the pupil area changes frame by frame from a video sequence acquired at constant light conditions. The pupil region was correctly detected in all the images getting a success rate of 100%. The proposed method is robust enough because: a) the chin rest avoid the involuntary head movements, b) the alignment of the IR camera guaranty that the pupil region is almost always at the center of the video frame and c) the pupil region is not affected by the eyelid and eyelashes noise. We found great inter-individual differences with regard to the extent as reflected in the Chernoff cartoon faces. In preliminary experiments, for the privately collected database, we observe that Equal Error occurs at a threshold of 5.812 and the error is roughly 0.4356%.

References

1. Basit, A., Javed, M.Y., Masood, S.: Non-circular Pupil Localization in Iris Images. In: International Conference on Emerging Technologies, IEEE-ICET 2008, Rawalpindi, Pakistan (2008)
2. Longtin, A.: Nonlinear Oscillations, Noise and Chaos in Neural Delayed Feedback department of Piysics, McGill University, Montréal, Canada (1989)

3. Wilhelm, B., Giedke, H., Dtke, H., Bittner, E., Hofmann, A., Wilhelm, H.: Daytime variations in central nervous system activation measured by a pupillographic sleepiness test. J. Sleep Res. 10, 1–7 (2001)
4. Heaver, B.: Psychophysiological Indices of Recognition Memory, University of Sussex (2011)
5. Toth, B.: Biometric Liveness Detection. Information Security Bulletin 10, 291 (2005)
6. Boyina, S.R., Anu, H., Moneca, K., Mahalakshmi Malini, G., Priyadharshini, R.: Pupil Recognition Using Wavelet And Fuzzy Linear Discriminant Analysis For Authentication. International Journal of Advanced Engineering Technology, IJAET III(II) (2012)
7. Cheung, F.: Iris Recognition, Department of Computer Science and Electrical Engineering, The University of Queensland (1999)
8. Frederick, W., Wheeler, A.G., Perera, A., Abramovich, G., Yu, B., Tu, P.H.: Stand-off Iris Recognition System. In: Biometrics: Theory, Applications and Systems, BTAS 2008 (2008)
9. Calcagnini, G., Censi, F., Lino, S., Cerutti, S.: Spontaneous Fluctuations of Human Pupil Reflect Central Autonomic Rhythms. Methods of Information in Medicine (2000)
10. Bouma, H., Baghuis, L.C.J.: Hippus Of The Pupil: Periods of Slow Oscillations of Unknown Origin. Vision Res., vol. II, pp. 134–151 (1971)
11. Ludtke, H., Wilhelm, B., Adler, M., Schaeffel, F., Wilhelm, H.: Mathematical procedures in data recording and processing of pupillary fatigue waves. Vision Research 38, 2889–2896 (1998)
12. Matey, J.R., Naroditsky, O., Hanna, K., Kolczynski, R., LoIacono, D.J., Mangru, S., Tinker, M., Zappia, T.M., Zhao, W.Y.: Iris on the Move: Acquisition of Images for Iris Recognition in Less Constrained Environments. JPROC 94(11) (2006)
13. Lawrence Stark, M.D.: Stability, Oscillations, and Noise in the Human Pupil Servomechanism. In: Proceedings of the Ire (1959)
14. Warga, M., Lüdtke, H., Wilhelm, H., Wilhelm, B.: How do spontaneous pupillary oscillations in light relate to light intensity? Vision Research 49, 295–300 (2009)
15. Teikari, P.: Automated Pupillometry, Helsinki University of Technology, Centre for Metrology and accreditation, Metrology Research Institute (2007)
16. Bednarik, R., Kinnunen, T., Mihaila, A., Fränti, P.: Eye-Movements as a Biometric. In: Kalviainen, H., Parkkinen, J., Kaarna, A. (eds.) SCIA 2005. LNCS, vol. 3540, pp. 780–789. Springer, Heidelberg (2005)
17. da Costa, R.M., Gonzaga, A.: Dynamic Features for Iris Recognition. IEEE Transactions on Systems, Man, And Cybernetics—Part B: Cybernetics 42(4) (2012)
18. Ojala, S.: Biometric Authentication in Real-Time Large-Scale Applications Dynamic Features for Iris Recognition, Department of Electrical and Communications Engineering (2005)
19. Pamplona, V.F., Oliveira, M.M., Baranoski, G.V.G.: Photorealistic Models for Pupil Light Reflex and Iridal Pattern Deformation. ACM Transactions on Graphics (TOG) SIGGRAPH 2010 28(4), 106:1–106:12 (2009)
20. Liu, X., Chen, T.: Video-Based Face Recognition Using Adaptive Hidden Markov Models. In: Proceedings of the 2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2003, pp. 340–345 (2003)
21. Daugman, J.: How Iris RecognitionWorks (2004)
22. Ernesto Suaste webpage, `http://sugoergaleria.blogspot.mx/` (last acceded January 2013)

Image Segmentation Based on Representative Colors Detection and Region Merging

Giuliana Ramella and Gabriella Sanniti di Baja

Istituto di Cibernetica "E.Caianiello", CNR
Via Campi Flegrei 34, 80078 Pozzuoli, Naples, Italy
{g.ramella,g.sannitidibaja}@cib.na.cnr.it

Abstract. We present a color image segmentation algorithm, RCRM, based on the detection of Representative Colors and on Region Merging. The 3D color histogram of the RGB input image is built. Colors are processed in decreasing frequency order and a grouping process is accomplished to gather in the same cluster all colors that are close enough to the current color. Colormapping is done to originate a preliminary image segmentation. Segmentation regions having small size undergo a merging process. Merging is actually accomplished only for adjacent regions whose colors do not significantly differ. The parameters involved by the algorithm are set automatically by taking into account color distribution in the input image and geometrical features of the regions into which the image is partitioned. The algorithm has been tested on a large number of RGB color images originating satisfactory results.

Keywords*:* RGB color images, 3D histogram, color quantization, image segmentation.

1 Introduction

Segmentation is one of the most important processes when dealing with computer vision applications and has a crucial role since the quality of the obtained result is strongly conditioned by the quality of segmentation. It consists in a suitable partition of an image into non-overlapping, connected and homogeneous regions, each of which represents (part of) an object in the scene. All pixels in the same region are rather similar to each other in terms of a specific property (for example, color or texture). In turn, pixels belonging to adjacent regions significantly differ from each other as regards the same property.

While segmentation is a quite easy task for human beings, it is definitely hard for machines and a large number of different segmentation schemes can be found in the literature. Low level image features, such as color and texture, as well as higher level features, such as geometric models for the objects or Gestalt principle of perceptual organization, have been used. Some surveys covering most of the approaches suggested in the literature can be found for example in [1-3].

When color is the main feature used to achieve segmentation, segmentation methods can be framed as following two possible approaches, respectively involving

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 175–184, 2013.

edge detection and region reconstruction [1]. Edge detection schemes are based on the assumption that local discontinuities of colors exist at the boundaries of adjacent regions representing different objects. In turn, region reconstruction schemes are based on the assumption that a region is a subset of connected pixels sharing similar color properties. Region reconstruction methods can be furthermore divided into two categories, depending on whether color distribution is analyzed in the image plane or in the color space. Spatial analysis is based on region growing that, starting from a set of initial seeds, sequentially groups in the same regions neighboring pixels with similar colors as far as the aggregating conditions are satisfied. In turn, when the analysis of color distribution is accomplished in the color space, where each pixel of the input image is associated with its color point in the three-dimensional color space, regions that are homogeneous in the image plane originate clusters of color points in the color space. Image pixels with similar color properties are grouped in the same cluster. Clusters can be identified by analyzing the color histogram or by means of a cluster analysis procedure, and are successively mapped back to the original image plane to obtain the segmented image. Independently of whether the image plane or the color space are analyzed, the result is likely to be affected by the presence of a number of small size regions scattered through the image. Thus, a merging process is also generally taken into account to reduce over-segmentation [2].

The segmentation algorithm proposed in this paper can be classified as a region reconstruction method, where color distribution is analyzed in the color space. The algorithm, named RCRM, includes two steps, respectively aiming at the detection of Representative Colors in the 3D color space, and at Region Merging in the image plane. The first step originates a preliminary over-segmented partition of the input image. The second step reduces over-segmentation. RCRM is the follow up of a previous method [4]. The main novelties introduced by RCRM are the automatic setting of the parameters and a clever merging process. RCRM has been tested on a large dataset of images taken from publicly available repositories, producing satisfactory results from both the qualitative and the quantitative points of view.

The paper is organized as follows: basic notions are given in Section 2; the algorithm is described in Section 3; experimental results are discussed in Section 4; finally, concluding remarks are given in Section 5.

2 Basic Notions

Let I be a 2D RGB color image. We interpret colors of the pixels of I as 3D vectors, where each vector element has an 8-bit dynamic range. For each color in the 2D image, the coordinates of the corresponding color point along the three Cartesian axes are integer numbers in the range [0, 255]. Thus, colors present in the 2D image correspond to color points within a 3D cube. Of course, the same color is likely to characterize more than one pixel in I. By assigning to each color point in the 3D cube a value counting the number of pixels of I characterized by the color corresponding to that point, the 3D histogram of colors H becomes available.

Though in principle 16 millions of different colors are possible, the number of colors in a digital image is generally remarkably smaller. This is due both to the

obvious limitation given by the size of the images (for example, for images with size 1024×1024, about one million different colors are possible), and to the fact that the same color is likely to appear more than once in the image. Thus, sparse points having value different from zero generally exist in the 3D histogram H. Since the three components and the frequency of a color are expressed by integer numbers, H can be interpreted as a discrete 3D voxel image.

Let C_i be a connected component of voxels of H, where each voxel is characterized by the same frequency f_i. C_i has locally maximal frequency if each voxel of H that does not belong to C_i but has at least a neighbor in C_i has frequency smaller than f_i. For any component with locally maximal frequency, MC_i, we identify the centroid c_i. Obviously, if MC_i consists of a single voxel, the centroid coincides with the voxel itself and the coordinates of c_i are certainly expressed by integer numbers. When MC_i includes more than one voxel, we use the ceiling function to express the coordinates of the centroid c_i by integer numbers, i.e., we map the real numbers expressing the coordinates of c_i to the largest following integer numbers. The color corresponding to the centroid c_i is taken as the color for all voxels of MC_i.

For the sake of completeness, we remark that synthetical images can be built where MC_i is likely to include voxels so far from each other to correspond to very different colors. However, this does not generally happen for natural images. Thus, we can assume that any MC_i found in H includes only voxels corresponding to colors sufficiently similar to each other.

3 Segmentation

The segmentation algorithm RCRM consists of two steps. The first step involves the analysis of the 3D histogram H to identify the representative colors and colormapping to originate a preliminary partition of I. The connected components of colors having locally maximal frequency are identified in H. The color corresponding to the centroid of any such a component is taken as the color of the component itself. Then, a grouping process is done to select a subset of the centroids so as to detect the representative colors. The centroids are processed in decreasing frequency order, starting from the centroid with the largest frequency. The currently examined centroid c_i is taken as a representative color. A grouping process builds the cluster associated to c_i by including in the same cluster all colors present in the 3D histogram that satisfy the following conditions: i) have smaller frequency with respect to c_i, ii) are sufficiently close to c_i, and iii) have not been assigned to any other cluster already built. Since the cluster of c_i may include also colors with locally maximal frequency, the number of representative colors will generally result to be significantly smaller than the initial number of centroids. When all centroids have been considered, the color points corresponding to the colors of the input image result to be grouped into clusters, each of which obtained in correspondence with a centroid taken as representative color. Colormapping is then accomplished to complete the first step of the process and obtain the preliminary partition of the 2D input image. To this aim, the color of each pixel of the image is replaced by the representative color of the

corresponding cluster. The second step is done to reduce over-segmentation. In fact, since the histogram of colors does not take into account spatial information, the image obtained at the end of the first step is likely to be over-segmented. During the second step, a merging process is done, based on the size of the partition regions and on the distance of colors of adjacent regions.

We point out that our representative colors detection technique can be classified as a color clustering technique. With respect to other color clustering techniques, such as K-means [5] and the fuzzy C-means [6], an advantage of our method is that we do not need to fix a priori the number of clusters and the distribution of the centroids. In fact, the clusters are as many as the centroids that survive the grouping process, and are found in correspondence with the connected components of colors with locally maximal frequency in the histogram, i.e., with the colors that appear more often than their neighboring colors in the image. Since we work directly on the 3D histogram, no information contained in the dependence among the three color components is lost.

RCRM involves four parameters, whose values are automatically computed during the process in terms of features characterizing the image, as it will be described in the following. Two of these parameters, π and δ, are employed during Step 1 and are respectively concerned with the minimal value that the frequency of a centroid should have in order that centroid is considered as a possible representative color, and with the maximal distance from a centroid that a color should have to be assigned to the cluster associated to that centroid. The remaining two parameters, α and β, are used during Step 2. The parameter α is concerned with the maximal size of regions that undergo the merging process. The parameter β is concerned with the distance of the color of a small size region from the colors of the adjacent regions.

3.1 Step 1- Selection of Representative Colors and Colormapping

Preliminarily, the centroid c_H of the color distribution for the input image I is detected and the arithmetic mean λ of the distances of all colors in H from c_H is computed. All MC_i, i.e., the connected components of voxels of H with locally maximal frequency, are identified and the arithmetic mean μ of their frequencies is computed. The centroid c_i of any MC_i is identified. If all centroids are taken as possible representative colors, a large number of representative colors is likely to be obtained, which are not all equally meaningful. Moreover, a longer computation time would be necessary to complete Step 1. Thus, we introduce the parameter π and consider as possible representative colors only the centroids with frequency larger than π. Of course, the same value of π would not produce equally good results for images with different color distribution. Accordingly, we relate the value of π to the arithmetic mean μ of the frequencies of the centroids, so as to have a dependence of π on color distribution. To avoid a too drastic reduction of the number of possible representative colors, we should keep the parameter π rather small. Thus, we tentatively set π=0,01μ and use the ceiling function to get an integer approximation for π. The final value for π is computed by taking into account also the number of pixels of I with colors that would not be taken as representative colors due to their small frequency. To this purpose, we compute the sum S of the frequencies of the centroids that, according to the tentative

value of π, would not be regarded as possible representative colors, and increase or decrease by one the tentative value of π depending on the sum S. In practice, we compare S with the total number N of pixels of I. If S is less than a given percentage of N (set to 40% in our work), the tentative value of π is increased, otherwise the value is decreased. The sum S is newly computed and π is again suitably increased/decreased. The final value of π is the largest possible value for which S remains less than the selected percentage of N. We have experimentally verified that, in the average, satisfactory results are achieved by computing the value of π according to the above strategy and by using the two suggested percentages ($1\%\mu$ and $40\%N$). Such a value is in fact large enough to reduce the number of possible representative colors so as to limit the computation time, and at the same time is small enough to guarantee that only for a small number of pixels of I a representative color is not identified. These are the pixels of I whose colors have frequency smaller than π (and, hence, are not taken as representative colors themselves) and largely differ from the colors of detected representative colors (and, hence, are not grouped with any detected representative color).

All centroids with frequency not smaller than π are examined in decreasing frequency order, starting from the centroid with the largest frequency. The rationale for checking centroids in decreasing frequency order is that we regard the relevance of a color as proportionally related to the number of times that the color appears in the image. The current centroid c_i is taken as a representative color. The Euclidean distance is computed between c_i and each color in H that has not yet been assigned to clusters built in correspondence with already selected representative colors and has smaller frequency with respect to c_i. Colors at distance from c_i smaller than the value of the parameter δ are grouped with the centroid c_i. The value of δ should be fixed by taking into account the maximal distance between two colors that a user would not distinguish from each other. The larger is δ the smaller are the numbers of representative colors and of segmentation regions, but the quality of the segmentation becomes rougher. To relate δ to the distribution of colors in I, we set its value to a percentage of the arithmetic mean λ of the distances of all colors in H from the centroid c_H of the color distribution for the input image I. We have experimentally verified that satisfactory results are generally obtained by setting $\delta=0{,}5\lambda$. The above process is likely to group with the current centroid also some centroids with smaller frequency placed at distance from c_i smaller than δ. Thus, not all centroids with frequency larger than π will actually be selected as representative colors.

Once the representative colors have been selected, the preliminary segmentation I' is obtained by colormapping. Any pixel that in the input image I has a given color is set in I' to the corresponding representative color. As already remarked, some colors present in I may not be associated to any representative color and, hence, remain temporarily without any assigned color in I'. The two parameters π and δ play an important role in this respect and their values have to be carefully selected to guarantee at the same time a limited number of representative colors and a small number of pixels without any assigned color in I'. Large values for π favor reduction of the number of centroids, i.e., of possible representative colors, but may originate large portions of pixels without any assigned color. Analogously, large values of δ

definitely favor color grouping, which reduces the number of pixels without any assigned color, but can produce an excessive color fusion that would result in unnatural colors. The criteria to compute the values for π and δ described in this paper have experimentally resulted adequate to have only tiny regions of *I'* consisting of pixels without any assigned color. These pixels are assigned the color of their neighbors. When more than one color can be assigned, we follow a majority rule and assign to the pixel the more frequent color among the colors of its neighbors. While coloring the pixels of *I'* that were not assigned a color during colormapping, also noise cleaning is performed by using the same majority rule: pixels of *I'* grouped into components having very small size (set to at most four pixels in this paper) are assigned the color of their neighbors. We remark that pixels of *I'* that are colored with the same representative color do not necessarily belong to the same connected component. In fact, the histogram of colors counts the number of times that a color appears in the image and the grouping process measures the distance among points in the histogram, but spatial information is not taken into account. As an example of the performance of Step 1 refer to Fig. 1. For the input image 118035 consisting of 321×481 pixels with 23786 different colors (Fig. 1 left), 13134 centroids were detected in the 3D histogram characterized by μ=348. Only 14 representative colors were selected by using the automatically computed parameter values π=3 and δ=46. The resulting image (Fig. 1 middle) is over-segmented, since it includes 348 regions.

Fig. 1. The input image 118035, left, preliminary segmentation, middle, final segmentation, right

3.2 Step 2 - Region Merging

Step 2 is aimed at region merging to reduce over-segmentation. Preliminarily, connected component labeling is accomplished on *I'* to distinguish all regions of the partition, some of which are possibly characterized by the same representative color. At the same time, the area of each connected component is also computed as well as the arithmetic mean *A* of the areas of all partition regions.

During one inspection of *I'*, regions with area smaller than an a priori fixed value α are considered for merging. The parameter α is related to a percentage of *A*, so that different values of α will be used for images characterized by preliminary segmentations that differ from each other for the number of partition regions or for their geometric properties. We have used different values for α to process the images of our dataset. Our experiments suggested to set α=0,2*A* as default value. Let R_k be a region with size smaller than α. Merging of R_k is done only if the color of R_k does not

significantly differ from the colors of the adjacent regions that will absorb the pixels of R_k. Practically, for R_k with identity label l_k and color c_j, merging is accomplished by ascribing to the pixels of R_k the identity label l_n of the adjacent region whose color c_m is the closest to c_j, provided that the distance of c_m from c_j is smaller than a parameter β. Since the distance between colors of adjacent regions in I' is not smaller than δ, a value larger than δ should be used for β. We suggest $\beta=\lambda$ as default value, by taking into account that we set $\delta=0{,}5\lambda$, where λ is the arithmetic mean of the distances of all colors in H from the centroid c_H of the color distribution for the input image I.

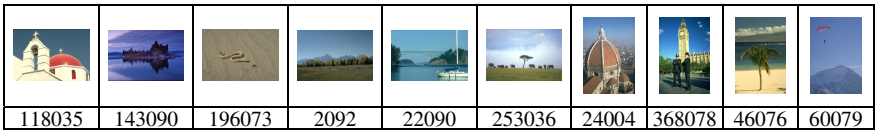
We point out that merging may reduce the number of final representative colors with respect to the number of those selected during Step 1. This happens whenever all pixels that were assigned a given representative color c_j are grouped into regions satisfying the merging conditions. We also point out that the number of regions is generally larger than the number of final representative colors.

In Fig. 1 right, the final segmentation for the input image 118035 is shown.

The arithmetic mean of the area of the 348 regions detected in I' before merging is $A=439$, which implies $\alpha=88$. Out of 312 regions with size smaller than α, 270 regions are actually merged to adjacent regions. Only 78 regions remain after merging and the number of final colors is 11.

4 Experimental Results

RCRM has been tested on about 100 images with different size and color distribution, taken from databases publicly available. A subset of the above collection, including 10 test images taken from Berkeley Segmentation Dataset and Benchmarking (http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/) is shown in Fig. 2.



118035	143090	196073	2092	22090	253036	24004	368078	46076	60079

Fig. 2. Test images

As concerns performance evaluation, we remark that this can be accomplished by following either a supervised approach or an unsupervised approach. In the former case, a ground truth reference is necessary for the evaluation. However, the ground truth is often manually generated and different users may originate a different ground truth, so that evaluation is somehow subjective. In the latter case, a quality score based only on the segmented image is used. Several goodness measures are available in the literature, but unfortunately a unique completely satisfactory measure has not yet been found and the combination of different measures to obtain a single evaluation function is subjective. Thus, we prefer to resort to an unsupervised approach and instead of combining different measures we simply count the number of measures for which the performance of a given segmentation method is better with respect to that of other methods. For each segmented image we compute the number

of representative colors NC, the number of regions NR, and three goodness measures (the Liu and Yang function F, the Borsotti et al. function Q and the entropy E [7]).

We have compared the performance of RCRM with that of the Color Structure Code CSC [8]), the Edge Detection and Image SegmentatiON EDISON [9], and the Recursive Hierarchical Image Segmentation RHSEG [10]. The results are summarized in Table 1. To qualitatively compare the performances of RCRM, CSC, EDISON and RHSEG, refer to Fig. 3, where the results obtained for the test image 46076 are shown.

Table 1. Performance evaluation

Image	Methods	NC	NR	F	Q	E
118035	CSC	347	662	3.681	0.732	10.947
	EDISON	30	86	0.155	0.082	**10.588**
	RHSEG	3	375	0.961	0.238	10.680
	RCRM	11	**78**	**0.100**	**0.036**	10.650
143090	CSC	68	83	0.123	0.135	13.746
	EDISON	40	**71**	0.162	0.170	13.848
	RHSEG	4	1011	0.810	0.703	13.863
	RCRM	13	102	**0.114**	**0.046**	**13.382**
196973	CSC	339	736	1.785	0.574	9.323
	EDISON	56	163	0.093	0.053	9.324
	RHSEG	2	633	0.219	0.654	**9.290**
	RCRM	7	**60**	**0.059**	**0.030**	9.353
2092	CSC	203	246	0.285	0.136	12.230
	EDISON	34	**78**	**0.112**	0.089	12.213
	RHSEG	4	1327	1.297	2.478	**12.183**
	RCRM	15	164	0.255	**0.060**	12.271
22090	CSC	758	1093	3.873	0.631	12.654
	EDISON	144	242	**0.322**	0.108	**12.595**
	RHSEG	5	1432	1.044	1.503	12.532
	RCRM	13	**224**	0.381	**0.097**	12.599
253036	CSC	192	378	1.666	0.390	12.340
	EDISON	27	**31**	0.100	0.152	12.355
	RHSEG	2	46	**0.059**	0.061	**12.306**
	RCRM	6	43	0.073	**0.042**	12.418
24004	CSC	2027	3585	52.316	45.661	14.593
	EDISON	214	**621**	3.159	0.740	14.598
	RHSEG	6	8064	52.731	1455.036	14.631
	RCRM	17	655	**3.116**	**0.532**	**14.567**
368078	CSC	723	1071	5.948	1.562	14.527
	EDISON	233	**403**	**1.848**	0.808	14.882
	RHSEG	**5**	3383	7.888	49.315	**14.500**
	RCRM	21	597	2.943	**0.600**	14.621
46076	CSC	167	235	0.444	0.151	12.954
	EDISON	34	**82**	**0.227**	0.208	**12.814**
	RHSEG	3	1437	2.328	3.748	12.852
	RCRM	18	223	0.440	**0.109**	12.913
60079	CSC	134	241	0.445	0.091	**9.251**
	EDISON	14	39	0.081	0.109	9.249
	RHSEG	3	1622	0.672	5.475	9.572
	RCRM	10	**33**	**0.019**	**0.010**	9.399

The smallest values of F, Q and E (in bold in Table 1) denote the best performance with respect to that goodness measure. As regards NR, note that a too small (large) value could denote under-segmentation (over-segmentation). It is then difficult to say whether the smallest (largest) value of NR denotes the best performance. To facilitate

reading of Table 1, we have set in bold the smallest value for NR. As for NC, we just give it as an indication of the performance by taking into account the complexity of the input image, but don't consider it as an evaluation measure. In the majority of cases, RCRM has a better performance for a larger number of goodness measures. However, we remark that counting the number of goodness measures for which the performance of a given segmentation method is better with respect to that of other methods is not enough to claim that a given method outperforms with respect to the others. The results should also be at least qualitatively evaluated.

Fig. 3. From left to right, segmentation results with RCRM, CSC, EDISON and RHSEG

5 Concluding Remarks

We have presented the algorithm for image segmentation RCRM based on the detection of representative colors in the 3D histogram and on region merging. Colors present in the input image *I* are considered as possible representative colors on the basis of their frequency in the histogram and are processed in decreasing frequency order. The currently processed color is taken as a representative color and a grouping process is accomplished to gather with it all colors with smaller frequency, provided that they have not yet been grouped with any previous representative color and are such that their distance from the currently processed representative color is smaller than an a priori fixed value, set depending on color distribution. Once representative colors have been selected, the preliminary segmented image *I'* is built by colormapping. Since *I'* is likely to result over-segmented, a merging process is accomplished. Only regions with size smaller than an a threshold, whose value is related to the arithmetic mean of the areas of all regions of *I'*, undergo to merging. Regions are actually merged only provided that their colors do not significantly differ from the colors of the adjacent regions that will absorb their pixels.

The values for the four parameters involved by the process are automatically computed by taking into account properties evaluated on the image at hand. RCRM has been tested on about 100 RGB color images with different size and color distribution, and the obtained results can be regarded as generally satisfactory.

References

1. Cheng, H.D., Jiang, X.H., Sun, Y., Wang, J.: Color image segmentation: advances and prospects. Pattern Recognition 34, 2259–2281 (2001)
2. Busin, L., Vandenbroucke, N., Macaire, L.: Color spaces and image segmentation. In: Advances in Imaging and Electron Physics, ch. 2, JCR Science Edition, Orlando, FL, USA, vol. 1, pp. 65–168 (2008)
3. Bhattacharyya, S.: A Brief Survey of Color Image Preprocessing and Segmentation Techniques. J. of Pattern Recognition Research 1, 120–129 (2011)
4. Ramella, G., Sanniti di Baja, G.: Color histogram-based image segmentation. In: Real, P., Diaz-Pernil, D., Molina-Abril, H., Berciano, A., Kropatsch, W. (eds.) CAIP 2011, Part I. LNCS, vol. 6854, pp. 76–83. Springer, Heidelberg (2011)
5. Lloyd, S.P.: Least squares quantization in PCM. IEEE Trans. Information Theory 28(2), 129–136 (1982)
6. Berkhin, P.: Survey of clustering data mining techniques. Accrue Software, San Jose, CA. Technical Report (2002)
7. Zhang, H., Fritts, J., Goldman, S.: An entropy-based objective evaluation method for image segmentation. In: Proceedings of SPIE 5307 Storage and Retrieval Methods and Applications for Multimedia, pp. 38–49 (2004)
8. Rehrmann, V., Priese, L.: Fast and robust segmentation of natural color scenes. In: Chin, R., Pong, T.-C. (eds.) ACCV 1998. LNCS, vol. 1351, pp. 598–606. Springer, Heidelberg (1997)
9. Comaniciu, D., Meer, P.: Mean Shift: A Robust Approach Toward Feature Space Analysis. IEEE Trans. Pattern Anal. Machine Intell. 24(5), 603–619 (2002)
10. Tilton, J.: D-dimensional formulation and implementation of recursive hierarchical segmentation, Disclosure of Invention and New Technology: NASA Case No. GSC 15199-1 (2006)

Integer Wavelet Transform Based Watermarking System for PCM Signals

Mario González-Lee[1], Luis J. Morales-Mendoza[1], Rene F. Vázquez-Bautista[1], Celia Calderon-Ramon[1], Héctor Pérez-Meana[2], and Mariko Nakano-Miyatake[2]

[1] Facultad de Ingenieria en Electronica y Comunicaciones, Universidad Veracruzana
Av. Venustiano Carranza S/N, Col. Revolucion.
Poza Rica Veracruz, Mexico
{mgonzalez01,javmorales,favazquez}@uv.mx
[2] Seccion de Estudios de Posgrado e Investigacion
Instituto Politecnico Nacional
Av. Santa Ana No. 1000, Col. San Francisco Culhuacan
Mexico D.F.

Abstract. In this paper, a watermarking system based on the integer wavelet transform is proposed. A set of different wavelets was chosen for this propose so their performance as watermark transmission channel can be analyzed. The watermark is embedded in the second decomposition level of the wavelet transform of a PCM signal in a block wise approach, the coefficients of the wavelet transform are assumed to be accurately modeled as a Laplacian channel. A set of attacks were performed in order to test system's robustness, we found that this approach is robust against several attacks such as additive white noise, low pass filtering, cropping, among other attacks; the best suited wavelet class for developing watermarking systems was identified too.

Keywords: Watermarking, Integer wavelet transform, Optimal detector, Laplacian channel.

1 Introduction

A very handy approach to watermarking is to establish analogies to the very strong field of the theory of communications, in this context, we can think of a watermark as a signal that propagates through a communications channel.

The watermarking model is shown in Fig. 1, we can identify the main input variables: the cover, which is an audio signal that will carry the watermark, a user's key which is used to generate a pseudo-random signal and the embedding gain which is related to the embedding energy of the watermark.

In this work, a watermark is a binary signal $\mathbf{W} = [w_i]$ with $w_i \in \{-1, 1\}$ and is zero mean with variance 1. This watermark is embedded in the cover $\mathbf{X} = [x_i]$ so we get the watermarked signal $\mathbf{Y} = [y_i]$; Ideally, the cover does not interfere the watermark, however in practice this is not true, in consequence, we model the effects of the cover within the channel block, and attacks to the watermark are modeled as noise in the channel during the propagation of the watermark.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 185–194, 2013.

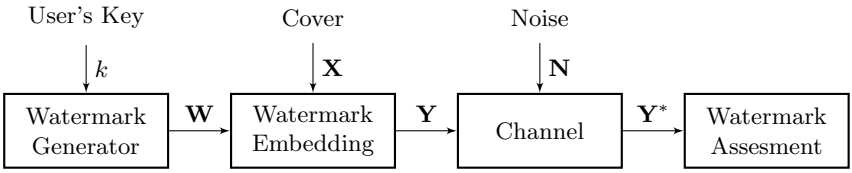


Fig. 1. Watermark propagation model

Once the watermark reaches the detector, it has to assess the presence of the watermark, usually by computing two statistics, one that measures the presence of the watermark in the possibly watermarked audio signal and the other is a threshold. If the computed statistics surpasses the threshold value, the watermark is detected, otherwise, the watermark is considered to be absent. The computed statistics are often known as the decision variable d and the decision threshold Th respectively.

The remaining of this paper is organized as follows in Sect. 1.1 related works are discussed, then in Sect. 1.2 we will introduce the integer wavelet transform, in Sect. 2 the proposed system is discussed in detail and the corresponding experimental results are presented in Sect. 3; finally the conclusions in Sect. 4 and references.

1.1 Related Works

There are some works in the field of audio watermarking that make use of the Laplacian channel model, for example, in [1] a method that uses MPEG 1 Layer 3 compression as a reference to determine where and how the watermark must be embedded. The results show that their watermarking scheme is robust against the attacks of the audio Stirmark benchmark.

The authors of [2] change the length of the intervals between salient points of the audio signal to embed data. Their results suggest that the algorithm is robust to common audio processing operations e.g. MP3 lossy compression, low pass filtering, and time-scale modification. The major drawback of this proposal is its low bit embedding rate.

In [3], the analysis filterbank decomposition, the psychoacoustic model and the empirical mode decomposition (EMD) techniques are used. This algorithm embeds the watermark bits in the final residue of the subbands in the transform domain. The authors claim that the scheme is robust against MP3 compression and Gaussian noise attacks. A drawback is that it might not be robust to common attacks such as band-pass filtering and cropping.

The adaptive tabu search (ATS) has been explored in order to develop watermarking systems, in [8], the Daubechies wavelet decomposition is used for watermark embedding. The optimal intensity of watermark is searched by using the ATS. Experimental results show that watermark is inaudible and robust to many digital signal processing, such as resamplig, cropping, low pass filter, additive noise and MP3 compression.

In [4,5], the authors propose to use the Laplacian channel model for audio signal in temporal domain, they derive an optimal detector and the threshold equation, their system is semi-fragile, but has several advantages, first, since no transform domain is involved, the resulting system has very low computational complexity, second, memory requirements could be easily satisfied, and finally, the system performs very well for attacks such as additive white noise, cropping and echo attacks.

In this paper, propose a system in the Integer Wavelet Transform domain assuming they can be accurately modeled as a Laplacian communication channel, so the optimal detector equation and the threshold equation proposed in [4,5] can be used for detecting watermarks. Several Wavelets were tested in order to select the best suited wavelet for a robust watermarking system. Once identified, a watermarking system was derived, the resulting scheme has low complexity and it proved to perform remarkably well, furthermore, it proved to be unaffected for low pass filtering attack and additive white noise and phase inverting attack whilst it is almost unaffected for cropping attack and echo attacks.

1.2 Integer Wavelet Transforms

An Integer Wavelet Transform (IWT) maps a set of integers into another set of integers, this property is well suited for signal coding for example in lossless image compression [7]. In our case, integer coefficients reduce damage to embedded watermark caused by quantization, in addition, the well know advantages of the IWT can be exploited for robust watermarking systems design.

The following equations are mathematical definitions of several IWT used in this work, here the notation IWT(m,n) is used to denote an IWT with m and n vanishing moments in the analysis and synthesis high pass filters respectively ([6,7]).

IWT(1,1):

$$d_n = x_{2n+1} - x_{2n} \quad , \tag{1}$$

$$s_n = x_{2n} + \left\lfloor \frac{d_n}{2} \right\rfloor \quad . \tag{2}$$

IWT(2,4):

$$d_n = x_{2n+1} - \left\lfloor \frac{1}{2}(x_{2n} + x_{2n+2}) + \frac{1}{2} \right\rfloor \quad , \tag{3}$$

$$s_n = x_{2n} + \left\lfloor \frac{19}{64}(d_{n-1} + d_n) - \frac{3}{64}(d_{n-2} + d_{n+1}) + \frac{1}{2} \right\rfloor \quad . \tag{4}$$

IWT(4,2):

$$d_n = x_{2n+1} - \left\lfloor \frac{9}{16}(x_{2n} + x_{2n+2}) - \frac{1}{16}(x_{2n-2} + x_{2n+4}) + \frac{1}{2} \right\rfloor \quad , \tag{5}$$

$$s_n = x_{2n} + \left\lfloor \frac{1}{4}(d_{n-1} + d_n) + \frac{1}{2} \right\rfloor \quad . \tag{6}$$

IWT(6,2):

$$\begin{aligned} & d_n = x_{2n+1} \\ & - \left\lfloor \frac{75}{128}(x_{2n} + x_{2n+2}) - \frac{25}{256}(x_{2n-2} + x_{2n+4}) + \frac{3}{256}(x_{2n-4} + x_{2n+6} + \frac{1}{2}) \right\rfloor \quad , \end{aligned} \tag{7}$$

$$s_n = x_{2n} + \left\lfloor \frac{1}{4}(d_{n-1} + d_n) - \frac{1}{2} \right\rfloor \quad . \tag{8}$$

In this paper, we denote the coefficient sequence ant the *k-th* decomposition level of any given signal as $\mathbf{S}_{(k)} = [s_{(k)i}]$ and its related detail sequence as $\mathbf{D}_{(k)} = [d_{(k)i}]$. The operator $\lfloor \cdot \rfloor$ denotes the floor operation.

The IWT equation set just introduced is going to be used for computing the IWT for the proposed watermarking system discussed right away.

2 Proposed System

In this section, we introduce the proposed system, as we stated early in this paper, the system embeds a watermark in the second decomposition label of the IWT, first, the signal is divided in non overlaping blocks of length 16 times the sampling frequency, then the IWT of the blocks is computed, the detail sequences $\mathbf{D}_{(1)}$ and $\mathbf{D}_{(2)}$ are stored for later use, and the wavelet coefficients $\mathbf{S}_{(2)}$ is watermarked, the audio signal is reconstructed by reversing the transformations as shown in Fig. 2. At the receiver, the system computes the second level IWT decomposition in order to detect the watermark as shown in Fig.3.

The embedding rule and detection variables are discussed in next section.

2.1 Embedding Algorithm

The embedding algorithm is the multiplicative embedding rule since it exhibits several desirable properties, for example, one the most important is the masking effect that allows greater embedding strength while imperceptibility holds. The multiplicative watermark embedding rule is given as:

$$s'_{(2)i} = s_{(2)i}(1 + gw_i), \tag{9}$$

where $s'_{(2)i}$ is the *i-th* watermarked coefficient, w_i is the *i-th* watermark bit and g is the watermark embedding gain, which controls the watermark energy.

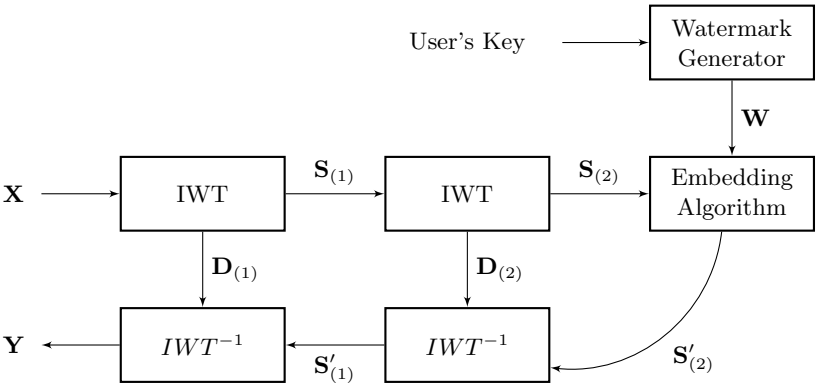


Fig. 2. Block diagran for the proposed watermarking system. IWT is the forward integer wavelet transform whilst IWT^{-1} is the inverse of IWT.

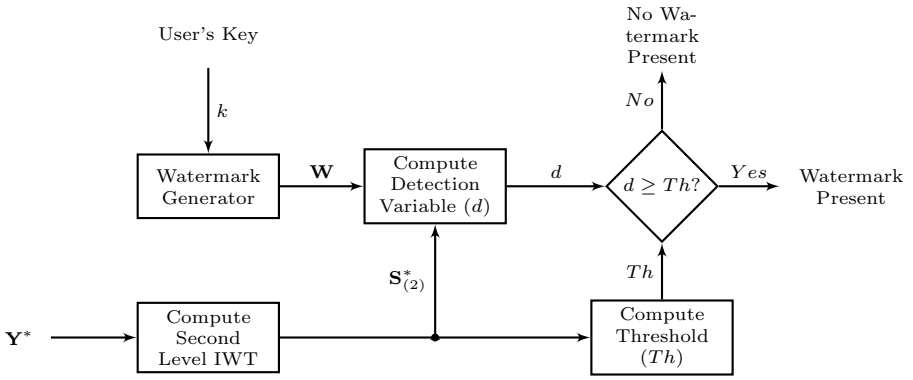


Fig. 3. Block diagram for a generic watermark detection system

2.2 Optimal Watermark Detector for Laplacian Channel

The coefficients of the IWT, and thus the channel, are considered to be statistically modeled using a Laplacian PDF, so the optimal detector variable is computed as [4,5]:

$$d = \hat{g} = \frac{1}{N\alpha_y} \sum_{i=1}^{N} |s'_{(2)i}| w_i \quad , \tag{10}$$

where,

$$\alpha_y = \frac{1}{N} \sum_{i=1}^{N} |s'_{(2)i}| \quad . \tag{11}$$

Equation (10) can not be used "as is" since the IWT coefficients must to be normalized first. We found that as one goes deeper in wavelet decomposition, the resulting coefficients must to be scaled to ensure they hold the Laplacian channel assumption, our experiments showed that in order to enforce the IWT coefficients to hold the Laplacian assumption, (10) should be scaled this way:

$$d = k\frac{1}{N\alpha_y}\sum_{i=1}^{N}|s'_{(2)i}|w_i \quad , \tag{12}$$

where $k = \frac{1}{32}$. One can normalize individual samples, however for a block size of L, L divisions must to be made, whilst by scaling (10) it takes just one.

In order to properly detect the watermark, the decision variable d must be compared to a threshold, a watermark is present if $d \geq Th$, the general threshold equation derived from the Neyman-Person criterion in [4,5] is:

$$Th = erfc^{-1}(1 - 2p_{fp})\sqrt{\frac{2}{N}}, \tag{13}$$

where $erfc^{-1}(\cdot)$ is the inverse error function complementary.

Since (13) does not depend upon the samples, then no scaling is needed for this equation. Equations (12) and (13) are meant to be used for detecting watermarks in the IWT domain, the corresponding experimental results are shown in next section.

3 Computer Simulations

All test were carried out under the following scenario: the watermark was embedded in non overlapping blocks with length of 16 times the sampling frequency of an audio signal using (9). Detection is made in the same block wise approach, d and Th are computed for each block using (10) and (13) and the responses for each block are accumulated and averaged. We let $P_{fp} = 10^{-6}$ and the embedding gain was set to 0.5. All audio signal used for our tests were uncompressed 16-bit stereo WAV files with 48000 Hz sampling rate.

We repeated the same attacks using the wavelets defined in section 1.2, so we can identify which of them has the best performance for practical watermarking system design. The performance of the system is presented in next section.

3.1 Main Results

In this section we present simulation results that validate that (12) and (13) provide an accurate watermark detection model. Several audio were used in the following test, however, we show our worst case.

First we tested detector performance; we present in Fig.4 the response for IWT(4,2). Given that watermark number 500 is the watermark that was embedded in an audio signal. One can verify that the response is way bigger that

any other watermark, furthermore, the response to any watermark different to the one embedded is far from crossing the decision threshold which confirms the Laplacian channel model is accurate for IWT(4,2) coefficients.

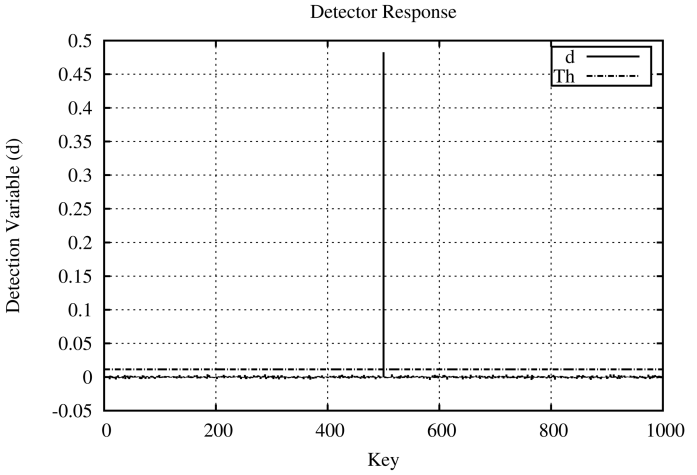


Fig. 4. Detector characteristics. It is shown that the system performs remarkably for IWT(4,2) even for our worst case host signal.

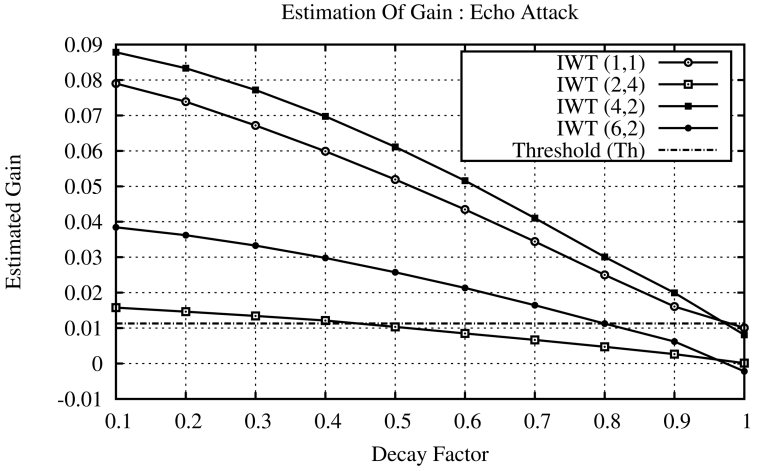


Fig. 5. Watermark detection for different IWT for various values of the decay factor

In the second test, we added a echo signal, The echo signal has a delay of 1 second for a given value of decay factor which determines how fast the echo fades away. In Fig.5 the response for various values of the decay factor is shown, it is clear that IWT(4,2) perform better whilst IWT(2,4) performed poorly.

In the next test, additive white noise was added to the carrier signal, the detection characteristic for several values of amplitude of the noise is shown in Fig.6. It can be seen that again IWT(4,2) outperforms the other wavelet decompositions and once again IWT(2,4) has the worst performance.

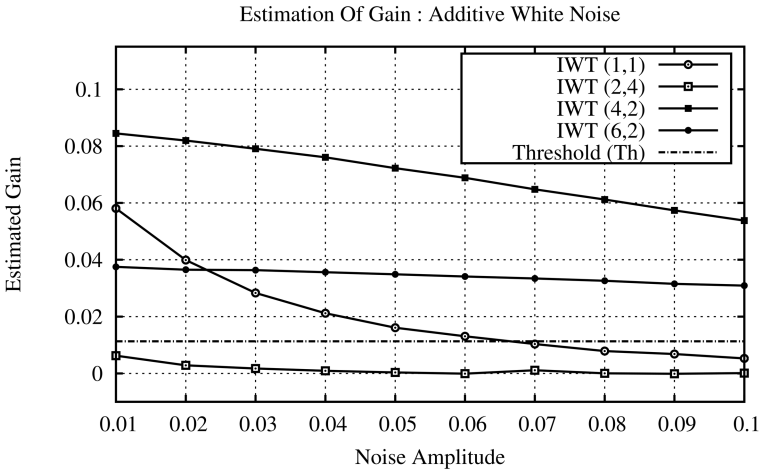


Fig. 6. Detector characteristics for additive white noise at various amplitudes

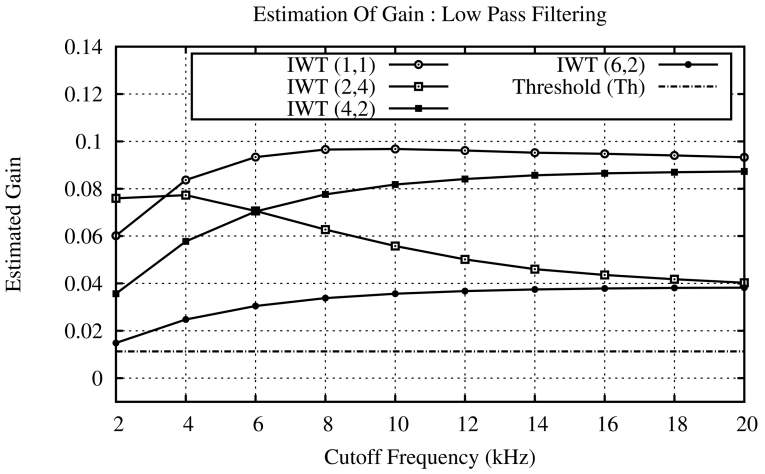


Fig. 7. Low pass filtering attack for various cutoff frequencies

A low pass filter was applied and detector response was measured for various cutoff frequencies, the results are shown in Fig.7, in this test all wavelets exhibit very good performance against low pass filtering with IWT(1,1) being the best and IWT(6,2) was the worst.

Another attack is cropping, Fig.8 shows detection characteristics as the cropping percentage is varying. In this case IWT(4,2) performed best again and once more IWT(2,4) performed poorly.

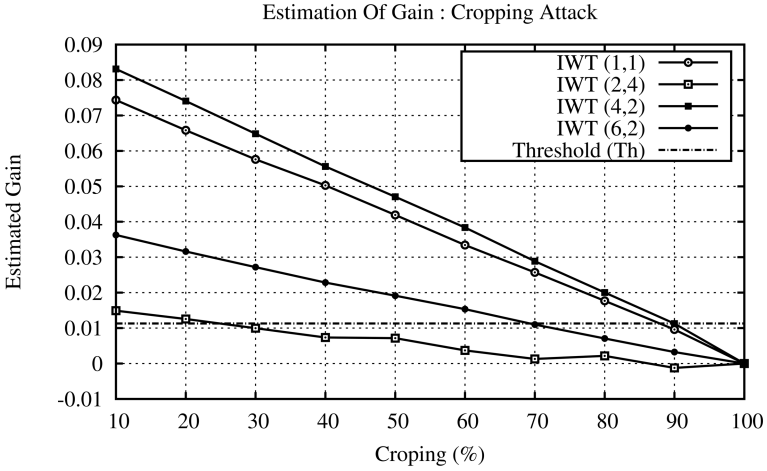


Fig. 8. Cropping attack; horizontal axis is the cropping percentage

Inverting attacks exploits the fact that the human auditory system is unable to perceive phase changes, the system response is $d = 0.0813$, $d = 0.0162$, $d = 0.0899$, $d = 0.0394$ for wavelets IWT(1,1), IWT(2,4), IWT(4,2) and IWT(6,2) respectively ,threshold value is $Th = 0.0113$. So the system is not affected by inverting attacks and once more, IWT(4,2) is clearly the winner.

These results clearly show a trend: the IWT(4,2) is best in almost all cases whilst IWT(2,4) is worst in almost all cases.

4 Conclusions

We can draw several conclusions, first, IWT based watermarking proved to be more robust than other previously proposed systems, it has the additional advantage of being a blind algorithm, so the original audio is not needed.

Second, IWT(4,2) is the best choice among the set of wavelets tested in this paper whilst IWT(2,4) performed worst in almost all test. Furthermore, a test not shown in this paper using IWT(2,2) performed so bad that we decided not to include it in the tests presented in this work, its poor performance prevented the proposed structure from detecting watermarks accurately even when no attacks were made.

The resulting system has low complexity so the detection of the watermark can be done in a couple of seconds for a full length song using a consumer laptop.

Future works include: Analyzing the performance of other classes of IWT and to investigate the influence of different decomposition levers in the wavelet transform and the proper derivation of the scaling factor of (12).

Acknowledgments. The authors want to thank the FIEC Poza Rica and the Universidad Veracruzana for their support to this work.

References

1. Megias, D., Herrera-Joancomarti, J., Minguillon, J.: A robust frequency domain audio watermarking scheme for monophonic and stereophonic PCM formats. In: Procedings of the 30th Euromicro Conference, pp. 1089–6503 (2004) ISSN 1089-6503
2. Mansour, M.F., Tewfik, A.H.: Audio watermarking by time-scale modification. In: Proceedings of the 2001 IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP 2001), vol. 3, pp. 1353–1356 (2001) ISSN 1520-6149
3. Wang, L., Emmanuel, S., Kankanhalli, M.S.: EMD and psychoacoustic model based watermarking for audio. In: Procedings of the 2010 IEEE International Conference on Multimedia and Expo (ICME), pp. 1427–1432 (July 2010) ISSN 1945-7871
4. Gonzalez-Lee, M., Morales-Mendoza, L.J., Vazquez-Bautista, R.F., Morales-Mendoza, E.: Optimal Detection of Multiplicative Watermarks in Laplacian Channels. In: Proceedings of the 9th International Conference on Electrical Engineering, Computing Science and Automatic Control, Mexico City, Mexico (2012)
5. Gonzalez-Lee, M., Morales-Mendoza, L.J., Vazquez-Bautista, R.F., Morales-Mendoza, E.: Semi Fragile Watermarking System in Temporal Domain. In: CERMA 2012, Cuernavaca Morelos, Mexico (2012)
6. Calderbank, R., Daubechies, I., Sweldens, W., Yeo, B.L.: Lossless Image Compression Using Integer To Integer Wavelet Transform. In: Proceedings of the International Conference on Image Processing, vol. 1, pp. 596–599 (October 1997)
7. Sheng, F., Bilgin, A., Sementilli, P.J., Marcellin, M.W.: Lossy and Lossless Image Compression Using Reversible Integer Wavelet Transform. In: Proceedings of the 1998 International Conference on Image Processing, vol. 3, pp. 876–880 (1998)
8. Sriyingyong, N., Attakitmongcol, K.: Wavelet-Based Audio Watermarking Using Adaptive Tabu Search. In: Proceedings of the 2006 1st International Symposium on Wireless Pervasive Computing, pp. 1–5 (2006)

Enhancing CCL Algorithms by Using a Reduced Connectivity Mask

Uriel H. Hernandez-Belmonte, Victor Ayala-Ramirez,
and Raul E. Sanchez-Yanez

Universidad de Guanajuato DICIS
hailehb@laviria.org, {ayalav,sanchezy}@salamanca.ugto.mx

Abstract. In this paper, we propose to use a Reduced Connectivity Mask (RCM) in order to enhance connected component labeling (CCL) algorithms. We use the proposed RCM (a 2×2 spatial neighborhood) as the scanning window of a Union-Find labeling method and of a fast connected component labeling algorithm recently proposed in the literature. In both cases, the proposed mask enhances the performance of the algorithm with respect to the one exhibited by each algorithm in its original form. We have tested the two enhanced approaches proposed here against the fast connected component labeling algorithm proposed by He and the classical contour tracing algorithm. We have compared their execution time when labeling a set of uniform random noise test images of varying occupancy percentages. We show detailed results of all these tests. We also discuss the differences in behavior shown by the set of algorithms under test. The RCM variants exhibit a better performance than the previous CCL algorithms.

1 Introduction

Main tasks in computer vision include the detection and recognition of objects of interest in a binary image. Binary images are often the result of a feature detection step, where a binary feature is detected in a pixel-wise approach. This problem arises in image analysis tasks such as figure-background discrimination and motion detection, as two examples. Typically, images are preprocessed accordingly with the task and they are thresholded to output a binary image. A second step involves the association of connected pixels into single entities to be handled by further processing steps. A set of methods to perform this task are known as Connected Components Labeling (CCL) algorithms. The goal of a CCL algorithm is to label all the foreground pixels in a binary image. The label for each foreground pixel has to be the same for all the pixels belonging to the same connected component. Each connected component has to be labeled with a unique identifier. Two pixels are said to belong to the same component if there exists a path that links them.

Recent work on CCL has been directed towards: i) reducing the number of scans and ii) optimizing the connectivity mask and the definition and usage of the data structures for the labeling procedure. For the first case, the contour tracing

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 195–203, 2013.

algorithm (Chang, Chen & Lu 2004) is an example. In this kind of methods, there is only a single scanning pass to the image because a set of heuristic rules guides the decision of what pixel is the next requiring analysis. For the second concern, two of the reference works are: i) the Union-Find approach, originally proposed by Fiorio and Gustedt (Fiorio & Gustedt 1996) and further enhanced by Wu *et al.* (Wu, Otoo & Shoshani 2005) and ii) the Connection-List approach proposed recently by He *et al.* (He, Chao & Suzuki 2008).

We propose to use a 2×2 neighborhood as the spatial support of the connectivity mask for the CCL task. The rationale behind this design choice is that in such a region, all the pixels are connected among themselves and that the use of such a reduced connectivity mask (RCM) will speed up the computation of the label assignment for all the foreground pixels of the input image. The connectivity analysis implied by this shape of mask is performed by using a sort of operational look-up table. For any pixel under analysis, we access the look-up table entry that corresponds to the actual configuration of the neighborhood to determine what operations need to be executed. In this work, we compare the performance of our method against the enhanced Union Find approach (UF) (Wu, Otoo & Shoshani 2005) and to the Connection List (CL) method (He et al. 2008).

Rest of this paper is organized as follows: In Section 2, we review previous work for the CCL problem. The CCL problem is formulated in Section 3. A detailed presentation of the proposed method is addressed in Section 4. Results and their discussion are presented in Section 5. Finally, we present our main findings and conclusions in Section 6.

2 Taxonomy of CCL Methods

A number of methods have been developed to address the connected component labeling task. Suzuki *et al.* (Suzuki, Horiba & Sugie 2003) and Wu *et al.* (Wu, Otoo & Suzuki 2005) have classified the CCL methods in four main classes accordingly to the characteristics of the algorithms:

i) Single-pass algorithms like those proposed by (Chang et al. 2004) and (Chang & Chen 2003),
ii) Multi-pass algorithms, see, for example, (He, Chao & Suzuki 2007), (Wu, Otoo & Shoshani 2005), (Fiorio & Gustedt 1996), (Di Stefano & Bulgarelli 1999), (He et al. 2008), (Asano & Tanaka 2010), (He, Chao & Suzuki 2010a), (He, Chao, Suzuki & Wu 2009) ,(Wu, Otoo & Suzuki 2008),
iii) Parallel computing approaches ((Greiner 94), (Han & Wagner 1990), (Hardwick 1997), (Krishnamurthy, Lumetta, Culler & Yelick 1994)), and
iv) Hierarchical image representation methods ((Samet 1981), (Wang & Davis 1986), (Dillencourt & Samet 1992), (Samet & Tamminen 1988)).

In the three first classes (i, ii and iii) of the above taxonomy, the image under analysis is represented as a data matrix whose elements will be assigned a connected component label as the output of the algorithm.

Class iv) algorithms, instead, use more complex data structures to represent the binary image. Examples of data structures used for this end include bin-trees and quad-trees. In this work, we will focus on issues related to methods that use a data matrix representation of the image under analysis and it is out of the scope of this paper to study further details of the CCL methods belonging to the fourth category in the enumeration above.

3 Methodology

3.1 Overview of a CCL Algorithm

Any CCL algorithm needs to scan the input image a number of times depending on the scanning strategy, the connectivity mask used and the label equivalence handling method.

As we traverse the image on the first scan, we assign temporary labels to the foreground pixels. In order to ensure that all the pixels belonging to the same connected component are marked with the same label, each CCL algorithm uses a particular equivalence handling method. Some approaches use a set of heuristic rules (Chang et al. 2004) and some others use particular data structures designed for that purpose (Fiorio & Gustedt 1996), (He et al. 2009). Further scans of the input image may be needed to resolve the label assignment for all the foreground pixels.

In each traversal, the image is scanned by using a connectivity mask (CM) that serves to analyze the connectivity among the pixels in the neighborhood defined by the spatial support of the CM. Different authors have proposed different CMs. We show several examples of previously used CMs in Figure 1. The spatial configuration of the CM is a critical factor in the performance of any CCL algorithm.

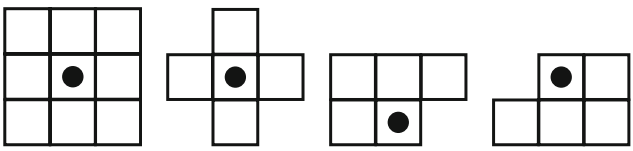
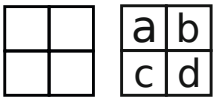


Fig. 1. Typical shapes of neighborhood masks used for connectivity analysis in CCL algorithms, the pixel under analysis is marked on them: (a) 3×3 mask, (b) 4-neighbor mask, (c) forward scan mask and (d) backward scan mask

3.2 Reduced Connectivity Mask for CCL Algorithms

Finding ways to simplify the connectivity analysis step is an important way of reducing the computational load required by CCL methods. That is why we propose a reduced connectivity mask (RCM) that covers only 2×2 pixels. Figure 2 shows the RCM and the notation used for the pixels in this mask. For this type of mask we do not need to use a central pixel to perform the analysis.



(a) (b)

Fig. 2. Reduced Conectivity Mask: (a) the spatial neighborhood, and (b) the notation used to identify each pixel in the mask

4 Our Approach

We analyze the use of two labeling strategies: i) Union-Find, originally proposed by Fiorio and Gustedt (Fiorio & Gustedt 1996) and enhanced by Wu *et al.* (Wu, Otoo & Suzuki 2005) and ii) the Connection-List approach proposed by He *et al.* (He et al. 2008), but using a 2×2 connectivity (RCM) mask instead of their choices for the connectivity mask. We call these approaches as UF-RCM and CL-RCM respectively.

4.1 Labeling Procedure

The labeling procedure in CCL algorithms that use a RCM requires two image scans. The first scan is used to assign a temporary label to each foreground pixel in the image. The second scan is used to resolve the equivalence relationships between labels and to assign definitive labels to all foreground pixels in the image.

Table 1. Look-up table operations associated to each neighborhood configuration in the proposed approaches

Row	a	b	c	d	$L(x,y)$	*Union-Find*	*Connection-List*
0	0	0	0	0	NOP	NOP	NOP
1	0	0	0	1	m+1	NOP	NOP
2	0	0	1	0	NOP	NOP	NOP
3	0	0	1	1	c	NOP	NOP
4	0	1	0	0	NOP	NOP	NOP
5	0	1	0	1	b	NOP	NOP
6	0	1	1	0	NOP	$Union(c,b)$	$resolve(c,b)$
7	0	1	1	1	b	$Union(c,b)$	$resolve(c,b)$
8	1	0	0	0	NOP	NOP	NOP
9	1	0	0	1	a	NOP	NOP
10	1	0	1	0	NOP	NOP	NOP
11	1	0	1	1	c	NOP	NOP
12	1	1	0	0	NOP	NOP	NOP
13	1	1	0	1	a	NOP	NOP
14	1	1	1	0	NOP	NOP	NOP
15	1	1	1	1	a	NOP	NOP

(a) (b)

Fig. 3. Look up table operations: (a) The only case where a new label is created for the pixel under analysis. (b) RCM configuration where two labels are merged.

The first step in label handling is to initialize the structures for each specific approach. In the case of the UF method, the *Parent* vector is initialized, whilst in the CL approach, *next*, *tail* and *rtable* arrays are initialized.

The input image is augmented by a dummy row and a dummy column composed of background pixels that are located before the first row and the first column of the image under analysis, respectively. These supplementary pixels do not interfere with the connected components actually present in the image. Conversely, they help to reduce the number of conditions to analyze in the connectivity mask enabling a faster processing in the connectivity analysis step. The scanning of the augmented image is performed in a left to right, top to bottom sequence.

Table 1 summarizes the operations needed to analyze the connectivity of each pixel. Figure 2 shows the position of each pixel (a, b, c and d) in the Reduced Connectivity Mask proposed in this paper.

We can identify four different operations on this table:

- The first case is the NOP (no operation) circumstance. We do not perform any operation when the pixel d (the only pixel not yet analyzed in the RCM) is a background pixel.
- The second operation occurs when only one of the pixels a, b, c is a foreground pixel and it has already a label assigned. Under such a condition, the label to assign to the pixel under analysis is the same label of the already assigned pixel. This operation is needed for the cases in rows 3, 5, 7, 9, 11, 13 and 15.
- The third operation needed is a label creation operation. That operation requires that the pixel under analysis be the only foreground pixel in the neighborhood defined by the connectivity mask. The operation is defined as $m = m + 1$ because we need to increment m, the number of labels currently being used. We perform this operation only for the case in row 1.
- The fourth operation is the connectivity analysis for both labeling methods. The *Union* procedure for the Union-Find method, or the *resolve* operation for the Connection-List algorithm, is used to determine what is the correct label from the different labels already assigned to the pixels in the neighborhood. The cases for this operation are those presented in rows 6 and 7.

As said before, Table 1 specifies the operation needed for each condition that can present the 4-uple (a, b, c, d). In columns 1-4, a "0" stands for a background

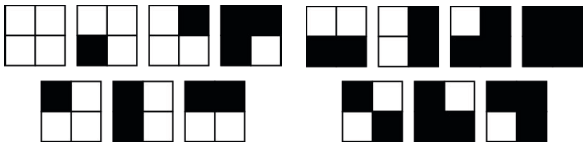


Fig. 4. Look up table operations: (a) RCM configurations where no operation (NOP) is needed. (b) RCM configurations where the label is copied from a previously labeled pixel.

pixel, and a "1" for a foreground pixel. Fifth column of this table shows the label to assign to the d pixel and the sixth and seventh columns show the operations needed to perform in the merge operation to store the equivalence information for each method (Union-Find for the sixth column and Connection-List for the seventh column, respectively).

We call the merge procedure for each foreground pixel in the image accordingly to the labeling method. In the case of the UF-RCM approach, we call the *Find* procedure with the label associated to the pixel under analysis. The root label found by the aforementioned procedure is then assigned as the definitive label for the pixel under analysis.

In the case of the CL-RCM method, we perform instead a look up of the *rtable* array. That is $L'(x, y) = rtable[L(x, y)]$, with $L'(x, y)$ being the definitive label for the pixel under analysis. After completing the second scan, all the foreground pixels are correctly labeled.

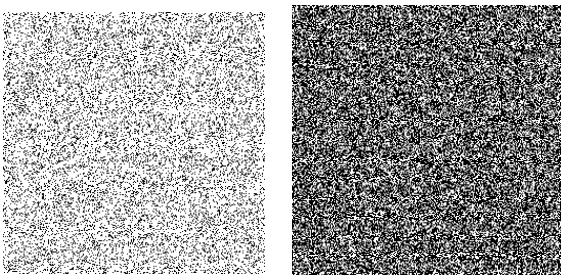
5 Results

5.1 Test Protocol

In order to evaluate our algorithm, we have experimented with a set of uniform random noise images to measure the execution time. We have compared the performance of the Contour Tracing (CT) algorithm (Chang et al. 2004), and the fast connected component labeling method (He et al. 2009) to the proposed approaches, UF-RCM and CL-RCM. The comparison of the proposed approach to these methods can provide us some insight because their particular differences.

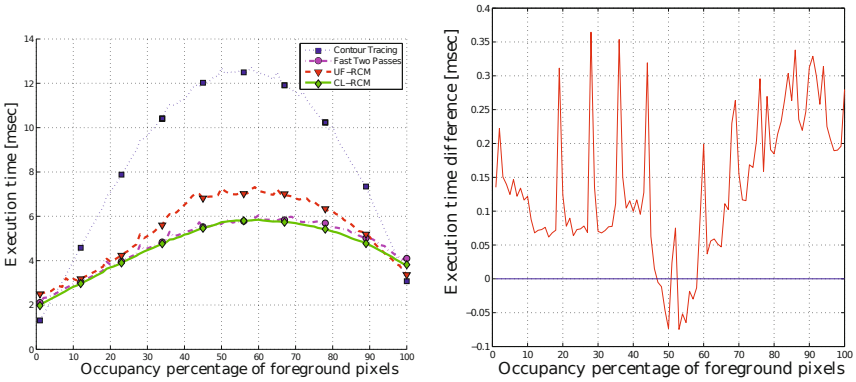
We have generated test images using a uniform random noise across the image. The goal of this test set is to demonstrate how the algorithm performs on random input (see Figure 5). We have used eleven occupancy percentages for the foreground pixels in the images (0 to 100 percent in 10 percent steps). We have used a size of 512×512 pixels for all the images in this data set. We have generated 40 random test images for each occupancy percentage, and we used all these images as input.

We have developed all these tests using C-language implementations for all the methods in an Intel Dual Core-based desktop computer with a processor running at 2.66 GHz and 4 GB RAM memory. The execution time figures are the average in 40 runs for each test.



(a) (b)

Fig. 5. Two examples of uniform random noise images with varying occupation percentage of the foreground pixels: (a) 20 % and (b) 60 %



(a) (b)

Fig. 6. (a) Performance results of the four methods using uniform random noise images of varying occupancy percentage of the foreground pixel as input. (b) Difference in time execution between the He's method and CL-RCM approach. A positive value implies a better performance of the CL-RCM algorithm.

5.2 Results

In Figure 6 (a) we can observe the comparison of the performance exhibited by the four methods under analysis when uniform random noise images are used as inputs. In this graph, we can observe that CL-RCM method performs the best, even if it is only slightly faster than He's method. The marginal advantage (see Figure 6 (b) for a direct comparison) of the CL-RCM method over the faster connected component labeling algorithm (He's method) is due to the reduced size of the connectivity mask used by that approach. The improvement in time is the same that the others authors in the field, report on their works (He, Chao & Suzuki 2010b) . We can also see that the UF-RCM method exhibits a poorer performance than the CL-RCM and the fast connected component

labeling method. Nevertheless, its performance is significantly better than the one exhibited by the Contour Tracing algorithm in this test.

6 Conclusions

In this paper, we have proposed the use of a reduced connectivity mask (RCM) in two CCL algorithms: the Union-Find approach previously proposed by Wu and the fast connected component labeling method proposed by He *et al.* We call these approaches UF-RCM and CL-RCM respectively. The RCM consists of a 2×2 pixel neighborhood as the scanning window. As we have seen in a set of tests involving random noise and typical benchmark pattern images, the RCM-based approaches proposed here exhibit an overall performance that is better than the others performance with respect to execution time.

We have also observed that the optimal labeling strategy strongly depends on the structure shown by the components in the input image. In the near future, we are going to explore other data structures to handle label equivalence that improve the overall method performance. We will work to automatically decide the CCL algorithm best suited to deal with a given input image, according to its content. That will also serve to improve memory usage required by our algorithm. We are also going to explore its applications in more complex tasks, as for example, motion detection clustering and optical character recognition.

Acknowledgements. Uriel H. Hernandez-Belmonte gratefully acknowledges Mexico's CONACYT (329356/229784) for the financial support through the scholarships given by the programs "Convocatoria de Becas Nacionales 2013 Primer Periodo" and "Convocatoria de becas CONACYT nacionales agosto - octubre 2009".

References

Asano, T., Tanaka, H.: In-place algorithm for connected components labeling. Journal of Pattern Recognition Research 5(1) (2010)

Chang, F., Chen, C.-J.: A component-labeling algorithm using contour tracing technique. In: International Conference on Document Analysis and Recognition, pp. 741–745. IEEE Computer Society (2003)

Chang, F., Chen, C.-J., Lu, C.-J.: A linear-time component-labeling algorithm using contour tracing technique. Computer Vision and Image Understanding 93(2), 206–220 (2004)

Di Stefano, L., Bulgarelli, A.: A simple and efficient connected components labeling algorithm. In: International Conference on Image Analysis and Processing, pp. 322–327 (1999)

Dillencourt, M.B., Samet, H.: A general approach to connected-component labeling for arbitrary image representations. Journal of the ACM 39, 253–280 (1992)

Fiorio, C., Gustedt, J.: Two linear time Union Find strategies for image processing. Theoretical Computer Science 154(2), 165–181 (1996)

Greiner, J.: A Comparison of Data-Parallel Algorithms for Connected Components. In: Proceedings Symposium on Parallel Algorithms and Architectures, pp. 16–25 (1994)
Han, Y., Wagner, R.A.: An efficient and fast parallel-connected component algorithm. Journal of the ACM 37(3), 626–642 (1990)
Hardwick, J.: Practical Parallel Divide-and-Conquer Algorithms, PhD thesis, School of Computer Science, Carnegie Mellon University (1997)
He, L., Chao, Y., Suzuki, K.: A linear-time two-scan labeling algorithm. In: IEEE International Conference on Image Processing, pp. 241–244 (2007)
He, L., Chao, Y., Suzuki, K.: A run-based two-scan labeling algorithm. IEEE Transactions on Image Processing 17(5), 749–756 (2008)
He, L., Chao, Y., Suzuki, K.: An efficient first-scan method for label-equivalence-based labeling algorithms. Pattern Recognition Letters 31(1), 28–35 (2010a)
He, L., Chao, Y., Suzuki, K.: An efficient first-scan method for label-equivalence-based labeling algorithms. Pattern Recognition Letters 31, 27–35 (2010b)
He, L., Chao, Y., Suzuki, K., Wu, K.: Fast connected-component labeling. Pattern Recognition 42(9), 1977–1987 (2009)
Krishnamurthy, A., Lumetta, S.S., Culler, D.E., Yelick, K.: Connected components on distributed memory machines. In: Parallel Algorithms: Third DIMACS Implementation Challenge, pp. 1–21. American Mathematical Society (1994)
Samet, H.: Connected component labeling using quadtrees. Journal of the ACM 28(3), 487–501 (1981)
Samet, H., Tamminen, M.: Efficient component labeling of images of arbitrary dimension represented by linear bintrees. IEEE Transactions on Pattern Analysis and Machine Intelligence 10(4), 579–586 (1988)
Suzuki, K., Horiba, I., Sugie, N.: Linear-time connected-component labeling based on sequential local operations. Computer Vision and Image Understanding 89(1), 1–23 (2003)
Wang, X., Davis, W.A.: Connected component labeling using modified linear quadtrees. In: Graphics Interface 1986, pp. 235–240 (1986)
Wu, K., Otoo, E., Shoshani, A.: Optimizing connected component labeling algorithms. In: Society of Photo-Optical Instrumentation Engineers (SPIE) Conference Series, vol. 5747, pp. 1965–1976 (2005)
Wu, K., Otoo, E., Suzuki, K.: Two strategies to speed up connected component labeling algorithms, Tech Report LBNL-59102 (2005)
Wu, K., Otoo, E., Suzuki, K.: Optimizing two-pass connected-component labeling algorithms. Pattern Analysis and Applications 12(2), 117–135 (2008)

Watershed Based Iris SEgmentation

Maria Frucci[1], Michele Nappi[2], and Daniel Riccio[3]

[1] Istituto di Cibernetica "E. Caianiello", CNR
via Campi Flegrei 34, 80078, Napoli
[2] Università degli Studi di Salerno,
via Ponte Don Melillo, 84084 Fisciano (Salerno) Italy
[3] Università degli Studi di Napoli Federico II,
via Cintia 21, 80126, Napoli
m.frucci@cib.na.cnr.it, mnappi@unisa.it,
daniel.riccio@unina.it

Abstract. Recently, the research interest on biometric systems and applications has significantly grown up, aiming to bring the benefits of biometrics to the broader range of users. As signal processing and feature extraction play a very important role for biometric applications, they can be thought as a particular subset of pattern recognition techniques. Most of iris biometric systems have been designed for security applications and work on near-infrared (NIR) images. NIR images are not affected by illumination changes in visible light making systems working both in darker and lighter conditions. The reverse of the medal is a very short distance allowed between the acquisition camera and the user, further than a strictly controlled pose of the eye. For those reasons, the viability of NIR image based systems in commercial applications is quite limited. Several efforts have been devoted to designing new iris biometric approaches on color images acquired in visible wavelength light (VW). However, illumination changes significantly affect the iris pattern as well as the periocular region making both segmentation and feature extraction harder than in NIR. In the specific case of iris biometrics, segmentation represents a crucial aspect, as it must be fast as well as accurate. To this aim, a new watershed based approach for iris segmentation in color images is presented in this paper. The watershed transform is exploited to binarize an image of the eye, while circle fitting together with a ranking approach is applied to first approximate the iris boundary with a circle. The experimental results demonstrate this approach to be effective with respect to location accuracy.

Keywords: iris segmentation, watershed, circle fitting.

1 Introduction

In the last years iris recognition system under constrained acquisition protocols has proven to be effective if compared with automatic fingerprint in several scenarios (e.g., physical access control, computer log-in, international border crossing and national ID cards) [1, 9]. However, the effectiveness and feasibility of iris

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 204–212, 2013.

technology under controlled data acquisition conditions may significantly decay in real data scenario with subjects moving and at widely varying distances under both visible wavelength and artificial lighting sources. Although deployed iris recognition systems operate mainly in the near infrared (NIR) spectrum, the majority of ICT devices, including new smart mobile generation, includes visible wavelength cameras. Just the rapid spread of mobile and its evident economic and security implications that it would have in modern societies, is the spring board of throwing for both the detection and recognition of degraded iris images acquired in visible wavelengths. So recently, the scientific community promoted two international evaluation initiatives about iris detection and recognition, named NICE I [13] and NICE II (Noisy Iris Challenge Evaluation) [14]. The UBIRIS database, including heavily degraded iris image, was used as data sources for this event.

First papers on iris detection and recognition, operating in the NIR illumination, have been proposed by Daugman [2] and Wildes [17]. Just a few years after Puhan [11] proposed a novel framework for iris analysis using data source acquired under the visible wavelength light. In this scenario the goal of some participants in NICE I was to optimize the iris localization [15]. Others aimed to reduce the computing time, as for example Li et al [6] that adopt Haar based AdaBoost technique or Labbati and Scotti [5] that propose a novel approach using a rough estimate of the center of the pupil and iris to focus search only in a limited area of the image. Along the same idea ISIS [4] operates only on iris edges extracted by Canny filter performing a precise estimate of the center and radius of the pupil with a low computing time. Iris detection is then determined by using circle fitting algorithm. Even if ISIS demonstrated to be suitable for quasi real-time applications, it always provides, as output, a circle as approximation of both pupil and iris contours.

This paper proposes a novel Watershed based Iris Segmentation, namely WISE that demonstrated to be able in providing the real iris contour and not simply a blurry circle. The watershed transform is a well known that true color and gray level image segmentation performing low asymptotical time complexity of $O(n)$ and also the real computation speed. WISE, by means of an iterative region growing transform to limit any over-segmentation uses a merging technique specifically designed for the iris context.

The output of the watershed transform suitably thresholded, is subject to a further phase of processing that produces as its final result the contours of both the iris and pupil. The rest of the paper has the following structure. Section 2 describes WISE basic terms, while Sections 3 and 4 show experimental protocol and results. Conclusions and the future work follow in Section 5.

2 How the WISE Approach Works

Iris localization (segmentation) is a pattern recognition task aimed to extract the iris texture from a snapshot of the whole periocular region. The most of the existing approaches rely on that iris outer edge is characterized by a strong contrast to the white of the sclera and to darker pixels of the eyelids. Otherwise, the inner edge

separating the iris from pupil can be more or less contrasted depending on the pigmentation of the iris texture. In both cases, the segmentation process should consider the differences in pixel contrast to accurately locate both iris boundaries. The gradient of an image comes out as a powerful tool to analyze derivatives of brightness, as it underlines the directions of its maximum intensity variations. In particular, gradient information are exploited by the watershed transform [7] to agglomerate pixels in regions, generally named basins and surrounded by gradient banks. As the number of basins tends to be very high, the watershed image is often affected by the problem of over-segmentation; so a merging criterion is implemented to remove week banks and bring together similar regions. In this specific case, the watershed transform is directly applied to an RGB color snapshot of the eye and a binary image is obtained with eyelid and sclera marked as foreground, while iris texture and pupil belonging to the background.

The foreground binary image is inputted to a morphological operator that sets a pixel to 0 if all its 4-connected neighbors are 1, thus only retaining foreground contours. Since the limbus should often be included as part of a more complex contour (see Fig. **2**-c), curvature analysis is exploited to split edges at high curvature points. Edges are treated as connected components and circle fitting is applied to find the circle that better approximate the iris boundary. Many approaches search circular shapes, possibly representing the iris, through Hough transform [17] or adaptations [8], which suffer for a high computational cost. Ellipse fitting is faster than the Hough transform, but it also shows some limitations. Indeed, the presence of noise (e.g. spurious branches) may cause the erroneous detection of an elliptical shape even where the expected result would be a circular one. WISE detects circular objects within the image using a precise and fast *circle detection* procedure presented by Taubin in [16]. The difference between Taubin's approach and *ellipse fitting* is a relevant aspect to consider; as a matter of fact, given a set of points on a plane, the former identifies a circle which better approximates them, while the latter identifies an ellipse. Since many circles are found while searching for the iris, and WISE does not rely on the Hough transform, a ranking criterion is defined in order to only select the best candidate iris circle.

2.1 The Watershed

The watershed is a segmentation approach that builds agglomerations of pixels (region growing) by using gradient ridges as a barrier for growing algorithms [12]. According to this technique, the gradient image is considered as a 3D surface characterized by ridges (banks) and catchment basins (basins). The ridges correspond to curves of maximum gradient, and each basin represents a homogeneous region in which all pixels are connected to a local minimum of the gradient. Generally, the positions of ridges produced by the watershed fits well with the way they would be placed by the human perception. Unfortunately, the number of basins may be very high, as it is proportional to the number of local minima of the gradient, thus resulting in an over-segmented image. In such cases, specific merging criteria are designed to bring together basins with very similar characteristics.

WISE exploits the watershed to transform an RGB image into a binary one, in which iris is separated as much as possible from the rest of the eye elements (sclera, eyelid, eyelashes). First of all, a median filter with window size $n \times n$ (n has been set to 7, according to the resolution of the input images 800×600 pixels) is applied to the image (Fig. **1**-b), in order to remove noise as well as to smooth sharper color differences (a first way of preventing over-segmentation). In order to produce the initial partitioning of the image, the Sobel's filter is used to extract the gradient information that is inputted to the watershed transform (Fig. **1**-c). Regions D_i produced by the watershed are a covering of the image (Fig. **1**-d), that is they are all disjoint ($D_i \cap D_j = \varnothing$), while their union gives back exactly the whole image ($\cup D_i = I$). In order to reduce the number of segmented regions, a color based merging criterion is implemented. In more details, each region D_i is assigned with its average color $C_i=(R_i, G_i, B_i)$ that is computed as the mean value of the colors of pixels belonging to it. The distance between average colors C_i and C_j of each pair of adjacent regions is computed according to the following formula (eq. 1):

$$d\left(C_i, C_j\right) = \sqrt{\left(R_i - R_j\right)^2 + \left(G_i - G_j\right)^2 + \left(B_i - B_j\right)^2}\ , \tag{1}$$

The color distance d represents the criterion WISE uses to decide if two adjacent basins must be merged into a larger region. In other words, given a basin D_i, the color distances to its adjacent basins are calculated and it is merged with the one having the minimum color distance, provided that this distance is lower than a fixed threshold δ_d (in this work δ_d has been experimentally set to 50) (Fig. **1**-e). The average color is then re-computed over all pixels belonging to the new basin (Fig. **1**-f). Assuming that white color (W) is represented by R-,G- and B-component with value 255, while black (K) has R-, G- and B-component equal to 0, the couple of distances $d(C_i,W)$ and $d(C_i,K)$ is computed for each region D_i. The mean values W_C and K_C are obtained by averaging $d(C_i,W)$ (symmetrically $d(C_i,K)$) distances over all regions D_i. Furthermore, the distance $d(W, K)$ is considered as the maximum value the distance d can assume and it equals to 441.6730. In order to separate foreground regions from the background, that is to produce a binarization of the input image, the value of the distance d must be thresholded. In other words, we need a threshold ε_d to which color distances must be compared with. Distances d_W and d_K are defined as follows (eq. 2):

$$\begin{aligned} d_W &= W_C \cdot \left(\frac{W_C}{d(W,K)} + \frac{1}{2} \right), \\ d_K &= K_C \cdot \left(\frac{K_C}{d(W,K)} + \frac{1}{2} \right), \end{aligned} \tag{2}$$

The threshold ε_d is used to binarize all regions produced by the watershed and is computed just once for a given image according to the following equation (eq. 3):

$$\varepsilon_d = d(W,K)\cdot\frac{d_K}{d_W + d_K}, \tag{3}$$

Obviously it changes from an image to another, as it depends on distances between pair of regions that come out from the watershed (Fig. **1**-g). Notice that in the binary image (Fig. **1**-g) the foreground corresponds to black regions, while the background is represented by the white ones. The separation between foreground and background in the binary image is further improved by applying a refining process that can switch the membership of a basin from foreground into background. In more details, let be C_F the average color of the foreground computed by averaging all C_i of the basins D_i that belong to it. In a symmetric way, C_B is calculated as the average color of the background. For each basin D_i in foreground the distance of the corresponding C_i to the average colors C_F and C_B is calculated; so if D_i belongs to the foreground, while holding that $d(C_i,C_B) \leq d(C_i,C_F)$, its membership is switched into background (see Fig. **1**-h).

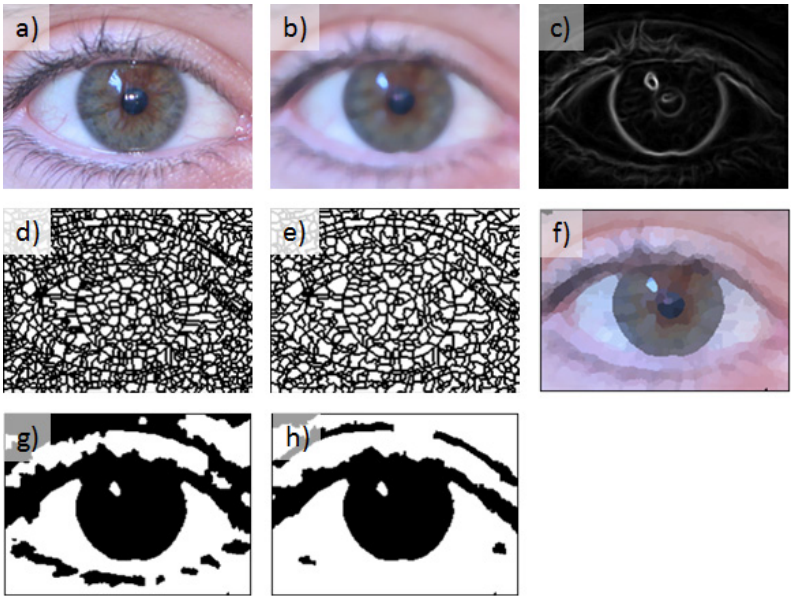


Fig. 1. Some images produced by WISE during the binarization process: a) the original colour image, b) the smoothed image, c) the gradient image, d) the output of the watershed transform (1100 basins), e) the pool of basins after the merging operation (690 basins), f) the quantized image, g) the binary image produced by thresholding the watershed, h) the improved binary image

2.2 Curvature Analysis and Circle Fitting

The output of the thresholded watershed transform is a binary image, in which the background is represented by black pixels and the foreground by white ones. As

WISE searches for a circle approximating the iris boundary, a considerably amount of computation can be saved by focusing the attention only on contour information. To this aim, an edge image is computed by applying a morphological operator to the binary mask produced by the watershed (Fig. **2**-c). This operator sets a pixel to 0 if all its 4-connected neighbors are 1, thus leaving only the boundary pixels on. It is worth noting that the watershed transform well separate lighter regions from darker ones in the original image. For this reason the sclera is marked as foreground in the binary mask, unlike the iris texture which belongs to the background. Pupil and iris correspond to a black concavity hollowed into the foreground region of the sclera, making the iris boundary to be part of the sclera contour. It is essential to isolate pixels belonging to the iris boundary from the rest of the sclera contour, in such a way that a circle fitting technique can be used to approximate it. Fig. **2**-c shows that contour direction changes very smoothly along the iris boundary, while a sharp bend in this line characterizes its end points. This peculiar behavior of the iris contour makes the curvature analysis particularly suitable for detecting sharp diverts in the edge image.

However, the aim of this processing step is only to cut edges at points with high curvature and the point-wise computation of the curvature of a generic shape entails a high computational cost, so we can rather estimate it by using such an heuristic criterion that would be easier to calculate. Edges are treated as connected components. For each connected component C a contour tracing algorithm is applied to sorts all points $P_k=(x_k,y_k)$ belonging to C in a list $L_C=\{P_1, P_2,\ldots, P_n\}$. According to this list, the contour C is linearly scanned point by point. For each P_k, a further point P_{k+t} is also considered, where $t=4\cdot\lfloor log_2(|C|)\rfloor$ is a parameter, whose value is proportional to the contour length. In particular, WISE considers the midpoint $M=(m_x,m_y)$ of the segment S ended by P_k and P_{k+t} and calculates the Euclidean distance $d_{k,t}(M, P_n)$ between M and P_n, where $n=\lfloor (k+t)/2\rfloor$. The values of $d_{k,t}(M, P_n)$ are divided by $d_{\max} = \arg\max_k(d_{k,t})$, to ensure they belong to the range [0,1]. The higher is $d_{k,t}(M, P_n)$, the larger is the value of the estimated curvature at point P_k. Given a fixed threshold σ (in this case σ =0.5), all points P_k for which the estimated curvature $d_{k,t}$ is larger than σ are deleted from L_C, so that a complex contour C is partitioned in a set of smoothed edges $E_1, E_2, \ldots, E_q$, as shown in Fig. **2**-c.

A connected component search is applied to the resulting edge image, so that all pixels belonging to the edge E_i are grouped in the same component Q_i (Fig. **2**–d), while only those containing a number of pixels greater than a threshold Th_Q are included in a unique list H (here $Th_Q=150$). Taubin's algorithm [16] is applied to each element Q_i in H to compute the corresponding circle R_i and its approximation error $e(Q_i, R_i)\in[0,1]$. The final list H_Q is obtained by pruning from H all components whose circles are not completely contained inside the image. In order to identify the iris, each circle R_i undergoes a voting procedure, according to a score value s_i, which is computed as follows:

$$s_i = e_h - e(Q_i, R_i),$$
$$\text{where } h = \arg\max_i (e(Q_i, R_i)) \tag{4}$$

The circle R_{max} with highest score s_{max} is considered as the circular shape which better approximates the iris (Fig. **2**–e).

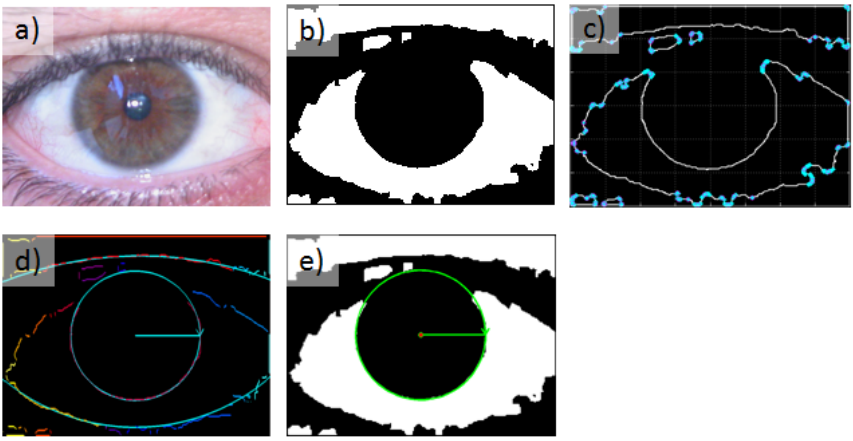


Fig. 2. A graphical representation of outputs produced by the intermediate steps of WISE: a) the original colour image, b) the binary image produced by the thresholded watershed, c) points where curvature is larger than a given threshold σ, d) circle fitting of the connected components, e) the circle with best rank

3 The Experimental Setup and Results

We evaluated the precision of iris detection and compared the results obtained by our WISE system with those provided by a system built from Masek's implementation of various iris-related algorithms, where segmentation is performed according to Wildes' method [17] as well as those reached by the IS_{IS} approach [4].

In order to assess the accuracy of the iris segmentation approaches we considered in this paper, a precise manual localization of the iris centre and radius was used as ground truth for each iris image. A manual annotating system was used, in which iris images were shown to a human operator who selected their centre (x, y) and radius (r) by clicking them with the mouse.

The different approaches were tested on UBIRIS v1 Session 2 database [10]. The UBIRIS v1s2 database is composed of 1877 images with a resolution of 800×600 pixels, which have been collected from 241 persons. The presented experimental results report location accuracy iris centre (x, y) and radius (r). Table 1 reports the relative precision of iris location by ISIS, Wildes, and WISE with respect to manual segmentation. Mean and variance of error are relative to the image size. Image coordinates are normalized according to the respective direction (x with 800, y with 600). Radiuses are normalized according to the horizontal direction. Experimental

results demonstrated that if the distances are greater than a 20 pixels threshold, the iris can't be detected because the eye is closed or the detection is wrong. The columns "no out." ("no outliers") report mean and variance computed by eliminating completely wrong locations (error greater than 20 pixels) or also UBIRIS closed eyes. Under the method's label we report the number of eliminated outliers.

Table 1. Location results for UBIRIS database

UBIRIS v1s2		Iris x		Iris y		Iris radius	
		all	**no out.**	**all**	**no out.**	**all**	**no out.**
ISIS	**mean**	2.4	0.3	4.3	1.6	2.2	0.8
out. = 12%	**std**	6.3	0.3	6.4	1.4	5.0	0.6
WILDES	**mean**	24.8	24.9	24.8	24.9	10.4	10.2
out. = 38%	**std**	7.0	2.4	7.9	3.7	1.7	1.7
WISE	**mean**	1.3	0.5	2.2	1.7	1.2	0.8
out. = 18%	**std**	4.2	0.8	3.7	1.9	2.3	1.1

A first observation is the slightly worst performances on the vertical coordinate of the pupil centre. This might be due to eyelids, which especially affect such dimension. When outliers are not dropped from whole set of iris images, WISE always outperforms the other methods. Otherwise, in the case in which outliers are not considered for evaluating the segmentation accuracy, WISE performance overcomes that of WILDES, being comparable with that of ISIS. Similarly, the number of outliers produced by WISE is comparable to that provided by ISIS, which is the smallest one.

It is worth to notice that even when images are affected by a wide number of problems due to occlusions or reflections, WISE still achieves acceptable performances, while those by the other methods dramatically decrease.

4 Conclusions

This paper has illustrated a new watershed based approach for iris segmentation on color images acquired in visible wavelength light. The technique has been designed to provide a more robust segmentation that is a crucial step when illumination changes significantly. The adopted watershed transform combines a classical iterative region growing and a novel merging technique specifically designed for the iris context. The experimental results demonstrate this approach to be effective with respect to location accuracy if compared with today's algorithms.

References

1. Daugman, J.G.: New methods in iris recognition. IEEE Transactions on Systems, Man, and Cybernetics - Part B: Cybernetics 37(5), 1167–1175 (2007)
2. Daugman, J.G.: How iris recognition works. IEEE Transactions on Circuits and Systems for Video Technology 14(1), 21–30 (2004)

3. Daugman, J.G.: How Iris Recognition Works. IEEE Trans. on CSVT 14(1), 21–30 (2004)
4. De Marsico, M., Nappi, M., Riccio, D.: IS_IS: Iris Segmentation for Identification Systems. In: Proc. of the International Conference on Pattern Recognition, pp. 2857–2860 (2010)
5. Donida Labati, R., Scotti, F.: Noisy iris segmentation with boundary regularization and reflections removal. Image and Vision Computing 28(2), 270–277 (2010)
6. Li, P., Liu, X., Xiao, L., Song, Q.: Robust and accurate iris segmentation in very noisy iris images. Image and Vision Computing 28(2), 246–253 (2010)
7. Meyer, F.: Color image segmentation. In: Proc. of the International Conference on Image Processing and its Applications, pp. 303–306 (1992)
8. Nguyen, V.H., Hakil, K.: A Novel Circle Detection Method for Iris Segmentation. In: Proc. of the Congress on Image and Signal Processing, vol. 3, pp. 620–624 (2008)
9. Phillips, P.J., Scruggs, T., O'Toole, A., Flynn, P.J., Bowyer, K.W., Schott, C., Sharpe, M.: FRVT 2006 and ICE 2006 Large-Scale (2006)
10. Proenca, H., Alexandre, L.A.: UBIRIS: A noisy iris image database. In: Proc. of the International Conference on Image Analysis and Processing, pp. 970–977 (2005)
11. Puhan, N.B., Sudha, N.: A novel iris database indexing method using the iris color. In: Proc. of the 3rd IEEE Conf. on Industrial Electronics and Applications, pp. 1886–1891 (2008)
12. Roerdink, J.B.T.M., Meijster, A.: The watershed transform: definitions, algorithms and parallelization strategies. Fundamenta Informaticae 41, 187–228 (2001)
13. Special Issue on the Segmentation of Visible Wavelength Iris Images Captured At-a-distance and On-the-move. Image and Vision Computing 28 (2010)
14. Special Issue on the Recognition of Visible Wavelength Iris Images Captured At-a-distance and On-the-move. Pattern Recognition Letters 33 (2012)
15. Tan, T., He, Z., Sun, Z.: Efficient and robust segmentation of noisy iris images for non-cooperative iris recognition. Image and Vision Computing 28(2), 223–230 (2010)
16. Taubin, G.: Estimation of Planar Curves, Surfaces And Nonplanar Space Curves Defined By Implicit Equations, With Applications To Edge And Range Image Segmentation. IEEE Trans. on PAMI 13, 1115–1138 (1991)
17. Wildes, R.: Iris recognition: an emerging biometric technology. Proceedings of the IEEE 85(9), 1348–1363 (1997)

A Multiscale Erosion Operator for Discriminating Ground Points in LiDAR Point Clouds

José Luis Silván-Cárdenas*

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

Abstract. Digital terrain models (DTM) are basic products required for a number of applications and decision making processes. Nowadays, high spatial-resolution DTMs are primarily produced through airborne laser scanners (ALS). However, the ALS does not directly deliver DTMs but a dense cloud of 3-d points that embeds both terrain elevation and height of natural and human-made features. Such a point cloud is generally rasterized and referred to as the digital surface model (DSM). The discrimination of aboveground objects from terrain, also termed ground filtering, is a basic processing step that has proved especially difficult for large areas of complex terrain characteristics. This paper presents the development of a multiscale erosion operator for removing aboveground features in the DSM, thus producing a surface that is close to the DTM. Such an approximation was used to separate ground from non-ground points in the original point-cloud and the discrimination accuracy was assessed using publicly available data. Results indicated an improvement over a previously published method.

Keywords: Remote sensing, LiDAR, Ground filtering, multiscale Hermite transform.

1 Introduction

The small-footprint, discrete-return light detection and ranging system (LiDAR) has become one of the most important means to produce high-resolution DTM data. This has been due in part to a number of advantages over competing aerial photogrammetric techniques, such as independence of sun light, higher vertical accuracy, less missing data by occlusion, low redundancy and because it does not rely on existence of textured surfaces and discontinuities for a successful point matching [1]. The discrete return LiDAR system is based on the accurate measurement of the elapsed time between emitted and backscattered laser pulses. The emitted pulse is typically short-time and unimodal, whereas

* The author thanks the ISPRS for providing the LiDAR data used in this study.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 213–223, 2013.

the backscattered may spread over longer times exhibiting multiple modes called returns. Returns are associated to distinct layers the laser interacted with. For each return, the 3-d position is determined and its intensity recorded. Hence the raw LiDAR data consists of a dense cloud of 3-d points with associated return intensity. Each point provides the location where the laser hit the Earth's surface during the scanning process, whereas the intensity is a digital representation of the fraction of pulse energy reflected at that location.

The z-coordinate of points correspond to terrain elevation with respect to a horizontal datum, typically the mean see level, plus the height of non-terrain features in some instances. In order to produce a DTM by interpolation of ground points, a discrimination of ground from non-ground points must be carried out first. This discrimination process is referred to as ground filtering and is generally considered a preprocessing for generating not only the DTM, but also the height of non-ground components [2, 1]. Further classification of non-terrain points into meaningful features, such as trees, buildings, roads, and so on, is also common task that is needed for detailed analysis and quantification of landscape characteristics.

Methods to ground filtering can be grouped into two big categories, namely point-based and raster-based methods. Methods in the first category classify directly the point cloud whereas methods in the second category first rasterize the point cloud onto a regular grid through an interpolation method. Each approach has advantages and disadvantages. In general, rasterizing the data first allows to take advantage of digital image processing algorithms which run much faster than point-based operations, whereas point-based processing tends to be more accurate [2]. A recent review of methods and critical issues of the ground filtering problem is provided in [3]. Interestingly, the need for processing at multiple scales/resolutions has been increasingly recognized by several studies [4–6]. This has let to the adoption of multiresolution image decomposition techniques in raster-based approaches by some authors [4, 7]. These algorithms tend to be very efficient, less sensitive to parameter selection, less sensitive to point density and are able to remove large non-ground features, such as bridges and complex buildings.

The method presented in this paper falls in the raster-based category and is based on an wavelet-like transform termed the multiscale hermite transform (MHT), which is an overcomplete signal decomposition based on scaled and rotated Gaussian derivatives [8]. The original filtering method was introduced in [7] and, based on an extensive accuracy assessment, was ranked among the top three out of nine algorithms tested. Furthermore, its performance has been demonstrated in several real applications [9–11]. The rationale of the method is as follows. A DSM is first generated through a point-to-raster conversion of the point cloud. Then the DSM is decomposed using the MHT expansion with transform coefficients processed through the so-called erosion operator, so that when applying the inverse transform, the re-synthesized surface corresponds to an eroded version of the DSM. The resulting surface is then used to generate a ground/non-ground mask by comparing the original DSM with its eroded

version. The overall process is illustrated in Figure 1, but major improvement presented in this paper corresponds to the "Multiscale Erosion Operation" step. A more complete description of the overall algorithm is to appear in [12].

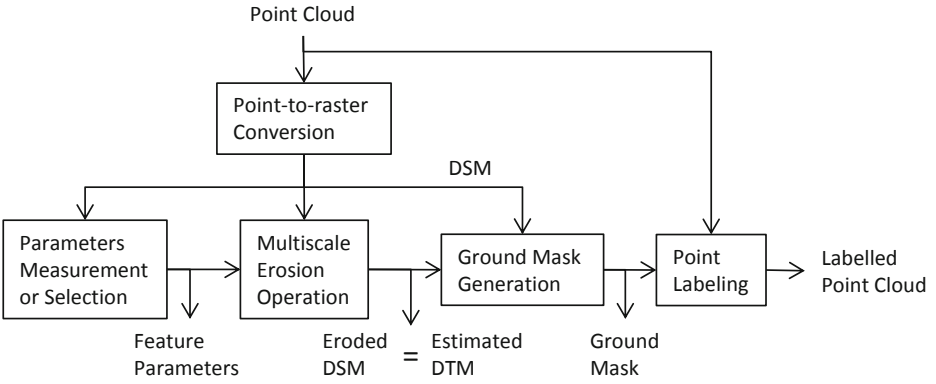


Fig. 1. Flow diagram of ground filtering method. Adapted from [12]

The rest of the paper is organized as follows. Section 2 provides a summary of the theoretical background around the MHT. Then, in Section 3 the multiscale erosion operator is defined and illustrated. Next, in Section 4 some filtering tests are described and results of the accuracy assessment presented. Conclusions drawn from the study are presented in Section 5.

2 Background

The MHT can be defined for one, two and higher dimensions and for both continuous and discrete signals, here we summarize the theory for two-dimensional, discrete signals, and refer the interested reader to the original source [8] for other forms. For simplicity, exposition starts with the single-scale case and then extend the result to multiple scales.

2.1 Single-Scale Discrete Hermite Transform

The discrete Hermite transform (DHT) of a signal $z(x, y)$ defined on a discrete domain can be expressed as a convolution of the input signal with a bank of 1-d filters along each dimension and followed by a subsampling with a rate factor of two. This is expressed in mathematical terms as follows:

$$z_{n,m}(p, q) = \sum_{x,y} z(x, y) b_n(x - 2p) b_m(y - 2q) \tag{1}$$

where

$$b_n(x) = 2^{-N}\sqrt{C_N^n}\sum_{j=0}^{n}(-1)^{n-j}C_n^j C_{N-n}^{x+N/2+j-n} \tag{2}$$

for $x = -N/2, \ldots, N/2$ and by $b_n(x) = 0$, for all other values of x, represent the bank of 1-d filters for $n = 0, \ldots, N$. The set of such filters is referred to as the binomial family. Since the binomial family is the discrete counterpart of Gaussian derivatives, a transform coefficient $z_{n,m}(p, q)$ approximates (up to a normalizing factor) the partial derivative of order n with respect to x and of order m with respect to y of a Gaussian-smoothed version of the input signal z at the location (p, q). The degree of smoothness of the Gaussian kernel is controlled through the scale parameter $s = N/4$.

Conversely, the full set of transform coefficients allows recovering the input signal through the inversion procedure summarized in the following equation:

$$z(x, y) = \sum_{n,m=0}^{N}\sum_{p,q} z_{n,m}(p, q)\tilde{b}_n(2p - x)\tilde{b}_m(2q - y) \tag{3}$$

where the interpolation filters $\tilde{b}(x) = 2b(-x)$, for $n = 0, 2, \ldots, N$, are binomial filters reflected around the origin.

In general, the DHT expansion compacts most of the signal information in the first few coefficients. Hence, a near perfect reconstruction can be obtained with a truncated expansion. Further compaction of the information can be achieved through local orientation of the coordinate axis along the strongest signal variation. More specifically, the expansion coefficients can be defined 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). Since different orientation angles are chosen for different sampling points, this operation is referred to a local spatial rotation. Figure 2 shows an example of original versus rotated coefficients.

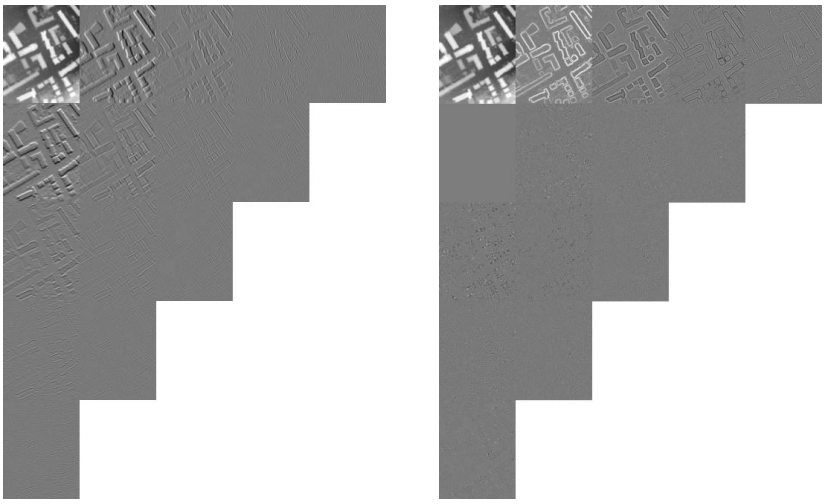
2.2 Local Spatial Rotation

Let $z_{n,m}^{(\theta)}$ denote the rotated coefficients at a generic sampling location, where location is purposely omitted in the notation for simplicity. This can be computed as linear combinations of the original (un-rotated) coefficients through the following mathematical relation:

$$\frac{z_{n-m,m}^{(\theta)}}{\sqrt{C_n^m}} = \sum_{k=0}^{n} a_{m,k}^{(\theta,n)}\frac{z_{k,n-k}}{\sqrt{C_n^k}} \tag{4}$$

The recovery of the original coefficients from the rotated ones is performed through rotation by a negative angle. The angle functions $a_{m,k}^{(\theta,n)}$ correspond to the generalized binomial filters (GBF), a family of discrete sequences with parameters n and θ, which are given by

$$a_{m,k}^{(\theta,n)} = s^k c^{-k}\Delta^m\{C_{n-m}^{k-m}c^{2k-m}s^{n-2k+m}\} \tag{5}$$



(a) (b)

Fig. 2. Example of DHT coefficients up to fourth order (a) and corresponding rotated coefficients (b). In both cases, the order of derivation with respect to x- and y-coordinates grows from left to right and from top to bottom, respectively.

for $m, k = 0, \ldots, n$, and $c = \cos\theta$ and $s = \sin\theta$. The first few GBFs can be expressed using the matrix notation $\mathbf{A}^{(\theta,n)} = [a_{m,k}^{(\theta,n)}]_{m,k=0,\ldots,n}$, so that

$$\begin{pmatrix} s & c \\ c & -s \end{pmatrix}, \begin{pmatrix} s^2 & 2sc & c^2 \\ sc & c^2 - s^2 & -sc \\ c^2 & -2sc & s^2 \end{pmatrix}, \begin{pmatrix} s^3 & 3s^2c & 3sc^2 & c^3 \\ s^2c & -s^3 + 2sc^2 & -2s^2c + c^3 & -sc^2 \\ sc^2 & -2s^2c + c^3 & s^3 - 2sc^2 & s^2c \\ c^3 & -3sc^2 & 3s^2c & -s^3 \end{pmatrix}$$

correspond to the matrices for $n = 1, 2$ and 3, respectively.

In all the examples presented here, the rotation was set to the orientation of the local gradient, that is:

$$\theta = \arctan \frac{z_{0,1}}{z_{1,0}} \tag{6}$$

This selection of the angle makes the rotated coefficients $z_{0,1}^{(\theta)} = 0$, and

$$z_{1,0}^{(\theta)} = \sqrt{z_{1,0}^2 + z_{0,1}^2} \tag{7}$$

The latter corresponds (up to a constant factor) to the magnitude of the local gradient. Other angles may be computed that involve higher order coefficients, however the gradient angle generally achieves higher energy compaction of the signal along the first coordinate. This is the case if the input signal embeds strongly oriented features, such as the edges of buildings in a DSM (see Figure 2).

2.3 Multiscale Discrete Hermite Transform

The multiscale discrete Hermite transform (MDHT) is implemented by recursively replacing the zero-order (low-pass) coefficient of previous level, $z_{0,0}^{(k-1)}(x,y)$, by its DHT expansion times a scaling factor, thus yielding a wavelet-like pyramidal decomposition.

Mathematically, the MDHT with $K+1$ pyramid levels is expressed through:

$$z_{n,m}^{(0)}(p,q) = \sum_{x,y} z(x,y) b_n(x-2p) b_m(y-2q), \quad \text{for } k=0 \tag{8}$$

and

$$z_{n,m}^{(k)}(p,q) = \sum_{x,y} z_{0,0}^{(k-1)}(x,y) c_n(x-2p) c_m(y-2q), \quad \text{for } k=1,\ldots,K \tag{9}$$

for $n,m=0,\ldots,N$ and $c_n(x) = (\sqrt{3}/2)^n b_n(x)$. Conversely, the reconstruction of the original signal is carried out through successive reconstruction of low-pass coefficients from the coefficients of the upper pyramid levels. This is expressed as:

$$z_{0,0}^{(k-1)}(p,q) = \sum_{n,m=0}^{6} \sum_{i,j} z_{n,m}^{(k)}(i,j) \tilde{c}_n(2i-p) \tilde{c}_m(2j-q), \quad \text{for } k=K,\ldots,1 \tag{10}$$

and

$$z(x,y) = \sum_{n,m=0}^{8} \sum_{p,q} z_{n,m}^{(0)}(p,q) \tilde{b}_n(2p-x) \tilde{b}_m(2q-y)) \tag{11}$$

with $\tilde{c}_n(x) = (\sqrt{3}/2)^{-n} \tilde{b}_n(x)$.

It should be noted that the replaced low-pass coefficients are not part of the MDHT expansion, but only used for the computation of the next coarser level. In contrast, since the coarsest low-pass coefficient is not replaced, it has to be considered part of the MDHT expansion. Also, the length of binomial filters of the first pyramid level is $N=8$, whereas that for higher levels is $N=6$. These filter length values ensure that the low-pass coefficients comprise a scale-space representation in normalized spatial coordinates for a discrete scale sequence. Such a representation is equivalent to filtering the original signal with a sequence of low-pass binomial filters of length $N_k = 8 \times 4^k$, for $k=0,\ldots,K$, and then resample the output by a factor 2^k. Because of this, the MDHT can also be interpreted as an expansion of the Gaussian pyramid in terms of its derivatives. Furthermore, the MDHT coefficients can also be rotated using the same formulas as stated previously. In this case, the notation is extended to $z_{n,m}^{(\theta,k)}$, which accounts for the rotation angle.

3 The Multiscale Erosion Operator

The multiscale erosion operator is a processing of the coefficients that allows to remove aboveground features in the transform domain. This can be seen as

a local spatial shifting of the signal, a process that is performed only around detected ground/non-ground transition pixels, so that the portion of the signal in the vicinity of a ground/non-ground transition is replaced by a portion of the signal located along neighbor ground points.

The local spatial shifting is based on a property of the binomial filters stating that the members of a binomial filter family can reconstruct the members of a binomial filter family at a shifted location and decreased filter length. Otherwise stated:

$$\frac{b_n(x - M/2; N - M)}{\sqrt{C_{N-M}^n}} = \sum_{m=0}^{M} (-1)^m C_M^m \frac{b_{n+m}(x; N)}{\sqrt{C_N^{n+m}})} \tag{12}$$

for $M < N$, where shifting occurs both in scale (N) and space (x). Since the DHT is a linear transform, a similar relation can be obtained for the DHT coefficients in 1-d. For 2-d and higher-dimensional signals, the spatial shifting operation can be combined with the rotation operation, thus resulting in a directional spatial shifting. In particular, for 2-d signals the spatial shifting along a direction defined by an angle θ is expressed through:

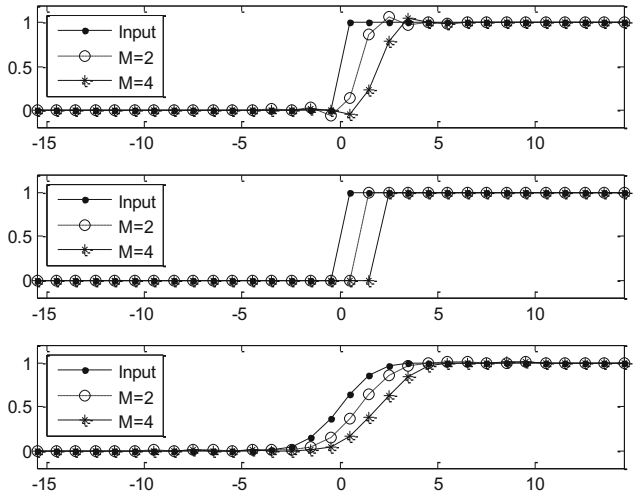


Fig. 3. These plots illustrate the effect of the shifting parameter (M) in reconstructed surface profiles (circles and stars). The input profile (dots) corresponds to a sharp step for the first and second case and to a smooth step for third case. The DHT of the input was computed with $N = 8$ and the transform coefficients were processed with a scale-space shifting of Eq. (13) for $M = 2$ and $M = 4$. Then, the profiles were reconstructed from processed coefficients with the inverse DHT using the same filter length ($N = 8$) for the first and third cases, but a decrease filter length ($N - M$) for the second case.

$$\frac{z_{n,l}^{(\theta,N-M)}(p-M/2,q)}{\sqrt{C_{N-M}^{n}}} = \sum_{m=0}^{M}(-1)^m C_M^m \frac{z_{n+m,l}^{(\theta,N)}(p,q)}{\sqrt{C_N^{n+m}}} \tag{13}$$

for $n = 0, \ldots, N-M$ and $l = 0, \ldots, N$, where now the filter length is included in the notation of the coefficients to make explicit the shifting in scale as well[1]. Figure 3 illustrates the effect of the shifting parameter M on a reconstructed profile from its shifted transform coefficients. The shifting in scale can be ignored only if the signal is smooth enough (bottom panel). However, for sharp transitions the inverse transform requires to account for the scale shifting (middle panel), or otherwise signal distortions may occur (upper panel) and this is accentuated with increasing values of M.

Of special interest is the case for $N = 8, M = 2$ and $n = l = 0$, for which the above equation reduces to:

$$z_{0,0}^{(\theta,6)}(p-1,q) = z_{0,0}^{(\theta,8)}(p,q) - \frac{z_{1,0}^{(\theta,8)}(p,q)}{\sqrt{2}} + \frac{z_{2,0}^{(\theta,8)}(p,q)}{\sqrt{28}} \tag{14}$$

which would correspond to the erosion operator developed in [7] if the scale shifting and the third term in the left-hand side were neglected. In that study, such an erosion operator was derived from an explicit model of a 1-d surface profile in the continuous domain, and then extrapolated for the 2-d discrete case. In contrast, the formula of Equation (13) represents a more general way as it allows shifting the higher order coefficients as well, instead of simply setting them to zeroes as in the previous work.

Additionally, in the original formulation, the terrain elevation underneath non-ground features was assumed flat. This was convenient because the erosion operator is essentially a local spatial shifting. In order to account for higher order variations of terrain elevations at those sites, a truncated Taylor expansion is also proposed in this study. Specifically, the following approximation was used to estimate terrain elevations at transition points:

$$z_{n,m}^{(\theta,N)}(p,q) \approx z_{n,m}^{(\theta,N)}(p-M/2,q) + c_1 z_{n+1,m}^{(\theta,N)}(p-M/2,q) \tag{15}$$

with the constant $c_1 = (M/4)\sqrt{n+3}$, which accounts for the relation between the transform coefficients and the signal derivatives. Hence, the new erosion operator is implemented by plugging the local spatial shifting of Equation (13) into Equation (15).

In all tests presented below the shifting parameter was set to $M = 2$ and the Equation (15) applied to the coefficients $z_{0,0}^{(\theta,k)}$ and $z_{1,0}^{(\theta,k)}$ at detected ground/non-ground transitions, while all other coefficients are set to zero. Such a processing was inserted in the computation of the MDHT, so that erosion is carried on the coarser layers. The processed coefficients are then used to synthesize a surface

[1] It should be noted that the scale rather than a scale index is used here.

that approximates the DTM, which in turn was used to classify the original point cloud into ground and non-ground points. Further details on both the detection of ground/non-ground transition points, as well as on the insertion of the erosion operator in the MDHT, are provided in [7].

4 Ground Filtering Tests

Both the previously developed and the new erosion operators were applied to the ISPRS datasets for two sites (`www.commission3.isprs.org/wg3/`). The first site corresponds to an urban area (CSite2) exhibiting large and irregular shaped buildings, as well as a road with bridge and small tunnel. The second site corresponds to a rural area (FSite5) with vegetation on steep slopes, quarry, vegetation on river banks and gaps. The original point cloud included elevation and intensity from first and last returns; however, only first return was used in the tests as it was generally the cleanest measurement.

Table 1. Selected filter parameters for each test site

Parameter	CSite2	Fsite5
Maximum Feature Width [m]	100	50
Maximum Terrain Elevation Difference [m]	30	90
Maximum Terrain Slope [Deg]	25	55
Maximum Elevation Tolerance [m]	0.25	0.25
Cell Size [m]	1	2

The parameters used for the filtering of each dataset are provided in Table 1. Cell sizes for point-to-raster conversions were roughly equivalent to the original average point spacing. Further details on the interpolation of the data are available in the reference [7]. Other parameters were interactively measured from visualizations of the DSM as suggested in [12]. Nonetheless, although a very steep slope of 88 degrees was measured for the forested area, a value of 55 degrees was used instead in order to avoid leaving most non-ground features unfiltered.

The accuracy assessment of the filtering results was carried out for up to four test insets per site, for which classification was available. Ground points from the reference data were interpolated to generate a reference DTM that served to compute the root mean square error (RMSE) of the estimated DTM (eroded DSM) (Table 2(a)). The latter was also used to classify the original point cloud and the overall classification accuracy (OCA) was computed for each sample set (Table 2(b)).

In general, the new method showed lower RMSE for both sites, thus indicating a superior method for estimating the DTM. In terms of classification accuracy, the new method was still better than the old one in the average, yet there were a few cases were the old method remained comparable or even superior to the new method. In either case, results for the urban site showed significant commission

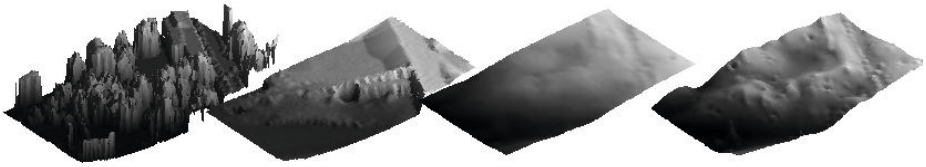


Fig. 4. This figure illustrates the surfaces derived for one of the test samples (samp24). From left to right: Input DSM, reference DTM, synthesized DTM with the old erosion method and synthesized DTM with the new erosion method.

Table 2. Root mean square error in meters of estimated DTM (a) and overall accuracy of ground/non-ground classification (b) for each test site and method

(a)

Sample Set	CSite2		Fsite5	
	Old	New	Old	New
samp*1	4.12	4.07	4.44	3.96
samp*2	6.85	4.26	9.03	2.30
samp*3	10.41	8.86	7.03	5.11
samp*4	6.32	3.77	4.62	4.00
average	6.92	5.74	6.28	3.84

(b)

Sample Set	CSite2		Fsite5	
	Old	New	Old	New
samp*1	70	68	77	75
samp*2	77	81	69	86
samp*3	78	84	62	82
samp*4	65	74	85	79
average	73	77	73	81

errors along the eastern side due to the edge effect of the multiscale filtering and the terrain elevation gradient towards the eastern direction, yet this was less significant than in the tests reported in previous work. The reason for this was a change from symmetric reflection to an anti-symmetric reflection of the boundary cells. According to this condition, the extension of a sloppy terrain near the edge will maintain its slope rather than change it with a symmetric reflection. For the forested site, the major problem was the omission errors along the southern side, where a sharp terrain shape was present.

5 Conclusions

This paper presented a filtering method based on a multiscale signal decomposition termed the multiscale Hermite transform, which has been formulated in the context of the scale-space theory for signal processing [8]. As the overall filtering method had been extensively tested and applied elsewhere [7, 9–11], the main focus of this work was on an extension of the erosion operator used in that method. Such an extension used a truncated Taylor expansion and a local spatial shifting operator in the transform domain for predicting terrain elevation underneath non-ground features, such as buildings and trees. When compared with the original method, it was found that synthesized DTM was more accurate than for the previous method, which also led to a better discrimination of ground points from the original point cloud. Although the synthesized DTM relied on

coefficients of up to first order only, there were involved coefficients of up to fourth order through the scale-space shifting operation, which was a major reason for the reconstruction of more details. Future work should develop practical ways for parameter selection and investigate the effects of incorporating even higher order coefficients.

References

1. Pfeifer, N., Mandlburger, G.: 11. In: LiDAR Data Filtering and DTM Generation, pp. 307–333. CRC Press, Boca Raton (2009)
2. Axelson, P.: Processing of laser scanner data - algorithms and applications. ISPRS Journal of Photogrametry and Remote Sensing 54(2-3), 138–147 (1999)
3. Meng, X., Currit, N., Zhao, K.: Ground filtering algorithms for airborne lidar data: A review of critical issues. Remote Sensing 2(3), 833–860 (2010)
4. Thuy, T., Tokunaga, M.: Filtering airborne laser scanner data: A wavelet-based clustering method. Photogrammetric Engineering and Remote Sensing 70(11), 1267–1274 (2004)
5. Chen, Q., Gong, P., Baldocchi, D., Xie, G.: Filtering airborne laser scanning data with morphological methods. Photogrammetric Engineering & Remote Sensing 73(2), 175–185 (2007)
6. Evans, J.S., Hudak, A.T.: A multiscale curvature algorithm for classifying discrete return lidar in forested environments. IEEE Transactions on Geoscience and Remote Sensing 45(4), 1029–1038 (2007)
7. Silván-Cárdenas, J., Wang, L.: A multi-resolution approach for filtering LiDAR altimetry data. ISPRS Journal of Photogrammetry and Remote Sensing 61(1), 11–22 (2006)
8. Silván-Cárdenas, J., Escalante-Ramírez, B.: The multiscale Hermite transform for local orientation analysis. IEEE Transactions on Image Processing 15(5), 1236–1253 (2006)
9. Silván-Cárdenas, J., Wang, C.W.L., Rogerson, P., 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)
10. Silván-Cárdenas, J.L., Wang, L.: Extraction of buildings footprint from lidar altimetry data with the hermite transform. In: Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Ben-Youssef Brants, C., Hancock, E.R. (eds.) MCPR 2011. LNCS, vol. 6718, pp. 314–321. Springer, Heidelberg (2011)
11. Silván-Cárdenas, J.L.: A segmentation method for tree crown detection and modelling from lidar measurements. In: Carrasco-Ochoa, J.A., Martínez-Trinidad, J.F., Olvera López, J.A., Boyer, K.L. (eds.) MCPR 2012. LNCS, vol. 7329, pp. 65–74. Springer, Heidelberg (2012)
12. Silván-Cárdenas, J., Wang, L.: A multiscale approach for ground filtering from LiDAR altimetry measurements. In: Scale Issues in Remote Sensing. John Wiley and Sons (to appear, 2013)

Enhancing Biomedical Images Using the UFIR Filters with Recursive Responses

Luis J. Morales-Mendoza[1], Rene F. Vázquez-Bautista[1], Mario González-Lee[1], M. Ibarra-Manzano[2], Y. Shmaliy[2], and J. Martínez-Castillo[1]

[1] FIEC – Universidad Veracruzana,
Av. Venustiano Carranza s/n, Col. Revolución, CP. 93390, Poza Rica Ver.
{javmorales,favazquez,mgonzalez01,jaimartinez}@uv.mx
[2] DICIS – Universidad de Guanajuato
Ctra. Salamanca – Valle Km 3.5 + 1.8, Comunidad Palo Blanco, CP. 36855, Salamanca Gto.
{shmaliy,mibarra,ibarram}@ugto.mx

Abstract. In this paper we present a novel computational scheme to determine the impulse response of the UFIR filters. A recursive form of impulse response is developed using the theory of the discrete orthogonal polynomials. An example of an enhanced of medical image is considered to compare its performance versus the matrix formulation of the impulse response UFIR filters. Finally, some quantitative and qualitative evaluations are carried out to verify its efficiency based on RMSE analysis.

Keywords: UFIR filters, biomedical images, discrete orthogonal polynomials, recurrent relation.

1 Introduction

Biomedical imaging is a powerful tool in many medical procedures, such as detection of abnormal mass, diagnosis of some diseases, proper treatment for illnesses, among others. Nowadays, exist a variety of medical imaging techniques, such as: X-ray, Magnetic Resonance Imagery (MRI), Positron Emission Tomography (PET), and Ultrasound images [1, 2]. These images are classified in two categories: images formed with electromagnetic waves and images formed from acoustic waves. The first, are formed with particles provided by a particles beam so that the penetration in tissue and the organs is used for forming images. On the other hand, the second category to group images formed with the scatter waves that return from both the tissue and organs, respectively. Particular features are identified in such images for example: attenuation, reflection, dispersion, and impedance [3, 4]. However, these features are the result of the change in density between different tissues. In addition, the noise is an important issue in image formation and the most important types of noise in biomedical imaging field are speckle and Gaussian noise.

The problem of saving sharp edges whilst enhancing the image corrupted by either speckle or Gaussian noise is typical in biomedical image processing [2]. An overall

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 224–231, 2013.

overview of nonlinear filtering has been given in [5] along with the important modifications for a large class of nonlinear filters employing order statistics. The algorithm problems for the FIR filter design have been discussed in [6]. In [7] the finite impulse response (FIR) median hybrid filters (MHF) strategy has been proposed with applications to image processing.

In the other hand, the family of Unbiased Finite Impulse Response (UFIR) filter is relatively new in signal processing applications [8]. This digital FIR filter was developed by Shmaliy in order to reduce the Time Interval Error (TIE) in Global Position System (GPS) signals. Later, in [9-11] a modification in the impulse response of the UFIR filters is introduced, this new approach makes use of the size of the step (p) of process which is very important, this development presented the next three processes: Filtering, when, $p = 0$, Prediction when, $p > 0$, and Smoothing when, $p < 0$. Some applications of UFIR [12, 13] and p-UFIR [14-16] filters in biomedical image processing were developed by Morales-Mendoza, for ultrasound images applications. Furthermore, in [17] a ramp UFIR filter was used in the new computational scheme for synthetic image processing also targeted the ultrasound image processing [18]. Finally, a recursive scheme for computing the UFIR filters impulse response was proposed in [19, 20].

In this paper, we present an application of UFIR filters with recursive response to preserve sharp edges with a simultaneous enhancing of biomedical images. The rest of the paper is organized as follows. In Sect. 2, we derive the polynomial image model. In Sect 3, we briefly discuss the derivation of the impulse responses of the filters UFIR using the matrix method approach. The orthogonality UFIR functions and the development of recursive impulse response of UFIR filters are discussed in detail in Sect. 4. An example of biomedical image processing is given in Sect. 5 and concluding remarks are drawn in Section 6.

2 Polynomial Image Model

A two-dimensional image is often represented as a $k_c \times k_r$ matrix, $\mathbf{M} = \{\mu_{i,j}\}$. In order to perform two dimensional filtering, this matrix can be written in the form of a row-ordered vector (known as lexicographic form) or a column-ordered vector, respectively

$$\mathbf{y}_r = \left[\mu_{1,1} \cdots \mu_{1,k_r} \cdots \mu_{k_c,1} \cdots \mu_{k_c,k_r}\right]^T, \tag{1}$$

$$\mathbf{y}_c = \left[\mu_{1,1} \cdots \mu_{k_c,1} \cdots \mu_{1,k_r} \cdots \mu_{k_c,k_r}\right]^T. \tag{2}$$

The filtering procedure is then often applied twice, first to (1) and then to (2), or vice versa.

If a two-dimensional image is accurately represented using both (1) and (2), then each of the vectors may also considered as a deterministic one-dimensional signal, y. The unbiased FIR filter estimate is provided by the discrete convolution as follows

$$\hat{s}(t) = h_n(t, N) \otimes y(t) \tag{3}$$

where, $h_n(t, N)$ is the filter's impulse response with order approximation n, the constant N is the averaging boundary of the samples and $s(t)$ is the measurement signal (lexicographic image). It has shown in [8] it is possible to compute an unbiased estimation of $s(t)$ at time t from $y(t)$ by using the convolution defined in (3). The UFIR filter of n-degree, $n \in [0, K-1]$, with impulse response function $h_n(t, N)$ and monic polynomial is defined as follows

$$h_n(t,N) = a_{0n}(N) + a_{1n}(N)t + a_{2n}(N)t^2 + \cdots + a_{nn}(N)t^n = \sum_{j=0}^{n} a_{jn}(N)t^j . \tag{4}$$

3 Matrix Model of the Impulse Response UFIR Filter

In this Section, we derive the general mathematical model for signals and images based on (4). First, we substitute t by discrete variable x so we have a polynomial that exist from 0 to $N-1$ with the following fundamentals properties [8, 10],

- Unit area

$$\sum_{x=0}^{N-1} h_n(x,N) = 1 . \tag{5}$$

- Zero Moments

$$\sum_{x=0}^{N-1} x^u h_n(x,N) = 0 , \qquad 1 \le u \le n. \tag{6}$$

- Finite norm (energy)

$$\sum_{x=0}^{N-1} h_n^2(x,N) = a_{0n} < \infty . \tag{7}$$

In order to compute the coefficients of (4), lets first recall the properties described previously in (5), (6) and (7), respectively. Thus, the coefficients are defined as follows

$$a_{jn}(N) = (-1)^j \frac{M_{(j+1)1}(N)}{|\mathbf{D}(N)|} , \tag{8}$$

where, $|.|$ is the determinant of $\mathbf{D}(N)$ and $M_{(j+1)1}(N)$ is the minor of the Hankel matrix $\mathbf{H}_n(N) \in \Re^{(n+1)\times(n+1)}$,

$$\mathbf{D}(N) \equiv \mathbf{H}_n(N) = \begin{bmatrix} c_0(N) & c_1(N) & \cdots & c_n(N) \\ c_1(N) & c_2(N) & \cdots & c_{n+1}(N) \\ \vdots & \vdots & \ddots & \vdots \\ c_n(N) & c_{n+1}(N) & \cdots & c_{2n}(N) \end{bmatrix} , \tag{9}$$

each component in (9) can be found using the recursive relation of the Bernoulli polynomials $B_n(x)$ established in [8, 10]. Note that, so far, the following low-degree polynomials $h_n(x, N)$ were found and investigated so far

$$h_0(x,N)=\frac{1}{N} \tag{10}$$

$$h_1(x,N)=\frac{2(2N-1)-6x}{N(N+1)} \tag{11}$$

$$h_2(x,N)=\frac{3(3N^2-3N+2)-18(2N-1)x+30x^2}{N(N+1)(N+2)} \tag{12}$$

$$h_3(x,N)=\frac{8(2N^3-3N^2+7N-3)-20(6N^2-6N+5)x+120(2N-1)x^2-140x^3}{N(N+1)(N+2)(N+3)} \tag{13}$$

4 Recurrent Relation of the Impulse Response UFIR Filter

4.1 Orthogonality of the UFIR Functions

By definition, a set of orthogonal polynomials is a special system of polynomials $\{P_m(x)\}$, $m = 0, 1, 2\ldots$ that are orthogonal with respect to some weight function $\rho(x)$ on an interval $[a, b]$. Given the n-degree polynomials functions $h_n(x, n)$ defined by (4) and (8) from 0 to $N - 1$ with the properties (5)-(7), a class of functions $\{h_n(x, N)\}$ is orthogonal on $x \in [0, N - 1]$, satisfying

$$\sum_{x=0}^{N-1} h_k(x,N)h_n(x,N)\rho(x,N)=d_n^2(N)\delta_{kn}, \tag{14}$$

where, the subscripts k and $n \in [0, K - 1]$, δ_{kn} is Kronecker delta, $d_n^2(N)$ is defined as the square of the weighted norm of $h_n(x, N)$ and $\rho(x, N)$ is the ramp probability density function defined earlier, both functions are defined in [19, 21] as follows

$$\rho(x,N)=\frac{2x}{N(N-1)}\geq 0 \tag{15}$$

and $d_n^2(N)$ is defined as the weighted norm of $h_n(x, N)$.

4.2 Recurrent Equation of UFIR Polynomials

It is known that any real orthogonal polynomials $P_n(x)$ such as (4) satisfy the Favard's theorem they are a sequence of polynomials satisfying a suitable three-term recurrence relation. The recursive relationship for determining the impulse response of UFIR filters [18-21] is defined as

$$h_n(x,N) = 2\frac{n^2(2N-1) - x(4n^2-1)}{n(2n-1)(N+n)} h_{n-1}(x,N) - \frac{(2n+1)(N-n)}{(2n-1)(N+n)} h_{n-2}(x,N) \tag{16}$$

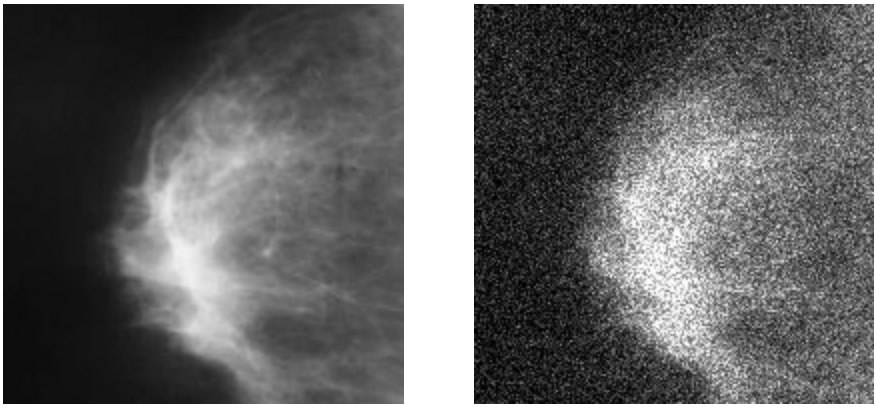
where, the above ratio, are constrained to the following initial conditions

$$h_{-1}(x,N) = 0\,, \qquad \text{and} \qquad h_0(x,N) = 1/N \tag{17}$$

using the relation (16) with the initial conditions defined in (17) yields the impulse response defined in (11) to (13), respectively.

5 Simulations

For further investigation, we chose a biomedical image of 200 × 200 pixels to grayscale showed in Fig. 1a. The image was contaminated with both additive white Gaussian and speckle noise as shown in Fig. 1b. The simulation conditions were set as follows: The average of horizon is $N = 11$ samples and the noises added to image are the white and speckled noise with a variance of 0.1 each one.



a) b)

Fig. 1. a) Original biomedical Image, b) Noisy Image with $\sigma^2 = 0.1$

The computational evaluation is carried out using a biomedical imaging. Image improvement is obtained using low-order polynomial approximation of the impulse response of the filter. Fig. 2 shows the enhanced images using different impulse responses. The evaluation in terms of RMSE metric is shown in Table 1.

The computational evaluation is carried out in a personal computer that used an Intel® CORE 2 DUO processor with 2.4GHz of speed and 2GB of RAM. In Fig. 3, the time requested to pixel processing using the matrix and recursive forms of the impulse response of UFIR filter is shown.

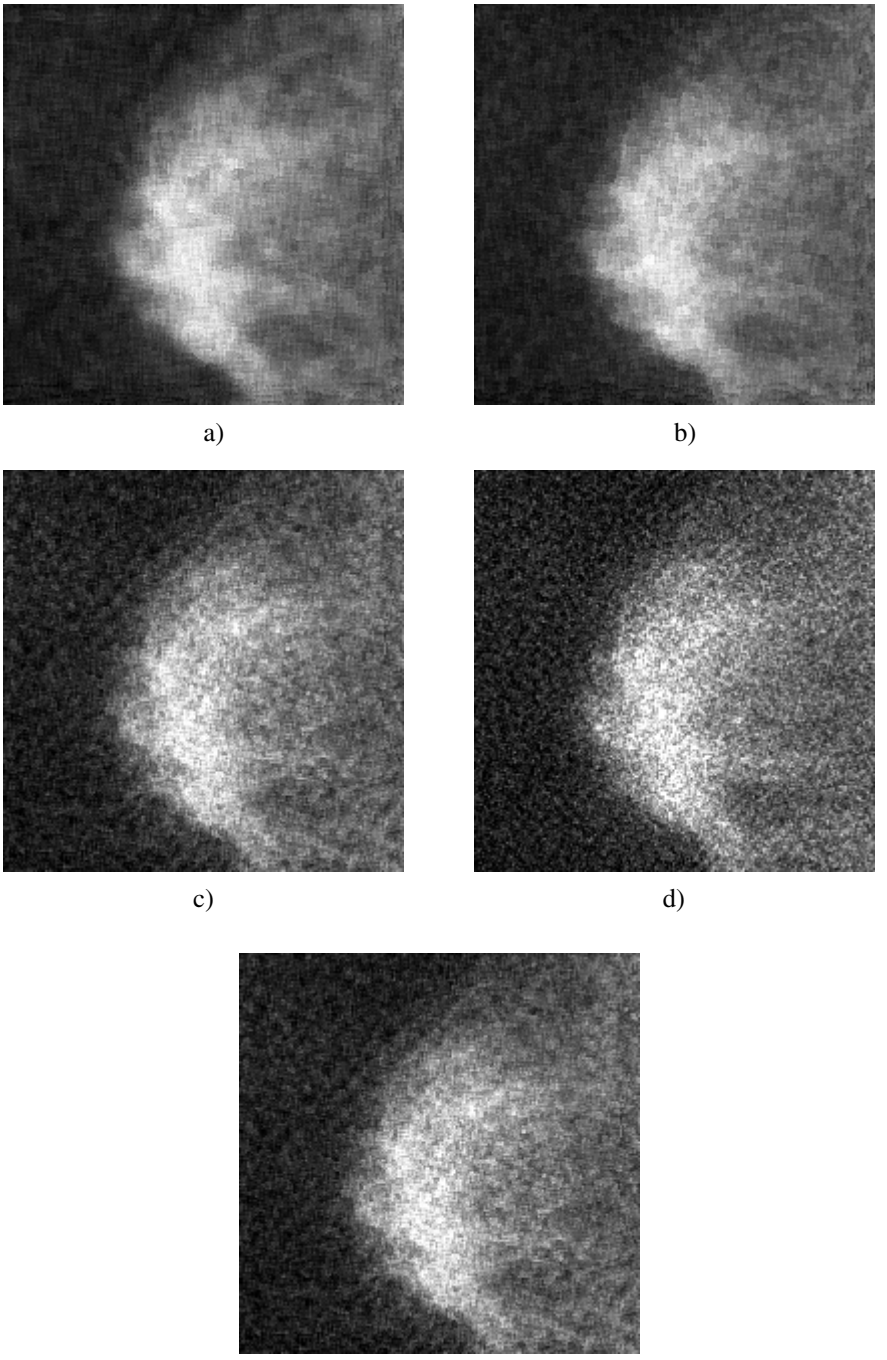


Fig. 2. Enhancing images with a) $h_1(N, x)$, b) $h_5(N, x)$, c) $h_{10}(N, x)$, $h_{15}(N, x)$ and e) $h_{20}(N, x)$

Table 1. I. Quantitative evaluation

RMSE	$h_1(N, x)$	$h_5(N, x)$	$h_{10}(N, x)$	$h_{15}(N, x)$	$h_{20}(N, x)$
Matrix	0.1238	0.2934	0.4594	0.6256	0.6874
Recursive	0.1283	0.2301	0.4376	0.6211	0.6834

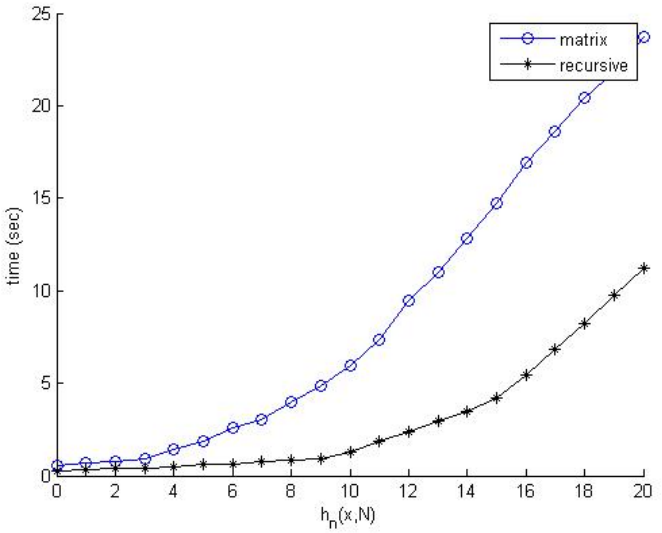


Fig. 3. Pixel processing using the Impulse Response of UFIR filter vs Time

6 Conclusions

In this paper we presented the development of two forms to derive the impulse response of the UFIR filter. The matrix and recursive form can be used to filter biomedical images interchangeably. The quantitative evaluation (Table I) of both forms is quite similar. The processing time of each form is the main factor to consider. In Fig. 3 shows that the higher order approximation, the greater the processing time required is, and we can conclude that recursive approach requires less processing time than the matrix form. Finally, this comparison would help to significantly reduce the computational complexity required to implement the algorithm in a signal processor.

References

1. Najarian, K., Splinter, R.: Biomedical Signal and Image Processing. CRC Taylor & Francis Group, New York (2006)
2. Jan, H.: Medical Image Processing, Reconstruction and Restoration – Concepts and Methods. CRC Taylor & Francis Group, New York (2006)

3. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. Prentice Hall, New Jersey (2002)
4. Jähne, B.: Digital Image Processing. Springer, Berlin (2002)
5. Pitas, I., Venetsanopoulos, A.: Nonlinear Digital Filters – Principles and Applications. Kluwer Academic Publishers (1990)
6. Kalouptsidis, N., Theodoridis, S.: Adaptive System Identification and Signal Processing Algorithms. Prentice Hall (1993)
7. Astola, J., Kuosmanen, P.: Fundamentals of Nonlinear Digital Filters. CRC Press (1997)
8. Shmaliy, Y.: An unbiased FIR filter for TIE model of a local clock in applications to GPS-based timekeeping. IEEE Trans. on Ultrasonic, Ferroelectrics and Frequency Control 53(5), 862–870 (2006)
9. Shmaliy, Y.: An unbiased p-step predictive FIR filter for a class of noise-free discrete time models with independently observed states. Signal, Image & Video Processing 3(2), 127–135 (2009)
10. Shmaliy, Y.: GPS-based Optimal FIR Filtering of Clock Models. NOVA Science Publishers, New York (2009)
11. Shmaliy, Y., Morales-Mendoza, L.J.: FIR smoothing of discrete-time polynomial signals in state space. IEEE Trans. on Signal Processing 58(5), 2544–2555 (2010)
12. Morales-Mendoza, L.J., Shmaliy, Y., Ibarra-Manzano, O.G., Arceo-Miquel, L.J., Montiel-Rodriguez, M.: Moving Average Hybrid FIR Filter in Ultrasound Image Processing. In: IEEE Proceeding of 18th CONIELECOMP, Puebla, México, pp. 160–164 (2008)
13. Morales-Mendoza, L.J., Shmaliy, Y.: Moving Average Hybrid Filter to the EnhancingUltrasound Image Processing. IEEE Trans. on America Latina 8(1), 9–16 (2010)
14. Morales-Mendoza, L.J., Shmaliy, Y., Ibarra-Manzano, O.G.: Enhancing Ultrasound Images using Hybrid FIR Structures. In: Image Process, ch. 16, pp. 287–310. In Tech, Vienna (2009)
15. Morales-Mendoza, L.J., Shmaliy, Y., Vázquez-Bautista, R.F., Ibarra-Manzano, O.G.: Smoothing of ultrasound images with the p-lag FIR structures. In: WSEAS Proceeding of 10th SIP, Canary Islands, Spain, pp. 47–52 (2011)
16. Morales-Mendoza, L.J., Vázquez-Bautista, R.F., Morales-Mendoza, E., Shmaliy, Y.: Speckle noise reduction in ultrasound imaging using the key points in low degree unbiased FIR filters. Computación y Sistemas IPN 16(3), 287–295 (2012)
17. Morales-Mendoza, L.J., Vázquez-Bautista, R.F., Morales-Mendoza, E., Shmaliy, Y., Gamboa-Rosales, H.: A new recursive scheme of the unbiased FIR filter to image processing. Procedia Engineering 35, 202–209 (2012)
18. Morales-Mendoza, L.J., Shmaliy, Y., Vázquez-Bautista, R.F., Ibarra-Manzano, O.G., Morales-Mendoza, E.: Ultrasound images processing using the recursive p-step unbiased FIR filter. In: WSEAS Proceeding of 12th ISCGAV, Istanbul, Turkey, pp. 100–106 (2012)
19. Morales-Mendoza, L.J., Shmaliy, Y., González-Lee, M., Morales-Mendoza, E., Varguez-Fernández, R.: Toward to recurrent form of the impulse response of UFIR filters. Procedia Technology (in press, 2013)
20. Gamboa-Rosales, H., Morales-Mendoza, L.J., Shmaliy, Y.: Unbiased impulse responses – a class of discrete orthogonal polynomials. ICIC Express Letters 7(7), 2005–2010 (2013)
21. Morales-Mendoza, L.J., Gamboa-Rosales, H., Shmaliy, Y.: A new class of discrete orthogonal polynomials for blind fitting of finite data. Signal Processing 93(7), 1785–1793 (2013)

EEG PATTERN RECOGNITION: Application to a Real Time Control System for Android-Based Mobile Devices

Liliana Gutiérrez-Flores, Carlos Avilés-Cruz,
Juan Villegas-Cortez, and Andrés Ferreyra-Ramírez

Universidad Autónoma Metropolitana, Azcapotzalco. Departamento de Electrónica,
San Pablo Xalpa No.180, Col. Reynosa Tamaulipas, CP 02200, México, D.F.
{ligf,caviles,juanvc,fra}@correo.azc.uam.mx

Abstract. This paper describes a new EEG pattern recognition methodology in Brain Computer Interface (BCI) field. The EEG signal is analyzed in real time looking for detection of "intents of movement". The signal is processed at specific segments in order to classify mental tasks then a message is formulated and sent to a mobile device to execute a command. The signal analysis is carried out through eight frequency bands within the range of 0 to 32 Hz. A feature vector is conformed using histograms of gradients according to 4 orientations, subsequently the features feed a Gaussian classifier. Our methodology was tested using BCI Competition IV data sets I. For "intents of movements" we detect up to 95% with 0.2 associated noise, with mental task differentiation around 99%. This methodology has been tested building a prototype using an Android based mobile telephone and data gathered with an EPOC Emotive headset, showing very promising results.

Keywords: EEG Pattern Recognition, Self-Paced Control, BCI application, Mental Tasks Differentiation.

1 Introduction

A Brain Computer Interface (BCI) is a system that enables communication between brain and computer. For the last two decades, there have been a great interest using Electro-Encephalographic (EEG) signals on applications in order to help people with motor disabilities, such as amyotrophic lateral sclerosis [1] or spinal cord injury [2]. There are works implementing EEG signal processing, such as Tanaka et al. [3], which used left and right thinking to control the direction of an electric wheel chair; and Müller et al. [2] which used motor imagery movements to control prosthesis for the upper extremities. Recently, control applications have been developed for other purposes, for example: differentiation of six limb movements was used to control the Khepera robot [4]; exploration, communication or monitoring space applications were improved with BCI technologies [5]; a BCI integrated to computer vision system was used to prioritize the interests of a person in different images [6]; and the differentiation of right

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 232–241, 2013.

and left imagery movements were used to control a video-game [7]. Our vision consists of people using thoughts to control mobiles in their daily lives. A lot of issues need to be solved, mainly concerning the usability of the EEG signal acquisition devices, but also with the effective differentiation of the mental task on real time, the efficient processing of the EEG signal over portable computers and of course wireless communications.

Synchronous or asynchronous approaches have been used to process the EEG signals. Synchronous is useful only for offline characterization and analysis of the signal; for online control, an asynchronous approach is needed. Plenty of work has to do with the fact of enabling self-paced control, the work of Borisoff et al. [8] was developed for enabling a switch to be activated with a movement of a finger, achieving a detection of 80%; and Faradji et al. [9] processed data from five mental tasks in a self-paced fashion with 70% of detection, but in some cases without noise. Feature extraction is a very important stage, the more properly built the features, the more efficient results we got from the classifier, even if it is a simple one. Feature vectors have been conformed on a great variety of forms, from the work of Keirn et al. [10] which used left-right asymmetric ratio and spectral density with 90% of classification results with a Quadratic Bayesian classifier; adaptive autoregressive parameters and event related synchronization/desynchronization of the μ and β rhythms where used by Pfurtscheller et al. [1], [11], [2]; Zhang et. al used power spectral entropy [12] achieving 90% of classification with a time-variable linear classifier; wavelet transform was used by Qin et al. [13] and Bostanov [14], the latter used a linear discriminant analysis to got 84% of classification, and Sun used energy difference with a support vector machine achieving 90% [15].

In the present work an EEG pattern recognition methodology and a prototype of one application in a real time control system of a mobile device are presented. The major functional blocks are shown in Fig.1. As the ongoing EEG signal is received, it must be continuously analyzed, we can not know in advance at what time the user is going to perform a mental task with the intention of executing a command, so we need to detect that precise time. In order to classify a mental task and correlates it to a command, the signal is processed in a window around the time detected in the previous stage. A feature vector is built over this window and then it is fed to a statistical parametric classifier (in our case a Gaussian classifier is used). The rest of the paper is organized as follows: In Section 2 our EEG pattern recognition methodology proposal is detailed; the evaluation data are shown in Section 3; experimental results and prototype are explained in Section 4; and finally conclusions and further work are presented in Section 5.

Fig. 1. Major functional blocks of the real time control system

2 EEG Pattern Recognition Methodology Proposal

2.1 Time-Frequency Pre-processing

The whole input EEG signal is conformed by $N-$separated signal, where N represents the number of available channels. The signal is processed over one-second windows. Lets call S^k the signal for any $k-$window (1), M represents the number of samples per second.

$$\mathbf{S}^k = \begin{pmatrix} s^k_{1,1} & \cdots & s^k_{1,j} & \cdots & s^k_{1,M} \\ \vdots & & \vdots & & \vdots \\ s^k_{i,1} & \cdots & s^k_{i,j} & \cdots & s^k_{i,M} \\ \vdots & & \vdots & & \vdots \\ s^k_{N,1} & \cdots & s^k_{N,j} & \cdots & s^k_{N,M} \end{pmatrix} . \tag{1}$$

Since the EEG signal is highly non-stationary, the shift between one window and the next consists of only one sample. A new window is conformed for each sample after the first second of the signal elapsed. The mean of the signal for each channel $\overline{S^k_i}$ is subtracted from every S^k_i row to eliminate the offset and produce $\widetilde{S^k_i}$. The spectral power P^k_i for each channel is calculated using the short-Fourier transform ($\mathscr{F}$) with a Hanning window using (2) and subsequently the result is multiplied by its complex conjugate (3).

$$\mathscr{F}\{\widetilde{S^k_i}\} = \sum_{n=-\infty}^{\infty} \widetilde{S^k_i} W_{Hanning}[n] \boldsymbol{e}^{-j\omega n} . \tag{2}$$

$$P^k_i = \mathscr{F}\{\widetilde{S^k_i}\} \cdot \mathscr{F}^*\{\widetilde{S^k_i}\} . \tag{3}$$

Low frequencies, between 0 and 32 Hz, associated to delta (δ), theta (θ), alpha (α) and beta (β) rhythms are the most important part of the spectral power P^k_i. Eight bands of frequency are conformed from P^k_i as follows: $(0-4], (4-8], (8-12], (12-16], (16-20], (20-24], (24-28], (28-32]$. A single value P^k_{ij} associated to each frequency band is calculated using (4), $i = 1, ..., N$ denotes the channel, $j = 1, ..., 8$ denotes the band of frequency, and $p^k_{i,m}$ the $m-$value of P^k_i, for $m = 1, ..., 32$.

$$P^k_{i,j} = \frac{1}{4} \sum_{m=4(j-1)+1}^{m=4(j-1)+4} p^k_{i,m} , \; m = 1, ...32 \; j = 1, ...8 . \tag{4}$$

As a result, each $k-$window produces eight values per channel. The k discrete variable can easily be associated to a specific instant of time, lets say the middle of the $k-$windows. Figure 2 shows the plots of the power values for each frequency band against time.

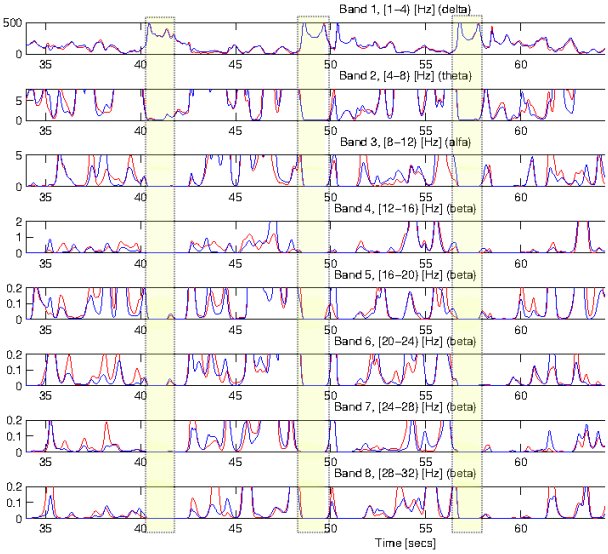


Fig. 2. Spectral power plots for the eight frequency bands

2.2 Detection of Intent of Movement

The "intent of movement" detection is important because it allows the signal to be processed, for differentiation purposes, at specific periods of time. When the subject imagines himself performing a movement, some frequency bands are blocked, others are enhanced and some others are synchronized with the signal from the electrode on the opposite side of the brain. An *up-flag* is turned *on* if the value of any frequency band is higher than an *upper threshold*, a *down-flag* indicates if the value is lower than a *bottom threshold* and a *sync-flag* indicates that the difference between two channels is lower than a *synchronization threshold*. For purposes of robustness, a $36 - sample$ stabilization window is defined. An event happens, if some of the flags are turned *on* during the stabilization window. Using this mechanism, a lot of noise associated with short-in-time jumps of the signal are eliminated. Even more, if only certain percentage of the stabilization window is required, not 100%, real mental tasks are not discarded if they do not fully reach the threshold during the complete stabilization window. As a result, each $i-$channel will have blocking ${}_{\boldsymbol{b}}\boldsymbol{e}_i^k$, enhancing ${}_{\boldsymbol{e}}\boldsymbol{e}_i^k$ and synchronizing events ${}_{\boldsymbol{s}}\boldsymbol{e}_i^k$, associated to each frequency band as shown in (5).

$$ {}_{\boldsymbol{b}}\boldsymbol{e}_i^k = [{}_b e_{i,1}^k, \cdots {}_b\, e_{i,j}^k, \cdots {}_b\, e_{i,8}^k] \,. \tag{5a} $$

$$ {}_{\boldsymbol{e}}\boldsymbol{e}_i^k = [{}_e e_{i,1}^k, \cdots {}_e\, e_{i,j}^k, \cdots {}_e\, e_{i,8}^k] \,. \tag{5b} $$

$$ {}_{\boldsymbol{s}}\boldsymbol{e}_i^k = [{}_s e_{i,1}^k, \cdots {}_s\, e_{i,j}^k, \cdots {}_s\, e_{i,8}^k] \,. \tag{5c} $$

If we are looking for an enhancing event, the *upper threshold* is relevant while the *bottom threshold* does not matter, it is set to zero this way is never reached. If we are looking for a blocking event, the bottom threshold is relevant while the upper does not matter and hence it set to infinite (or high enough) to be unreachable. Lets say that zero or infinite thresholds are simply not-relevant thresholds. Each person has characteristic numbers of blocking events N^-, enhancing events N^+ and synchronizing events $N^{\tilde{}}$whereas performs a "mental task". An "intent of movement" is detected if these characteristic numbers (N^-, N^+, $N^{\tilde{}}$)are reached at a specific instant of time.

The length of the interval of time where a mental task are assumed to be performed, varies according to the acquisition protocol between 2 and 8 secs. We presume that a motor-imaginary mental task takes only 0.3 secs, hence periods in the signal where the "mental task" is performed are really conformed with two classes of samples, one corresponding to the true mental task realization and other corresponding to relax-time. Thresholds for an specific data set are estimated through an statistical analysis of the signal over mental task realization periods, in comparison with relax-periods.

2.3 Feature Vector Construction and Classification

A feature vector is conformed from the spectral power values within a region around $P_{i,j}^{k}$, lets call $P_{i,j}^{k,r}$ the $r-$value in the region. Each r value of spectral power is normalized to obtain $\widetilde{P_{i,j}^{k,r}}$, first the minimum value in the segment is subtracted and subsequently it is divided by the difference between the maximum and the minimum. A gradient vector $\nabla \widetilde{P_{i,j}^{k,r}}$ is estimated using an spectral power difference at every inner point in the segment of the signal (6).

$$\nabla \widetilde{P_{i,j}^{k,r}} = \left(\widetilde{P_{i,j}^{k,r+1}} - \widetilde{P_{i,j}^{k,r-1}},\ t^{r+1} - t^{r-1} \right) . \tag{6}$$

Notice that the second element of the vector is constant. Instead of that value, the mean of the spectral power differences is used, it allows to enhance differentiation of the angles. A $4-$bin histograms with the gradients are built, vectors for every point are grouped according to their angles as follows: $bin\_1 = [45°,\ 90°)$, $bin\_2 = [0°,\ 45°)$, $bin\_3 = [-45°,\ 0°)$ and $bin\_4 = (-90°,\ -45°)$, subsequently they are counted to conform the histogram. For robustness, the count is weighted with the magnitude of the gradient (7). As a result, a $4-$bin histogram for each channel and each band associated to the "intent of movement" is obtained.

$$\mathbf{H}(bin\_i) = \sum_{r} |\nabla \widetilde{\mathbf{P}_r}| \ , \text{ for } r \text{ such that } \angle \nabla \widetilde{\mathbf{P}_r} \in bin\_i \ . \tag{7}$$

The featured vector fed to the classifier $\overrightarrow{F}$ is integrated with a selection of the histograms for certain bands of frequency and channels. In the present work, the feature vector was conformed on a $32-$sample segment, using $2^n d$ and $6^t h$

frequency bands and adding the *mode* and *mean* values for robustness, the final dimension of the vector is 48. For classification we use a Gaussian classifier (8), we believe that a simple classifier is efficient when the input feature vector is built properly.

$$p(\overrightarrow{F}/c_i) = \frac{1}{(2\pi)^{\left(\frac{n}{2}\right)}|\Sigma_i|^{\left(\frac{1}{2}\right)}} \exp\left[-\frac{1}{2}(\overrightarrow{F} - \overrightarrow{\mu_i})^T \Sigma_i^{-1}(\overrightarrow{F} - \overrightarrow{\mu_i})\right]. \quad (8)$$

3 Evaluation Data

The proposed strategy for the EEG pattern recognition was tested with two different input data. First, BCI Competition IV data sets I, provided by the Berlin Group [16]. Those data were selected because they have an asynchronous approach, suitable for our objective. In addition, data gathered in our laboratory using the EPOC headset from Emotiv Systems [1] were used too.

3.1 BCI Competition IV Data Sets I

EEG data from 59 channels, recorded at a rate of 100 samples per second, from healthy people whereas performing one motor imagery task. The classes of mental task to perform are: (i) imaginary movement of the left hand, (ii) imaginary movement of the right hand and (iii) imaginary movement of any foot. Calibration data were recorded as follows: an arrow was displayed on a computer screen indicating the class of the motor imagery task to perform, the arrow was presented for period of 4 seconds, during which the subject was supposed to imagine the performance of the movement. Periods were interleaved with 2 seconds of blank screen and 2 seconds with a cross in the center of the screen, the cross superimposed to the cue, so it was displayed during 6 seconds.

3.2 Data Gathered at Our Laboratory

EEG-data were recorded from the 14 available channels using an EPOC headset from Emotiv System, at a rate of 128 samples per second. Our objective was to gather data with variable time between each mental task realization to simulate self-paced control. We worked with six healthy people, between 20 and 30 years old, each person was asked to select two mental tasks from the following ones: (i) imaginary movement to the right of right hand, (ii) imaginary movement to the left of the left hand, (iii) imaginary movement over the head of both hands and (iv) imaginary movement of the head. In order to get 200 mental task records from each person, five recordings with 40 mental task were performed. A white circle was presented in the middle of a computer screen, it was moved during 2 seconds to indicate the kind of mental task to be performed. The time between each mental task varied between 6 and 10 seconds. Further explanation are available at our Web Page by request. [2]

[1] Emotiv System, `http://emotiv.com/`

[2] uamaML datasetI, `http://www.eegspiga.com`

Table 1. Detection of "intents of movement" and classification for BCI Competition IV dataset I

Data set	Detection rate	Noise	Classification rate
BCICIV_calib_ds1b	98%	0.18	99%
BCICIV_calib_ds1c	96%	0.23	97%
BCICIV_calib_ds1d	92%	0.19	92%
BCICIV_calib_ds1e	97%	0.21	97%
BCICIV_calib_ds1g	92%	0.18	98%
uamaML_datasetIb	76%	0.5	81%

4 Experimental Results and Discussion

Electrodes located over the sensorimotor cortex were taken, according to the $10-10$ electrode configuration system. For the BCI Competition IV data, signals in 27^{th} and 31^{st} channels, corresponding to $C3$ and $C4$ electrode positions were used in the present work. Similarly, for our own data, signal from 6^{th} and 13^{th} channels, corresponding to $FC5$ and $FC6$ electrode positions were used. To evaluate the "intent of movement" detection, a detection rate and an associated noise are calculated from (9) and (10), where TP stands for True Positives, FP for False Positives and FN for False Negatives.

$$\text{Detection Rate} = \frac{TP}{(TP + FN)} \, . \tag{9}$$

$$\text{Noise} = 1 - \frac{TP}{(TP + FP)} \, . \tag{10}$$

The proposed strategy detects up to the 98% of the imagery movements for the five evaluated dataset from the BCI Competition IV data. The noise associated to the false positives is around 0.2 as shown in Tab. 1. The work of Zhang et al. [17] used the same data for self-paced EEG-based motor imagery detection. As a results [17] shown a mean-squared-error of class label prediction for dataset *ds1b* around 0.3 and around 0.23 for dataset *ds1g*. For dataset *ds1b* we used [180, 0, 0, 0, 0, 0, 0, 0] for upper thresholds and [Inf, 0.2, 0.07, 0.07, 0.007, 0.007, 0.007, 0.007] for bottom ones, it means that an enhancing-event over band 1 and blocking-events in the other bands were searched. You can see how it is noticeable in Fig. 2, over the dashed areas while the imagery mental task occurs. The classification results using a Gaussian classifier, are close to 99%, leave-one-out evaluated for the BCI Competition IV data, as shown in Tab. 1. Results with our own data reach a detection up to 70% of "intent of movement" and 80% of good classification as shown in last row of Tab. 1.

The current system architecture of the prototype is shown in Fig. 3, it includes a MS-Windows Vista PC which receives wirelessly the EEG signal from the EPOC headset at a rate of 128 samples per second. The PC processes data and communicates to a mobile device via Bluetooth. The signal processing programs were developed using Matlab. The client-server Bluetooth application between

Fig. 3. System architecture of the prototype

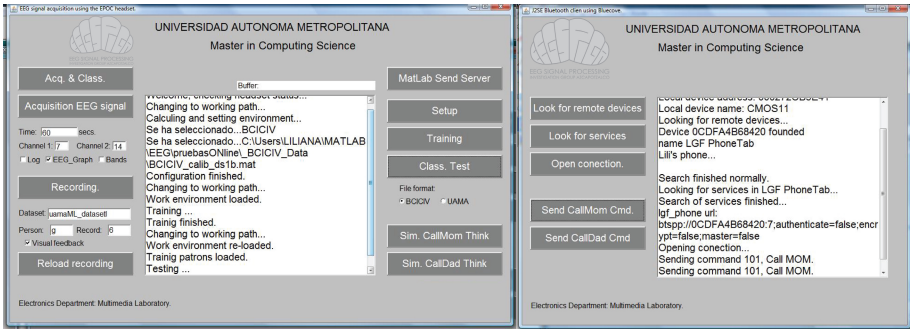


Fig. 4. Application GUI

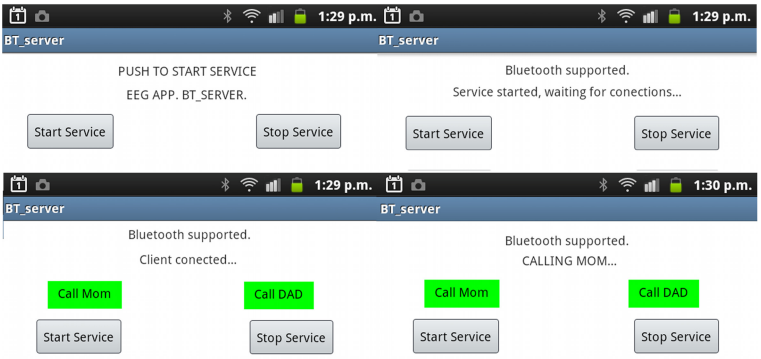


Fig. 5. Screens for the mobile device

the PC and the mobile device was developed using Blue Cove 2.1.0 API JSR-82 implementation library for J2SE [3] and the Bluetooth Android API. [4] The final application has a graphic interface conformed with two windows as shown in Fig. 4. The first one is to work with the EEG signal data: acquisition, training

[3] BlueCove 2.1.0 API, `http://bluecove.org/bluecove/apidocs/`
[4] Android Developers:Bluetooth, `http://developer.android.com/guide/topics/wireless/bluetooth.html`

and using databases files to simulate an input signal. The second one is for the management of Bluetooth connectivity: looking for the remote mobile device, searching for services and establishing connection. Functional evaluation were performed, mental task 1 was associated to the command "call mom" and mental task 2 was associated to "call dad". In the mobile device the server process is started, once the connection is established, when an "intent of movement" is detected and classified, a message is sent to the mobile device in order to make the phone call, as shown in sequence of Fig. 5.

5 Conclusions and Further Work

This work presents a comprehensive point of view for creating a real time control system suitable for mobile devices using EEG signals. Previous works addressed only a particular aspect such as: (i) enabling asynchronous control, (ii) strategy for integrate a feature vector and classification, (iii) final application and (iv) acquisition EEG signal device. Concerning to the EEG signal processing, a phased strategy is proposed, first the detection of an "intent of movement", then the feature vector conformed around the detected instant, as the signal is received and finally the classification. Excellent results were obtained using the BCI Competition IV data sets I. Comparing to related asynchronous works, Borisoff et al. [8] reached up to 80% of good detection for finger movements and Faradji et al. [9] reached up to 70% for five mental tasks, some cases with no-noise. Our methodology reached up to 98%. The submitted mental tasks were classified correctly with percentages of classification around 95% using a simple Gaussian-classifier. Comparing to other methodologies Keirn et al. [10] reached 90% of good classification, Zhang et. al [12] reached 90% and Bostanov [14] reached 84%.

Processing remains lightweight and can be carried out in the mobile device. Our vision for future work persues to gather and processes the signal from the mobile. Taking into account the convenience of using a relatively easy and affordable EEG signal acquisition device. Our results are promising, we reached a detection up to 70% of "intent of movement" and 80% of good classification. Much work has to be done in regard to the usability of those commercially available devices. Further work comprises a strategy to minimize the noise. Either with a mechanism to detect false positives, avoiding large amount of noise that is sent to the classifier; or by implementing a rejection class that catches all the noise.

Acknowledgments. The authors give thanks to UAM (Universidad Autonoma Metropolitana), ICYTDF (Instituto de Ciencia y Tecnologa del Distrito Federal) and PROMEP under grant UAM-PTC-328 for the economical support.

References

1. Pfurtscheller, G., Neuper, C., Schlöl, A., Legger, K.: Separability of EEG signals recordered during right and left motor imagery using adaptive autoregressive parameters. IEEE Transactions on Rehabilitation Engineering 6(3), 316–325 (1998)
2. Müller-Putz, G.R., Scherer, R., Pfurtscheller, G., Neuper, C., Rupp, R.: Non-invasive control of neuroprostheses for upper extremity: Temporal coding of brain patterns. In: EMBS (2009), doi:10.1109/IEMBS.2009.5333185.
3. Tanaka, K., Matsunaga, K., Wang, H.O.: Electroencephalogram-based control of an electric wheel-chair. IEEE Transactions on Robotics 21(4), 762–766 (2005)
4. Chakraborti, T., Sengupta, A., Banerjee, D., Konar, A., Bhattacharyya, S., Khasnobish, A., Janarthanan, R.: Implementation of EEG based control of remote robotic systems. In: International Conference on Recent Trends in Information Systems, pp. 203–208 (2011)
5. De Negueruela, C., Broschart, M., Menon, C., Millan, J.: Brain computer interfaces for space applications. Pers. Ubiquit. Comput. 15, 527–537 (2011)
6. Sajda, P., Pohlmeyer, E., Wang, J., Parra, L.C., Christoforou, C., Dmochowski, J., Hanna, B., Bahlmann, C., Singh, M.K., Chang, S.F.: In a blink of an eye and a switch of a transistor: Cortically coupled computer vision. Proceedings of the IEEE 98(3), 462–478 (2010)
7. Pour, P.A., Gulrez, T., AlZoubi, O., Gargiulo, G., Calvo, R.A.: Brain-Computer Interface: Next generation thought controlled distributed video game development platform. In: CIG, pp. 251–257 (2008)
8. Borisoff, J.F., Mason, S.G., Bashashati, A., Birch, G.E.: Brain computer interface design for asynchronous control applications: Improvements to the LF-ASD asynchronous brain switch. IEEE Trans. on Biomedical Eng. 51(6), 985–992 (2004)
9. Faradji, F., Ward, R.K., Birch, G.E.: A brain computer interface based on mental tasks with a zero false activation rate. In: 4th International IEEE EMBS Conference on Neural Engineering, pp. 355–358 (April-May 2009)
10. Keirn, Z.A., Aunon, J.I.: A new mode of communication between man and his surroundings. IEEE Trans. on Biomedical Engineering 37(12), 1209–1214 (1990)
11. Pfurtscheller, G., Lopes da Silva, F.: Event-related EEG/MEG synchronization and desynchronization: basic principles. Clin. Neuroph. 110, 1842–1857 (1999)
12. Zhang, A., Yang, B., Huang, L.: Feature extraction of EEG signals using power spectral entropy. In: IEEE International Conference on BioMedical Engineering and Informatics, pp. 435–439 (2008)
13. Qin, S., Ji, Z.: Extraction of feature information in EEG signal by virtual EEG instrument with the functions of time-frequency analysis. In: 6th International Symposium on Image and Signal Processing and Analysis, pp. 7–11 (2009)
14. Bostanov, V.: BCI Competition 2003 - Data Sets Ib and IIb: Feature extraction from event-related brain potentials with the continuous wavelet transform and t-value scalogram. IEEE Trans. on Biomedical Engineering 51(6), 1057–1061 (2004)
15. Sun, S.: Extreme energy difference for feature extraction of EEG signals. Expert System with Applications 37, 4350–4357 (2009)
16. Blankertz, B., Dornhege, G., Krauledat, M., Müller, K.R., Curio, G.: The non-invasive Berlin Brain-Computer Interface: Fast acquisition of effective performance in untrained subjects. NeuroImage 37(2), 539–550 (2007)
17. Zhang, H., Guan, C., Ang, K.K., Wang, C., Chin, Z.Y.: BCI competition IV - data sets I: Learning discriminative patterns for self-paced EEG-based motor imagery detection. Frontiers in Neuroscience 6 (2012), doi:10.3389/fnins.2012.00007

Regression via Logic Supervised Classification

Vladimir Ryazanov

Institution of Russian Academy of Sciences, Dorodnicyn Computing Centre of RAS
Vavilov st. 40, 119333 Moscow, Russia
rvvccas@mail.ru

Abstract. An approach to the restoration of dependences (regressions) is proposed that is based on solving problems of supervised classification. The main task is finding the optimal partitioning of the range of values of dependent variable on a finite number of intervals. It is necessary to find optimal number of change-points and their positions. This task is formulated as search and application of piece-wise constant function. When restoring piecewise constant functions, the problem of local discrete optimization using a model of logic supervised classification in leave –one-out mode is solved. The value of the dependent value is calculated in two steps. At first, the problem of classification of feature vector is solved. Further, the dependent variable is calculated as half of the sum of change-points values of the corresponding class.

Keywords: regression, supervised classification, discrete optimization, approximation.

1 Introduction

Many data analysis tasks may be written in the next standard form. Let training sample $\{z_i, \mathbf{x}_i\}, i = 1,2,...,m$, is given, and $\mathbf{x}_i = (x_{i1}, x_{i2},..., x_{in})$ is a feature description of an object, $z_i \in R$, $x_{ij} \in M_j$ (M_j - a set of allowed values of feature № j). Vector $\mathbf{x}_i$ will be considered as a vector of values of independent parameters, and scalar z_i as a dependent value (it is supposed that z_i may be calculated by $\mathbf{x}_i$, i.e. $z_i = f(\mathbf{x}_i)$). It takes to calculate $z = f(\mathbf{x})$, $z \in R$ for any new $\mathbf{x} = (x_1, x_2,..., x_n)$, $x_j \in M_j$.Vector $\mathbf{x}$ is the objects (situation, phenomenon or process) description in term of features, and z is the value of some of its hidden scalar characteristic.

This problem in the statistical formulation is known as the problem of the regression reconstruction. Regression is a function of conditional expectation, assuming the existence of conditional density $p(z \mid \mathbf{x})$. In this paper, we will not use any probabilistic models.

Currently, there are different approaches to the restoration of regressions when $x_i \in R, i = 1,2,..., n$, which can be conditionally divided into two types:

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 242–253, 2013.

parametric and nonparametric. Parametric approach [1] assumes the functional dependence of a certain type, depending on the parameters $\boldsymbol{\omega}$:

- linear regression - $f(\boldsymbol{\omega},\mathbf{x})=\sum_{j=1}^{n}\omega_j x_j+\omega_0$;
- polynomial regression of degree γ - $f(\boldsymbol{\omega},\mathbf{x})=\sum_{p_1=0}^{\gamma_1}\ldots\sum_{p_n=0}^{\gamma_n}\omega_{p_1\ldots p_n}x_1^{p_1}\ldots x_n^{p_n},\gamma=\sum_{i=1}^{n}\gamma_i$;
- curvilinear regression - $f(\boldsymbol{\omega},\mathbf{x})=\sum_{j=1}^{k}\omega_j\phi_j(x_1,\ldots,x_n)+\omega_0$, $\phi_1,\ldots,\phi_k$ - transformations $R^n\to R$;
- logistic regression: $f(\boldsymbol{\omega},\mathbf{x})=\dfrac{1}{1+\exp(-s)}$, $s=\sum_{j=1}^{n}\omega_j x_j+\omega_0$.

In the non-parametric approach [2] characteristic z for $\mathbf{x}$ is defined as $z=\dfrac{\sum_{i=1}^{m}\omega_i(\mathbf{x})z_i}{\sum_{i=1}^{m}\omega_i(\mathbf{x})}$, where $\omega_i(\mathbf{x})=K\left(\dfrac{\rho(\mathbf{x},\mathbf{x}_i)}{h}\right)$, $i=1,2,\ldots,m$, K - kernel function, h - the width of the window. Well-known methods of support vector machine regressions [3] may be considered as curvilinear regressions.

In recent years a large number of publications devoted to regression via classification. In [4] it was proposed general fuzzy piecewise regression where change-points and their positions are obtained simultaneously as a solution of a mixed-integer programming problem. It is supposed that $\mathbf{x}\in R^n$. Change-points which are the joints of the pieces are quoted from conventional statistical piecewise regression. The proposed fuzzy piecewise regression is the direct generalization of linear piecewise regression. In [5] the Bayesian regression algorithm for piecewise constant functions with unknown segment number, location and level is proposed. It is assumed that one-dimension measurements of some function at discrete locations are given and measurements are independently distributed. The common polynomial-time dynamic-programming-type algorithm is derived.

A heuristics approach for learning regression rules by transforming the regression problem into a classification task is proposed in [6]. The discretization of the class variable is integrated into the rule learning algorithm. The key idea is the dynamical definition of a region around the target value predicted by the rule. The common approaches to construction of regression via classification are observed in [7]. The authors note that standard methods

comprise two stages: the discretization of the numeric dependent variable in order to learn a classification model, and the reverse process of transforming the class output of the model into a numeric prediction. The discretization of target variable is considered usually as some unsupervised classification task. The universal estimator for the regression problem in supervised learning is considered in [8]. The based on a least-square fitting procedure estimator does not depend on any a priori assumptions about the regression function to be estimated. It is proved that if the regression function is of a certain class, then the estimator converges to the regression function with an optimal rate of convergence in terms of the number of samples. It is assumed the existence of probability measure on R^{n+1}. There are other closes in a matter of fact approaches.

It should be noted certain limitations of these approaches. Parametric approaches require a priori knowledge of the analytical form of functions. The presence of different types of features (real, nominal, binary, ordinal, etc.) requires additional tools for describing objects in a single scale. Nonparametric methods use widely frequency estimation, distance functions, which can be very approximate and practically difficult for samples of small length, with a large number of independent parameters under various information and diverse nature. Many studies suggest a probabilistic model of the data and $\mathbf{x} \in R^n$. Construction of the functions of multiple nonlinear regressions using the analytical methods of mathematical statistics is impossible in most cases.

In this paper we propose an approach not involving probabilistic assumptions and based on supervised classification. According to the training set are the change-points that define an optimal partition of the sample into a finite number of classes. To find the optimal number of classes and positions of change-points the logic supervised classification model with leave-one-out procedure is used. All problems related to different information content of features, their type, and metrics are transferred to the level of supervised classification where the effective logic methods for supervised classification are used. The optimal number of change-points, their positions, and the classification algorithm are obtained simultaneously by solution of the discrete optimization task and use fast procedure for re-training of neighboring supervised classification tasks.

The basic idea is as follows. We assume that the range of dependent variable is the interval $[a,b]$, $a = \min\limits_{i=1,2,...,m} z_i$, $b = \max\limits_{i=1,2,...,m} z_i$. There is a partition of the segment $[a,b]$ using points $a_1 < a_2 < ... < a_{l-1}$ for l segments $I_1 = [a_0, a_1], I_2 = (a_1, a_2],..., I_l = (a_{l-1}, a_l]$, $a_0 = a$, $a_l = b$. Then the problem of approximate calculating the value of the regression $z = f(\mathbf{x})$ can be solved as follows.

1. The set $\tilde{K}_j = \{\mathbf{x}_i : z_i \in I_j, i = 1,2,...,m\}$ will be corresponding to the segment I_j, $j = 1,2,...,l$. Denote class $K_j = \{\mathbf{x} : z = f(\mathbf{x}) \in I_j\}$, $\tilde{K}_j \subseteq K_j$, $j = 1,2,...,l$.

2. For any $\mathbf{x}$, supervised classification problem is solved with respect to classes $K_\nu, \nu = 1,2,...,l$.

3. We put $z = f(\mathbf{x}) = (a_i + a_{i-1})/2$, if $\mathbf{x}$ classified as $\mathbf{x} \in K_i$.

Function $f(\mathbf{x})$ is uniquely determined by the partition and classification algorithms. So, we must enter the criterion of $f(\mathbf{x}) \equiv f(\mathbf{x}, l, a_1, a_2, ..., a_{l-1})$ quality as a function of the number of change-points l and their positions $a_1, a_2, ..., a_{l-1}$, take classification model and find the optimal solution, which is to construct function $f(\mathbf{x})$.

Note that for the implementation the item number 2, there are various models of classification by precedent. The following sections will be offered to the implementation of this general approach for the case of some logical supervised classification algorithm. Without loss of generality, we assume that the values z_i are different, and the objects of the training sample in ascending order of values z_i, i.e. $z_i < z_{i+1}, i = 1,2,...,m-1$.

The paper is organized as follows. Section 2 develops the theoretical part of paper. The statement of main optimization task is formulated in 2.1. The standard local optimization algorithm is considered in 2.2. The logical supervised classification algorithm is explained in 2.3. It's modification does not require introduction of metric for some feature. The feature may be only ordered. The neighboring classes are considered in main optimization task. It is required to construct a classification algorithm for neighbor task and efficiently to compute the optimized criterion in leave-one-out mode. The problem of recalculation of considered supervised classification algorithm for neighbor classes is considered in 2.4. Section 3 gives some illustration of proposed method and the results of experiments for one practical task. Some remarks are denoted in paper conclusion (section 4).

2 Construction of Approximate Regressions as Piecewise Constant Functions

2.1 Main Optimization Task

Let $A^j \equiv A^j(a_1, a_2, ..., a_{l-1}), j = 1,2,...,m,$ is some classification algorithm with respect to the classes defined in accordance with item 1 of introduction. A^j corresponds to some numbers $a_1 < a_2 < ... < a_{l-1}$, and to sample $\{z_i, \mathbf{x}_i\}, i = 1,2,...,m$, $i \neq j$. Let $A^j(\mathbf{x}_j) = t$ denotes that $\mathbf{x}_j$ is classified by A^j as $\mathbf{x}_j \in K_t$.

Let $l \in \{2,3,...,[m/3]\}$ is fixed. We introduce $F(y_0, y_1, ..., y_{l-1}, y_l)$ as the criterion of $f(\mathbf{x}) \equiv f(\mathbf{x}, l, a_1, a_2, ..., a_{l-1})$ quality, and consider the next discrete optimization task for

$$F(y_0, y_1, ..., y_{l-1}, y_l) = \sum_{i=1}^{m} \left| z_i - (y_{A^i(\mathbf{x}_i)} - y_{A^i(\mathbf{x}_i)-1})/2 \right| \rightarrow \min_{y_1, y_2, ..., y_{l-1}}, \quad (1)$$

$$\left| \{ z_j, j = 1,2,...,m : y_0 \le z_j \le y_1 \} \right| \ge 3, \quad (2)$$

$$\left| \{ z_j, j = 1,2,...,m : y_i < z_j \le y_{i+1} \} \right| \ge 3, \ i = 1,2,...,l-1, \quad (3)$$

$$y_0 = z_1, \ y_i < y_{i+1}, i = 0,1,...,l-1, \ y_l = z_m,$$
$$y_i \in \{z_3, z_4, ..., z_{m-3}\}, i = 1,2,...,l-1, \quad (4)$$

Restrictions (2-3) are the consequence of leave-one-out mode and classification method that were used. The optimal value l and change points $a_1 < a_2 < ... < a_{l-1}$ are calculated by solving task (1-4) for various l.

So, efficiency of task (1-4) solution depends highly from efficiency of $A^j \equiv A^j(a_1, a_2, ..., a_{l-1}), j = 1,2,...,m$ construction. Later we use the local approach for $F(y_0, y_1, y_2, ..., y_l)$ optimization where one logical classification model will be used with fast procedure of classification algorithm re-training for neighboring classification tasks.

2.2 Local Optimization

Consider the problem (1-4), where the function $f(\mathbf{x})$ (and the corresponding classification algorithm A) given by the current values $(y_0, y_1, ..., y_{l-1}, y_l)$. Since objects ordered by increasing z_i, takes place $y_t = z_{i_t}, 1 \le t \le l-1$. A scheme of standard local optimization of $F(y_0, y_1, ..., y_{l-1}, y_l)$, $y_0 = z_1, \ y_l = z_m,$ has been considered. Points $\{(y_0^*, y_1^*, ..., y_{l-1}^*, y_l^*)$, $y_j^* \in \{z_{i_j-1}, z_{i_j+1}\}$, $y_t^* = y_t$, $t \ne j$, $y_0^* = y_0, \ y_l^* = y_l\}$, $j = 1,2,...,l-1$, are called neighboring for $(y_0, y_1, ..., y_{l-1}, y_l)$ if conditions (2-4) are satisfied.

Starting with an arbitrary admissible $(y_0, y_1^{(0)}, ..., y_{l-1}^{(0)}, y_l)$ we browse all nothing more than $2(l-1)$ a neighboring admissible points and find minimum for $F(y_0, y_1, ..., y_{l-1}, y_l)$ in neighborhood of $(y_0, y_1^{(0)}, ..., y_{l-1}^{(0)}, y_l)$. Later, the procedure is repeated for point $(y_0, y_1^{(1)}, ..., y_{l-1}^{(1)}, y_l)$ that is point of F

minimum in neighbor of $(y_0, y_1^{(0)},..., y_{l-1}^{(0)}, y_l)$. Finiteness of local optimization follows from the finiteness of the set of all possible values of $F(y_0, y_1,..., y_{l-1}, y_l)$.

2.3 Supervised Classification Model

Classification algorithms have been considered, which are modifications of estimation calculation algorithms [9]. We describe the principle of the estimation calculation algorithms (ECA).

Let a set X of admissible objects $\mathbf{x} \in R^n$ has the form $X = \bigcup_{j=1}^{l} K_j, K_\nu \cap K_\mu = \emptyset, \quad \nu \neq \mu$. Given training sample $\{z_t, \mathbf{x}_t, t = 1,2,...,m\}$, where $z_t = j$ if $\mathbf{x}_t \in K_j$. Let $\mathbf{x} = (x_1, x_2,..., x_n) \in R^n$, and a training sample contains representatives of all classes. Denote $\tilde{K}_j = \{\mathbf{x}_i : \mathbf{x}_i \in K_j, \ i = 1,2,...,m\}$, $\left|\tilde{K}_j\right| \geq 1$. Let the system of supporting sets $\Omega_A = \{\Omega\}$, $\Omega \subseteq \{1,2,...,n\}$ of algorithm A is the fixed one. Some supporting set Ω defines a subset of features. The proximity of classified object $\mathbf{x}$ to some training object $\mathbf{x}_t$ by support set Ω is defined as

$$\mathrm{B}_\Omega(\mathbf{x}, \mathbf{x}_t) = \begin{cases} 1, & |x_i - x_{ti}| \leq \varepsilon_i, \forall i \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{5}$$

There are the numerical parameters $\varepsilon_i, i = 1,2,...,n$ in (5) set by the user or calculated as, for example, $\varepsilon_i = \frac{2}{m(m-1)} \sum_{\alpha,\beta=1,\alpha<\beta}^{m} \left|x_{\alpha i} - x_{\beta i}\right|$. The estimation $\Gamma_j(\mathbf{x})$ for class $K_j, j = 1,2,...,l$ is calculated for the object $\mathbf{x}$.

$$\Gamma_j(\mathbf{x}) = \frac{1}{\left|\tilde{K}_j\right|} \sum_{\mathbf{x}_i \in \tilde{K}_j} \sum_{\Omega \in \Omega_A} B_\Omega(\mathbf{x}, \mathbf{x}_i). \tag{6}$$

Estimation $\Gamma_j(\mathbf{x})$ characterizes the heuristic degree of proximity of the object $\mathbf{x}$ to the class K_j. Next, apply the decision rule in the space of estimates: the object $\mathbf{x}$ is classified by algorithm A as belonging to class K_j when $\Gamma_j(\mathbf{x}) > \Gamma_i(\mathbf{x}), \forall i \neq j$. Otherwise, the rejection is made. Usually, the set $\Omega_A = \{\Omega : |\Omega| = k\}$, $1 \leq k \leq n$, k - integer, or all possible subsets of $\{1,2,...,n\}$ are used as a system of supporting sets of classification algorithm.

Parameter k is the external one, usually $k = \left[\frac{n}{3}\right]$ is used. In [9] proved that $\Gamma_j(\mathbf{x}) = \frac{1}{\left|\tilde{K}_j\right|} \sum_{\mathbf{x}_i \in \tilde{K}_j} C^k_{d(\mathbf{x},\mathbf{x}_i)}$ in the first method of supporting sets choice, and $\Gamma_j(\mathbf{x}) = \frac{1}{\left|\tilde{K}_j\right|} \sum_{\mathbf{x}_i \in \tilde{K}_j} (2^{d(\mathbf{x},\mathbf{x}_i)} - 1)$ in the second case, where $d(\mathbf{x},\mathbf{x}_i) = \left|\{ j : \left|x_j - x_{ij}\right| \le \varepsilon_j,\ j = 1,2,...,n\}\right|$. In this study, we used some modification of the proximity function (5) and estimation (6).

Let $\mathbf{x}_\alpha, \mathbf{x}_\beta \in \tilde{K}_j$. Define the proximity function $\tilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta)$ of the object $\mathbf{x}$ to the couple $\mathbf{x}_\alpha, \mathbf{x}_\beta$, and its estimation $\tilde{\Gamma}_j(\mathbf{x})$ for the class K_j by the following expressions.

$$\tilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta) = \begin{cases} 1, & (x_{\alpha i} \le x_i \le x_{\beta i}) \vee (x_{\beta i} \le x_i \le x_{\alpha i}), \forall i \in \Omega, \\ 0, & \text{otherwise.} \end{cases} \tag{7}$$

$$\tilde{\Gamma}_j(\mathbf{x}) = \frac{2}{\left|\tilde{K}_j\right|\left(\left|\tilde{K}_j\right| - 1\right)} \sum_{\mathbf{x}_\alpha, \mathbf{x}_\beta \in \tilde{K}_j, \alpha < \beta} \left(\sum_{\Omega \in \Omega_A} \tilde{\mathrm{B}}_\Omega(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta)\right).$$

It can be proved that here also we have similar effective formulas for calculating $\tilde{\Gamma}_j(\mathbf{x})$: $\tilde{\Gamma}_j(\mathbf{x}) = \frac{2}{\left|\tilde{K}_j\right|\left(\left|\tilde{K}_j\right| - 1\right)} \sum_{\mathbf{x}_\alpha, \mathbf{x}_\beta \in \tilde{K}_j, \alpha < \beta} C^k_{d(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta)}$, and

$\tilde{\Gamma}_j(\mathbf{x}) = \frac{2}{\left|\tilde{K}_j\right|\left(\left|\tilde{K}_j\right| - 1\right)} \sum_{\mathbf{x}_\alpha, \mathbf{x}_\beta \in \tilde{K}_j, \alpha < \beta} (2^{d(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta)} - 1)$, where

$d(\mathbf{x}_\alpha, \mathbf{x}, \mathbf{x}_\beta) = \left|\{t : (x_{\alpha t} \le x_t \le x_{\beta t}) \vee (x_{\beta t} \le x_t \le x_{\alpha t}), t = 1,2,...,n\}\right|$. Denote that here $\left|\tilde{K}_j\right| \ge 2$.

After calculating the estimations $\tilde{\Gamma}_j(\mathbf{x}), j = 1,2,...,l,$ the previously given decision rule is used. Note that any metric in feature space and parameters $\varepsilon_i, i = 1,2,...,n$ are not used in this case. Here $x_j \in M_j$ (M_j - linearly ordered set). The features may be ordinal. This algorithm does not contain parameters that require adjustment during training.

2.4 Recalculation of Classification Algorithm for Neighbor Samples

Denote $m_j = \left|\tilde{K}_j\right|, j = 1,2,...,l$. Consider the optimization task (1-4), when the classification of objects is carried out in a leave-one-out mode. The choice of change-points determines a partition of the training set for classification problems. Neighboring training samples (and the classification problems) can be obtained from

the original by deleting an object, or moving an object from one class to another. Efficiency of task (1-4) solution depends directly on quick recalculation of classification algorithm after transition to the neighbor sample.

In solving the problem (1-4) estimations $\widetilde{\Gamma}_j(\mathbf{x})$ are easily converted into a leave-one-out mode. Really, we calculate the matrices $\mathbf{D}^1 = \left\| D^1_{\alpha\gamma\beta} \right\|_{m\times m\times m}$, $D^1_{\alpha\gamma\beta} = \mathrm{C}^k_{d(\mathbf{x}_\alpha,\mathbf{x}_\gamma,\mathbf{x}_\beta)}$, $\mathbf{D}^2 = \left\| D^2_{\alpha\gamma\beta} \right\|_{m\times m\times m}$, $D^2_{\alpha\gamma\beta} = 2^{d(\mathbf{x}_\alpha,\mathbf{x}_\gamma,\mathbf{x}_\beta)} - 1$.

Let $\mathbf{x}_t$ is any object in the training sample, there is a current partition into classes $K_1, K_2, ..., K_l$, and $\mathbf{x}_t \in K_i$. Then in a leave-one-out mode $\widetilde{\Gamma}_i(\mathbf{x}_t) = \dfrac{2}{(m_i-1)(m_i-2)} \sum\limits_{\substack{\alpha<\beta:\alpha,\beta\neq t \\ \mathbf{x}_\alpha,\mathbf{x}_\beta\in K_i,}} D^h_{\alpha t\beta}$, $\widetilde{\Gamma}_j(\mathbf{x}_t) = \dfrac{2}{m_j(m_j-1)} \sum\limits_{\substack{\alpha<\beta: \\ \mathbf{x}_\alpha,\mathbf{x}_\beta\in K_j,}} D^h_{\alpha t\beta}, j \neq i \cdot$

Here $h \in \{1,2\}$. For simplicity, we omit here and further h in notations of $\widetilde{\Gamma}_j(\mathbf{x}_t)$. Consider the recalculation of estimations $\widetilde{\Gamma}_j(\mathbf{x}_t), j = 1,2,...,l$, during function (1) recalculation in arbitrary neighbor point on some step of optimization. In this case, the boundary between a pair of classes is changing as a result of the transfer of an object $\mathbf{x}_\tau$ from one class to the neighboring class. Denote the "new" classes as $K^*_1, K^*_2, ..., K^*_l$, and estimations for $\mathbf{x}_t$ through $\widetilde{\Gamma}^*_j(\mathbf{x}_t), j = 1,2,...,l$.

We have the following four possible variants:

1. $K^*_i = K_i \cup \{\mathbf{x}_\tau\}$, $\mathbf{x}_\tau \in K_u, u \neq i$,
   $K^*_u = K_u \setminus \{\mathbf{x}_\tau\}$, $K^*_j = K_j, j \neq i,u$.
2. $K^*_u = K_u \setminus \{\mathbf{x}_\tau\}$, $K^*_v = K_v \cup \{\mathbf{x}_\tau\}$, $K^*_j = K_j, j \neq u,v$, $u,v \neq i$.
3. $K^*_i = K_i \setminus \{\mathbf{x}_\tau\}, \tau \neq t$, $K^*_u = K_u \cup \{\mathbf{x}_\tau\}$, $K^*_j = K_j, j \neq i,u$.
4. $K^*_i = K_i \setminus \{\mathbf{x}_t\}$, $K^*_u = K_u \cup \{\mathbf{x}_t\}$, $K^*_j = K_j, j \neq i,u$.

Then estimations $\widetilde{\Gamma}^*_j(\mathbf{x}_t), j = 1,2,...,l$ are recalculated as follows:

1. $\widetilde{\Gamma}^*_i(\mathbf{x}_t) = \dfrac{2}{m_i(m_i-1)} \left(\dfrac{(m_i-1)(m_i-2)}{2} \widetilde{\Gamma}_i(\mathbf{x}_t) + \sum\limits_{\substack{\alpha:\alpha\neq t \\ \mathbf{x}_\alpha\in K_i, \\ \mathbf{x}_\tau\in K_u,u\neq i}} D^h_{\alpha t\tau} \right)$,

$\widetilde{\Gamma}^*_u(\mathbf{x}_t) = \dfrac{2}{(m_u-1)(m_u-2)} \left(\dfrac{m_u(m_u-1)}{2} \widetilde{\Gamma}_u(\mathbf{x}_t) - \sum\limits_{\substack{\alpha:\mathbf{x}_\alpha\in K_u, \\ \mathbf{x}_\tau\in K_u,u\neq i}} D^h_{\alpha t\tau} \right)$,

$$\tilde{\Gamma}_j^*(\mathbf{x}_t) = \tilde{\Gamma}_j(\mathbf{x}_t), j \neq i, u \quad .$$

$$2.\ \tilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u - 1)(m_u - 2)} \left(\frac{m_u(m_u - 1)}{2} \tilde{\Gamma}_u(\mathbf{x}_t) - \sum_{\substack{\alpha : \mathbf{x}_\alpha \in K_u, \\ \mathbf{x}_\tau \in K_u, u \neq i}} D_{\alpha t \tau}^h \right),$$

$$\tilde{\Gamma}_v^*(\mathbf{x}_t) = \frac{2}{(m_v + 1)m_v} \left(\frac{m_v(m_v - 1)}{2} \tilde{\Gamma}_v(\mathbf{x}_t) + \sum_{\substack{\alpha : \mathbf{x}_\alpha \in K_v, \\ \mathbf{x}_\tau \in K_u, u \neq i}} D_{\alpha t \tau}^h \right), u, v \neq i, \quad \tilde{\Gamma}_j^*(\mathbf{x}_t) = \tilde{\Gamma}_j(\mathbf{x}_t), j \neq u, v \cdot$$

$$3.\ \tilde{\Gamma}_i^*(\mathbf{x}_t) = \frac{2}{(m_i - 2)(m_i - 3)} \left(\frac{(m_i - 1)(m_i - 2)}{2} \tilde{\Gamma}_i(\mathbf{x}_t) - \sum_{\substack{\alpha : \mathbf{x}_\alpha \in K_i, \\ \mathbf{x}_\tau \in K_i, \tau \neq t}} D_{\alpha t \tau}^h \right),$$

$$\tilde{\Gamma}_u^*(\mathbf{x}_t) = \frac{2}{(m_u + 1)m_u} \left(\frac{m_u(m_u - 1)}{2} \tilde{\Gamma}_u(\mathbf{x}_t) + \sum_{\substack{\alpha : \mathbf{x}_\alpha \in K_u, \\ \mathbf{x}_\tau \in K_i, \tau \neq t}} D_{\alpha t \tau}^h \right),$$

$$\tilde{\Gamma}_j^*(\mathbf{x}_t) = \tilde{\Gamma}_j(\mathbf{x}_t), j \neq i, u \cdot$$

$$4.\ \tilde{\Gamma}_i^*(\mathbf{x}_t) = \tilde{\Gamma}_i(\mathbf{x}_t), i = 1,2,...,l.$$

Thus, the calculation of the function (1) in the next point of general optimization algorithm is carried out effectively. The complexity of one optimization step will be $O(m^2)$.

3 Illustrations and Experiments

To date, the set of initial experiments have been carried out that have confirmed practical use of suggested algorithm. In this model, we assumed at first that the number l of change-points (number of classes in classification) is fixed. For function (1) calculation, the leave-one-out mode is used. The optimal number of change-points and their positions are computed by solving the problem (1-4) with various values of l. Finally, the efficiency of built piecewise constant dependence was estimated in leave-one-out mode.

As an illustrative example, consider the function $y(x) = \sin(x) + x/3$ on the interval $[1:25]$. To create a training set has been used 100 points $x_i = 1 + i/4, i = 0,1,...,99$. Figure 1 shows the function $z(x)$ and optimal piecewise constant functions obtained with the ECA. Corresponding to the ECA values of the functions (1) are denoted by the symbols «−». The values of the mean modulus error were found to be 0.63 for the linear regression, and 0.19 for ECA with $l = 32$.

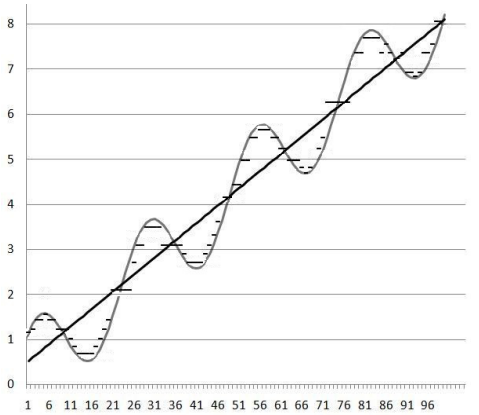


Fig. 1. Piece-wise constant function approximating $z(x) = \sin(x) + x/3$

Below is an example of constructing a piecewise constant function, according to data [10]. Considered the problem of automatic evaluation of the cost of housing in the suburbs of Boston. The training set consisted of 366 objects described in terms of 13 features (12 real and one binary). Figure 2 shows the distribution of housing costs in the axes of "number of the object - the cost of housing in thousands of dollars". Objects pre-ordered by increasing cost of housing. The real cost of housing is indicated by «▲». The calculated estimates of the cost of housing using the proposed in this work method are marked by symbols " _ ". Number of piecewise components found to be 32, the value of the mean modulus of error is 1.07. The average modulus error of linear regression was equal to 1.213.

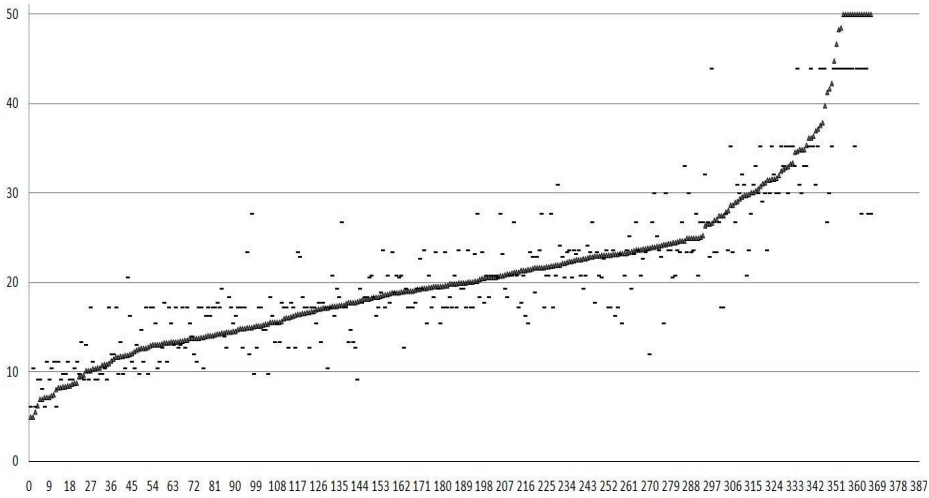


Fig. 2. Piece-wise constant function approximating the cost of housing in the suburbs of Boston

4 Conclusions

It is important to note the following details of this research.

1. Algorithms for reconstruction of the regression were considered for the case of numeric attributes $x_i \in R, i = 1,2,...,n$. It is easy to see that features may be dissimilar (numeric, binary, or ordinal) when we are constructing piecewise constant regressions by using the training samples. The proposed model of regressions restoration is based on the modification of the ECA, which does not require a metric in the feature space. The feature values are used only in (7) where the order relation for each feature is applied.
2. At the beginning of this paper, it was remarked some situations where the classical methods of regressions restoration are not applicable or restricted. It should be added the cases when the values of the dependent quantities are very unevenly shared, or dependent quantities are in fact the l-valued with large value of l. The most preferable case for supervised classification at a fixed length of the training sample is the case when number of classes is equal to 2. However, this case may not be optimal for the functional (7). Thus, it is expected that the model for constructing piecewise constant regressions will be helpful in addressing the many "bad " problems, inconvenient for standard regression approaches, and for classification tasks. In these cases, "bad" regression problem reduces to the problem of classification with an optimal choice of l.

Previously, we have considered the case $z = f(\mathbf{x}) = (a_i + a_{i-1})/2$. Of course, it can be used here other methods for $z = f(\mathbf{x})$ calculation by $a_1, a_2, ..., a_{l-1}$ (mean in I_i by training sample, median, etc.).

In future work a generalization of the approach is supposed to restore the piece-wise non-constant dependencies.

Acknowledgments. I would like to thank the postgraduate student of M.V.Lomonosov Moscow State University A.S. Schichko for experiments performed. This work was supported by RAS Presidium Program number 15, Program number 2 of Department of Mathematical Sciences of RAS, RFBR 12-01-00912, 11-01-00585, 12-01-90012-bel.

References

1. Draper, N., Smith, H.: Applied regression analysis. John Wiley & Sons, New York (1966)
2. Hardle, W.: Applied nonparametric regression. Cambridge University Press, Cambridge (1990)
3. Collobert, R., Bengio, S.: Support Vector Machines for Large-Scale Regression Problems. Journal of Machine Learning Research 1, 9/1/, 143–160 (2001)
4. Yu, J.-R., Tzeng, G.-H., Li, H.-L.: General fuzzy piecewise regression analysis with automatic change-point detection. Fuzzy Sets and Systems 119, 247–257 (2001)
5. Hutter, M.: Bayesian Regression of Piecewise Constant Functions. Technical Report IDSIA-14-05, Galleria 2, CH-6928 Manno-Lugano, Switzerland (2005)

6. Janssen, F., Fyurnkranz, J.: Heuristic Rule-Based Regression via Dynamic Reduction to Classification. In: Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence (IJCAI 2011), pp. 1330–1335 (2011)
7. Bibi, S., Tsoumakas, G., Stamelos, I., Vlahavas, I.: Regression via Classification applied on software defect estimation. Expert Systems with Applications 34, 2091–2101 (2008)
8. Binev, P., Cohen, A., Dahmen, W., DeVore, R., Temlyakov, V.: Universal Algorithms for Learning Theory. Part I: Piecewise Constant Functions. Journal of Machine Learning Research 6, 1297–1321 (2005)
9. Zhuravlev, Y.: Selected Scientific Publications, p. 420. M. Magistr Publishing (1998)
10. Harrison, D., Rubinfeld, D.L.: Hedonic prices and the demand for clean air. J. Environ. Economics & Management 5, 81–102 (1978)

Music Genre Classification: A Semi-supervised Approach

Soujanya Poria[1], Alexander Gelbukh[2], Amir Hussain[3],
Sivaji Bandyopadhyay[1], and Newton Howard[4]

[1] Computer Science and Engineering Department, Jadavpur University, India
soujanya.poria@ieee.org, sbandyopadhyay@cse.jdvu.ac.in
[2] CIC, Instituto Politécnico Nacional, 07738 DF, Mexico
www.Gelbukh.com
[3] Dept. of Computing Science and Mathematics, University of Stirling, United Kingdom,
ahu@cs.stir.ac.uk
[4] Brain Science Foundation
nhmit@mit.edu

Abstract. Music genres can be seen as categorical descriptions used to classify music basing on various characteristics such as instrumentation, pitch, rhythmic structure, and harmonic contents. Automatic music genre classification is important for music retrieval in large music collections on the web. We build a classifier that learns from very few labeled examples plus a large quantity of unlabeled data, and show that our methodology outperforms existing supervised and unsupervised approaches. We also identify salient features useful for music genre classification. We achieve 97.1% accuracy of 10-way classification on real-world audio collections.

1 Introduction

Downloading and purchasing music from online music collections has become part of the daily life of probably the majority of people in the world, and quality of music recommendation affects quality of life of billions of people. The users often formulate their preferences in terms of genre, such as jazz or disco. However, many tracks in existing collections are not classified by genre, or a genre is specified for an artist or an album but not for a particular track. Given huge size of existing collections, automatic genre classification is crucial for organization, search, retrieval, and recommendation of music.

While huge amount of unlabeled data is readily available, labeled data—tracks with the genre reliably assigned by human annotators—are scarce. In this paper we propose to use for genre classification a methodology that was proven to work well in a similar situation: affective labeling of words in natural language texts, where, similarly, unlabeled texts abound but few words have a manually assigned affective label [1]. For brevity we refer to this methodology as semi-supervised learning, to emphasize that it uses two kinds of data: few labeled examples and a large quantity of unlabeled data; however, internally our two-step procedure works differently from a

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 254–263, 2013.

typical semi-supervised learner. We show that this methodology outperforms a number of standard supervised learning techniques, such as Support Vector Machine (SVM) and k-Nearest Neighbor (kNN).

In addition, we present several features salient for the genre classification task. According to Lee *et al.* [3], musical features can be divided into three categories: short-time features, long-time features, and beat features. We show that long-term features are more salient for music genre classification than short-time ones, though using features of both types together gives best results. We explore various feature combinations and identify those that perform best on our task.

The rest of the paper is organized as follows: Section 2 describes related work. Section 3 gives an overview of our method. Section 4 presents the dataset and the features used in our experiments. Sections 5, 6, and 7 describe the three main steps of our algorithm: fuzzy clustering, mapping of the obtained clusters to labels, and the final hard categorization. Section 8 gives the experimental results and evaluation. Finally, Section 9 presents conclusions and future work.

2 Related Work

The state-of-the-art music genre classification systems can be classified into those based on supervised or unsupervised approach [4].

Unsupervised Approaches. These approaches mainly concern with determination of a genre taxonomy. Music files are clustered basing on an objective function to dynamically build a taxonomy depending on the clustering outcome. Shao *et al.* [5] used Agglomerative Hierarchical Clustering.

The main drawback of unsupervised methods is that the clusters are not labeled and the boundaries between clusters are not reliably defined. In this paper, we rely on a genre taxonomy well-defined by music experts and well-known to the users, which suggest using a supervised approach.

Supervised Approaches. Pampalk et al. [7] used a kNN classifier; Mandel et al. [10], Lidy et al. [11], and Scaringella et al. [12] obtained good results with SVM.

Semi-Supervised Approaches. Xu et al. [23] used co-training based semi-supervised classifier based on some novel "multi-view" [23] features. Yaslan et al. [24] stated how Random Subspace Method for Co-training [24] can help the genre classification process.

Our approach is a bit different from existing semi-supervised approaches in a manner that we used fuzzy for training supervised classifier in order to map a 10-way classification problem to 2-way or 3-way classification problem.

Feature Selection. Proper feature selection is crucial for classification. For this, a segment of audio is represented by numerical values of several audio features. Peeters [14] proposed a variety of features to characterize the timbre of instruments. These features are called low level features [15] because they usually describe sound on small scale, such as slices of 10 to 60 seconds. Spectral features have been used to

distinguish between speech and music [4] and to identify isolated sounds [16] and instruments [17]. Rauber et al. [6] used psychoacoustic features of music to determine similarities between music files. The importance of the size of texture window for extracting timbral features has been explored by Meng *et al*. [18], who stated that texture window of 1 sec. works best for music retrieval task and there is no significant gain in increasing the size of the texture window, while the accuracy decreases with smaller window size.

Tzanetakis [19] designed a basic music genre classification system based upon timbral, temporal, and beat features with 61% accuracy, outperformed by Lee *et al*. [3] with a spectral modulation-based approach.

While we work with a conventional genre classification, we explore the performance of different features and feature combinations, achieving 97.1% accuracy.

3 Overview of the Procedure

We followed a procedure suggested in [1] for a quite different task: effective classification of words [2]. The procedure consists in the following steps:

- Feature extraction: the real-world data, both labelled and unlabelled, are represented by numerical vectors, which then are used for classification.
- Fuzzy clustering: the whole available dataset, including both labelled and unlabelled data (the labels are ignored even when available), is clustered in unsupervised manner into the number c of clusters corresponding to the number of target categories (in our case, $c = 10$ music genres), in hope that the found clusters would roughly correspond to the target categories. Fuzziness accounts for uncertainty: a data point can be assigned more than one label, with a different degree. This ambiguity is resolved at the last step.
- Mapping: the obtained fuzzy clusters are one-to-one identified with the c target categories. The classes are identified through a majority voting technique, performed within each of the clusters. In our case we have all annotated music samples but for that situation when we have maximum number of unlabeled data and much lower number of labelled data we can still carry out our method by taking all of those unlabeled and labelled data for clustering and determine the fuzzy classes of the clusters through the majority voting with the help of available labelled data taking part in the clustering step.
- Hard clustering: the ambiguity of the fuzzy assignment of category labels to data items is resolved, leaving each data item assigned to exactly one category using a supervised technique.

4 Dataset and Features

As a dataset for the music genre classification task, we used the one presented by Tzanetakis [19]. The dataset is publicly available for research purposes.[1] It consists of

[1] `http://opihi.cs.uvic.ca/sound/genres.tar.gz`

1000 audio tracks, each being 30 sec. long, classified into 10 genres: BLUES, CLASSICAL, HIPHOP, COUNTRY, DISCO, POP, ROCK, JAZZ, METAL and REGGAE. Each genre is represented by 100 tracks. We followed this taxonomy for our classification.

All tracks are 22,050 Hz mono 16-bit audio files in .wav format, collected in 2000–2001 from a variety of sources including personal CDs, radio, and microphone recordings, in order for a variety of recording conditions to be represented.

For feature extraction, we used the Jaudio toolkit [20], a music feature extraction toolkit written in Java, freely available for research purposes.[2] As we have mentioned, we used the following three kinds of features: short-time features, long-time features, and beat features.

Short-time features are mainly used to distinguish the timbral characteristics of music and are usually extracted independently from each short time window (frame) during which the audio signal is assumed to be stationary. We used the following features [3, 20]: mel-frequency cepstral coefficients (MFCC; Jaudio gives first five coefficients of 13, and [3] states that these five coefficients give best classification result), spectral centroid, spectral roll-off, spectral flux, root mean square, compactness, and time domain zero crossing;

Long-time features can be obtained by aggregating the short-term features extracted from several consecutive frames within a time window. We have used derivate, standard deviation, running mean, derivative of running mean, and standard derivative of running mean as the aggregation methods of short-time features.

Beat features give meanings to audio signals in human-recognizable terms which generally reveal the human interpretation or perception of certain audio properties such as mood, emotion, tempo, genre, etc. We used the following four main beat features: beat histogram, beats per minute, beat sum, and strongest beat in the audio signal.

5 Fuzzy Clustering

The first step in our process is unsupervised: we cluster the music files into 10 categories, given that we consider 10 genres. On output, we define for each music file and each of the ten classes the membership value between 0 and 1 with which the given music file belong to the given class.

Fuzzy C-means Clustering Algorithm. For fuzzy clustering, we used the fuzzy c-means clustering algorithm [21] with a modified objective function as described in Section 5.2 below.

The well-known fuzzy c-means clustering algorithm takes as input a set of N data points $x_1, x_1, ..., x_N$ described via their coordinates in a P-dimensional feature space: $x_k = (x_{k1}, x_{k2},..., x_{kP})$. As output, it constructs two sets: a set of c centroids $v_1, v_2, ..., v_c$, points in the same feature space, that represent the constructed c clusters, and a set of cN membership values μ_{ik}, $i = 1, ..., c$; $k = 1, ..., N$, which represent the degree of membership of a point x_k in a class c_i, such that $0 \le \mu_{ik} \le 1$ and the values sum up to a unity for each point:

[2] http://sourceforge.net/projects/jmir/files/

$$\sum_{i=1}^{c} \mu_{ik} = 1, \quad k = 1, \ldots, N. \tag{1}$$

To find the optimal distribution of points by clusters and optimal placement of the centroids, it uses an given objective function J, which is minimized when the distribution is optimal: $(\mu_0, v_0) = \arg\min J(\mu, v)$, where $\mu = \{\mu_{ik}\}$ and $v = \{v_i\}$ represent the sets of the variables to be found and μ_0, v_0 are the optimal solutions. An expression often used for J is

$$J_p(\mu, v) = \sum_{i=1}^{c} \sum_{k=1}^{N} \mu_{ik}^p \left\| x_k - v_i \right\|^2, \quad \left\| x - y \right\|^2 = \sum_{l=1}^{P} \left(x_l - y_l \right)^2 \tag{2}$$

where the power $p > 1$ is a given parameter that controls the degree of fuzziness of the obtained clusters (we used $p = 2$). The optimal solution of a constraint optimization problem defined by (1) and (2) is given [1] by

$$v_i = \frac{\sum_{k=1}^{N} \mu_{ik}^p x_k}{\sum_{k=1}^{N} \mu_{ik}^p}, \quad \mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\left\| x_k - v_i \right\|}{\left\| x_k - v_j \right\|} \right)^{\frac{2}{p-1}}} \tag{3}$$

A stationary point (μ_0, v_0) of the system (3), which is the desired result of the algorithm, was found iteratively:

- Assigning random values to all μ_{ik}, normalized to satisfy the constraints (1);
- Iteratively re-calculate the values for all v_i and then all μ_{ik} according to (3);
- Stop when the objective function J changes from the previous iteration less than by a small number ε, a given parameter (we used $\varepsilon = 0.01$).

Modified Objective Function. To achieve more compact clusters in which the most similar elements are clustered together, we incorporated an additional term in the original objective function (2):

$$J_p(\mu, v) = \sum_{i=1}^{c} \sum_{k=1}^{N} \mu_{ik}^p \left(\left\| x_k - v_i \right\|^2 + \rho \sum_{x \in N_k} \left\| x - v_i \right\|^2 \right), \tag{4}$$

where the parameter ρ is intended to control the effect of the new term (we used $\rho = 1$) and N_k is the set constructed in the following way:

- For each data point x, we identified the nearest centroid

$$v(x) = \arg\min_i \| v_i - x \| \tag{5}$$

  (in case of a tie an arbitrary one was chosen);
- Now, $N_k = \{x \mid v(x) = v(x_k)\}$ is the set of all data points with the same nearest centroid as x_k.

This additional term forces the algorithm to increase the membership of a data point in the cluster with the nearest centroid, grouping similar points together.

In our implementation we constructed these sets on the fly while re-calculating the positions of the centroids according to (6) below, which is a modification of (3). I.e., when re-calculating v_2, we considered in (5) already re-calculated value for v_1.

The change of the objective function required modification of the formulas (3):

$$v_i = \frac{\sum_{k=1}^{N} \mu_{ik}^{p} \left(x_k + \rho \sum_{x \in N_k} x \right)}{\sum_{k=1}^{N} \mu_{ik}^{p} \left(1 + \rho |N_k|\right)}, \quad \mu_{ik} = \frac{1}{\sum_{j=1}^{c} \left(\frac{\|x_k - v_i\|^2 + \rho \sum_{x \in N_k} \|x - v_i\|^2}{\|x_k - v_j\|^2 + \rho \sum_{x \in N_k} \|x - v_j\|^2} \right)^{\frac{1}{p-1}}}; \tag{6}$$

the derivation can be found in [1].

6 Mapping Fuzzy Classes to Music Genre Labels

After the fuzzy clustering process had been completed, we identified which one of the $c = 10$ classes corresponded to which one of the ten music genre labels. For this, first we converted the fuzzy clustering into hard clustering; in our implementation we chose for each data point x_k a cluster $C(x_k) = \arg\max_i \mu_{ik}$ (in case of a tie, arbitrary class was chosen). Now, the music genre label for each hard cluster was chosen by majority voting.

Such procedure does not guarantee for a hard cluster to be non-empty, for the majority voting not to result in a tie, or for two clusters not to share the same genre label, in which case some labels would not be assigned at all. However, this is low probable and did not happen in our experiments. Moreover, correctness of the obtained mapping of the classes to genre labels is confirmed by the fact that we obtained over 90% accuracy of the final results, which is not possible with incorrectly mapped labels.

7 Hard Clustering

In our evaluation, we consider a label to be assigned correctly if the evaluation dataset assigns this label to the music file. To choose only one class for a token under classification, we used a two-step process.

Reducing the Confusion Set. For each data point, we chose K classes for which the fuzzy clustering gave the highest value of the membership function. The hard clustering technique used afterwards was only allowed to choose between those K labels preselected for a given music file.

In case of $K = 1$ no further processing is needed and the final result is determined by the greatest membership value of the fuzzy clustering. The case of $K = 10$ means no reducing of the confusion set. In case of $K = 2$ or $K = 3$—the values we experimented with—the confusion set is reduced to 2 or 3 options, correspondingly. We show in Section 8 that reducing the confusion set to 2 candidates increased the accuracy. However, selection of the proper size of confusion set depends to the problem: on another task a confusion set of, say, 3 might result in better accuracy.

Final Hard Categorization. Given the K options left after reducing the confusion set, we trained different classifiers for each of the $\binom{10}{K} = \frac{c!}{K!(c-K)!}$, $c = 10$, possible combinations of K genre labels: for example, with $K = 3$, a separate classifier was trained for choosing between BLUE, METAL, and COUNTRY, another one to choose between JAZZ, METAL, and ROCK, etc. For $K = 3$, therefore, 120 different classifiers were trained; for $K = 2$, 45 different classifiers were trained. To assign a label to data point, the K genre labels for the point are selected as explained above, and then the corresponding classifier is used.

For training, we have taken care of all training music files in ten genre lists of our dataset, i.e., we used 60 music files of each 10 genres lists for training. For example, to train a classifier for the confusion set {BLUE, METAL}, all music files extracted from the lists and have their either the label BLUE or the label METAL were used.

As features, we used the same feature vectors as for fuzzy clustering, extended by 10 extra dimensions: the membership values generated by the fuzzy classifier for the 10 genre labels, except the experiments where the fuzzy clustering was not used. As classes, the K selected labels were used for each classifier; in case of $K = 2$ the classification was binary, for $K = 3$ the classification is ternary.

As a hard clustering algorithm, we used the SVM framework. Specifically, we used the `libsvm` library of the WEKA toolset [22], which, for the case of $K > 2$, provides an implementation of a multiclass SVM. As a result, we obtained one music genre label for each music file in the test dataset.

8 Evaluation

Impact of Different Feature Combinations. Table 1 shows that we obtained better accuracy when we used long-time features than using short-time features. However, a much lower accuracy was obtained when we used only beat features. The highest accuracy was obtained when we used all three types of features: long-time, short-time, and beat features.

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

Feature Combination	Fuzzy	SVM	Fuzzy + SVM
Long-time features	59.12%	61.20%	63.25%
Short-time features	42.54%	44.15%	48.92%
Long-time + short-time features	68.21%	71.24%	75.34%
Beat features	39.15%	39.46%	41.27%
Long-time + short-time + semantic	76.33%	87.45%	96.23%
Long-time + beat features	68.67%	72.35%	76.25%
Long + short + semantic + fuzzy vector	79.21%	—	**97.10%**

We have done the evaluation in two ways. In one experiment we performed tenfold cross-validation on each of the 120 and 45 classifiers mentioned in Section 7.2 using all 1000 music files. The result and the corresponding confusion sets are given in Section 8. Using all three feature sets along with fuzzy membership vector as a feature, we obtained 97.10% accuracy.

Table 2. Impact of the selection of most likely fuzzy cluster

K	Accuracy
1	76.33%
2	**97.10%**
3	79.38%
4	77.51%
10	67.45%

Table 3. Accuracy obtained using different classifiers

Classifier	Accuracy
KNN	54.21%
Naïve Bayes	65.88%
MLP	74.23%
Our procedure	**97.10%**

In another experiment we split our dataset into 60% training and 40% test data. Using 60% training data we trained our 60 or 45 classifiers correspondingly, depending on the value of K, and tested them on the unseen test data. With this, we obtained 91.50% accuracy, which is probably explained by smaller size of the training data.

In particular, we observed that spectral centroid and MFCC are the most important features, because removing these two features significantly decreases accuracy.

Impact of the Fuzzy Clustering and Hard Categorization. In addition to the data presented in Table 1 for fuzzy-only and hard-only classifiers, we experimented with different values of *K*: the size of the confusion set after reduction based on the result of fuzzy clustering; see Table 2:

- $K = 1$ means that the final classification is made basing on the results of the fuzzy clustering and no further hard clustering is necessary;
- $K = 2$ means that the hard classification has to do only binary choices;
- $K = 3$ reduces the confusion set for the hard classification to three options;
- $K = 10$ means no reduction of the confusion set. It is not the same as not to use the fuzzy clustering phase at all, because the fuzzy clustering results are still used as additional features for final categorization.

We can see that SVM performed better on choosing between the category with the highest membership value and that of the second highest one. Here, we used all features, which corresponds to the last row of Table 1.

Table 4. Confusion Matrices

	Tenfold cross-validation										60% training / 40% test split									
Classified as	a	b	c	d	e	f	g	h	i	j	a	b	c	d	e	f	g	h	i	j
a. blues	100	0	0	0	0	0	0	0	0	0	38	0	0	0	1	0	0	0	0	1
b. country	0	98	0	0	0	0	0	0	0	2	0	37	0	0	0	0	0	0	0	3
c. classical	1	0	99	0	0	0	0	0	0	0	0	0	40	0	0	0	0	0	0	0
d. disco	1	0	1	94	4	0	0	0	0	0	1	0	1	34	4	0	0	0	0	0
e. hiphop	1	0	0	5	93	0	0	0	1	0	1	0	0	3	35	0	0	1	0	0
f. jazz	0	0	0	0	0	100	0	0	0	0	0	0	0	5	0	34	0	1	0	0
g. metal	0	0	0	0	0	0	100	0	0	0	0	0	1	0	0	0	39	0	0	0
h. pop	0	0	0	0	0	0	3	95	0	2	0	0	1	0	0	0	2	36	0	1
i. reggae	0	3	0	0	0	0	0	0	97	0	0	3	0	0	0	0	0	0	37	0
j. rock	0	4	0	0	0	0	0	1	0	95	0	2	0	0	0	1	0	1	0	36

Comparing with other Classifiers. We tried several classifiers, such as Multi-Layer Perception (MLP), Naïve Bayes, and kNN. While MLP performed better than Naïve Bayes and kNN, none of them outperformed our two-stage procedure; see Table 3.

Confusion Matrices. The confusion matrices obtained with our procedure are shown in Table 4. We can observe that misclassification problems are very rare, and quite similar in both cases.

9 Conclusions and Future Work

We have proposed a method of music genre classification in a large music dataset using a two-stage classification methodology. The methodology consists in fuzzy clustering followed by disambiguation using a hard classifier. As features of musical data, we used the values obtained with the Jaudio toolkit.

Soon, we plan to expand the classification tool on a different level by taking into account also lyrics associated with music tracks. In particular, we will extract conceptual and affective information associated with songs by means of semantic multidimensional scaling [25] and, hence, add these as additional semantic and sentic [26] features for classification.

Acknowledgements. The work was partly supported by the Governments of Mexico and India via the project "Answer Validation through Textual Entailment", CONACYT 122030 / Mexico-India 2011–2014; the Government of Mexico via the Instituto Politécnico Nacional, SIP 20131702 and CONACYT 50206-H; and the Government of UK via UK EPSRC grant ref: EP/G501750/1 "Common sense Computing to enable the development of next generation Sematic Web Applications".

References

1. Poria, S., Gelbukh, A., Cambria, E., Das, D., Bandyopadhyay, S.: Enriching SenticNet polarity scores through semi-supervised fuzzy clustering. IEEE ICDM, Brussels, 709–716 (2012)
2. Poria, S., Gelbukh, A., Hussain, A., Das, D., Bandyopadhyay, S.: Enhanced SenticNet with affective labels for concept-based opinion mining. IEEE Intelligent Systems (2013), doi:10.1109/MIS.2013.4
3. Lee, C., Shih, J., Yu, K., Lin, H.: Automatic music genre classification based on modulation spectral analysis of spectral and cepstral features. IEEE Transactions on Multimedia 11(4) (2009)
4. Scaringella, N., Zoia, G.: Automatic genre classification of music content: a survey. Signal Processing Magazine 23(2), 133–141 (2006)
5. Shao, X., Xu, C., Kankanhalli, M.: Unsupervised classification of musical genre using hidden Markov model. In: IEEE Int. Conf. of Multimedia Explore (ICME), Taiwan (2004)
6. Rauber, A., Pampalk, E., Merkl, D.: Using psycho-acoustic models and self-organizing maps to create a hierarchical structuring of music by sound similarity. In: 3rd Int. Conf. on Music Information Retrieval, France (2002)

7. Pampalk, E., Flexer, A., Widmer, G.: Improvements of audio based music similarity and genre classification? In: 6th Int. Symposium on Music Information Retrieval, UK, (2005)
8. Scaringella, N., Zoia, G.: On the modeling of time information for automatic genre recognition systems in audio signals. In: 6th Int. Symposium on Music Information Retrieval, UK (2005)
9. Soltau, H., Schultz, T., Westphal, M., Waibel, A.: Recognition of music types. In: IEEE Int. Conf. on Acoustics, Speech and Signal Processing (ICASSP), USA (1998)
10. Mandel, M., Ellis, D.: Song-level features and support vector machines for music classification. In: 6th Int. Symposium on Music Information Retrieval, UK (2005)
11. Lidy, T., Rauber, A.: Evaluation of feature extractors and psycho-acoustic transformations for music genre classification. In: 6th Int. Symposium on Music Information Retrieval, UK, (2005)
12. Scaringella, N., Mlynek, D.: A mixture of support vector machines for audio classification. Music Information Retrieval Evaluation exchange (MIREX) (2005), website, http://www.music-ir.org/evaluation/mirexresults/articles/audio_genre/scaringella.pdf
13. Berenzweig, A., Ellis, D., Lawrence, S.: Using voice segments to improve artist classification of music. In: AES 22nd International Conference on Virtual, Synthetic and Entertainment Audio (2002)
14. Peeters, G.: A large set of audio features for sound description (similarity and classification) in the CUIDADO project. CUIDADO I.S.T. Project Report (2004)
15. Saunders, J.: Real time discrimination of broadcast speech/music. In: Int. Conf. Acoustics, Speech, Signal Processing (ICASSP), pp. 993–996 (1996)
16. Wold, E., Blum, T., Keislar, D., Wheaton, J.: Content-based classification,search, and retrieval of audio. IEEE Multimedia 3(2) (1996)
17. MPEG-7, "Information Technology – Multimedia Content Description Interface – Part 4: Audio", ISO/IEC JTC 1/SC29, ISO/IEC FDIS 15938-4:2002 (2002)
18. Meng, A., Ahrendt, P., Larsen, J.: Improving music genre classification by short-time feature integration. In: 6th Int. Symposium on Music Information Retrieval, UK (2005)
19. Tzanetakis, G.: Music Genre Classification of Audio Signal. IEEE Transactions on Speech and Audio Processing 10(5) (2002)
20. McEnnis, D., McKay, C., Fujinaga, I., Depalle, P.: Jaudio: A Feature Extraction Library. In: ISMIR 2005(2005)
21. Bezdek, J.C.: Pattern Recognition with Fuzzy Objective Function Algoritms. Plenum Press, New York (1981)
22. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: An update. SIGKDD Explorations 11(1) (2009)
23. Xu, Y., Zhang, C., Yang, C.: Semi-supervised classification of musical genre using multiview features. In: International Computer Music Conference ICMC (2005)
24. Yaslan, Y., Zehra, C.: Audio genre classification with semi-supervised feature ensemble learning. In: 2nd International Workshop on Machine Learning and Music (2009)
25. Cambria, E., Song, Y., Wang, H., Howard, N.: Semantic multi-dimensional scaling for open-domain sentiment analysis. IEEE Intelligent Systems (2013), doi:10.1109/MIS.2012.118
26. Cambria, E., Hussain, A.: Sentic Computing: Techniques, Tools, and Applications, pp. 978–994. Springer, Dordrecht (2012) ISBN: 978-94-007-5069-2

An Empirical Study of Oversampling and Undersampling Methods for LCMine an Emerging Pattern Based Classifier

Octavio Loyola-González[1,2], Milton García-Borroto[1], Miguel Angel Medina-Pérez[2], José Fco. Martínez-Trinidad[2], Jesús Ariel Carrasco-Ochoa[2], and Guillermo De Ita[3]

[1] Centro de Bioplantas, Universidad de Ciego de Ávila. Carretera a Morón km 9, Ciego de Ávila, Cuba, C.P. 69450
{octavioloyola,mil}@bioplantas.cu
[2] Instituto Nacional de Astrofísica, Óptica y Electrónica. Luis Enrique Erro No. 1, Sta. María Tonanzintla, Puebla, México, C.P. 72840
{migue,fmartine,ariel}@ccc.inaoep.mx
[3] Benemérita Universidad Autónoma de Puebla, Faculty of Computer Science. Av. San Claudio y 14 sur, Puebla, México
deita@cs.buap.mx

Abstract. Classifiers based on emerging patterns are usually more understandable for humans than those based on more complex mathematical models. However, most of the classifiers based on emerging patterns get low accuracy in those problems with imbalanced databases. This problem has been tackled through oversampling or undersampling methods, nevertheless, to the best of our knowledge these methods have not been tested for classifiers based on emerging patterns. Therefore, in this paper, we present an empirical study about the use of oversampling and undersampling methods to improve the accuracy of a classifier based on emerging patterns. We apply the most popular oversampling and undersampling methods over 30 databases from the UCI Repository of Machine Learning. Our experimental results show that using oversampling and undersampling methods significantly improves the accuracy of the classifier for the minority class.

Keywords: supervised classification, emerging patterns, imbalanced databases, oversampling, undersampling.

1 Introduction

Supervised classification is a branch of Pattern Recognition that finds relations between unseen objects and a set of objects previously classified, in order to predict the class of those unseen objects. Due to the high diversity in pattern recognition problems, there is a large collection of techniques (classifiers) to find out these relations. Commonly, for a given problem, the user has to test different classifiers to select the most accurate. Nevertheless, for many learning tasks [12],

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 264–273, 2013.

a high accuracy is not the only goal; the result of the classifier should also be understandable by humans [11].

An important family of both understandable and accurate classifiers is the one based on emerging patterns [7]. A pattern is an expression, defined in a language, which describes a collection of objects [9]. An emerging pattern is a pattern that frequently appears in objects of a single class, but it barely appears in objects belonging to other classes. This way, emerging patterns can be used to predict the class of unseen objects. Classifiers based on emerging patterns are valuable tools that have been used to solve real-world problems in fields like Bioinformatics, streaming data analysis, intruder detection, human activity recognition, anomaly detection in network connection data, rare event forecasting, and privacy preserving data mining; among others [12].

Like most classifiers, those based on emerging patterns do not have a good behavior when they are trained with imbalanced datasets, where objects are not equally distributed into the classes, and therefore, classifiers get results which are biased by the class with more objects. These classifiers generate many emerging patterns for the majority class and only a few patterns (or none) for the minority class. This fact leads to low accuracy for the minority class. Imbalanced databases often appear in fields like finance [2], biology and medicine [15].

Currently, applying oversampling or undersampling methods [1,8,4] is the most common approach to deal with databases containing imbalanced classes. However, to the best of our knowledge, there is not any study about the impact of these methods for emerging pattern based classifiers.

In this paper, we present a study of applying oversampling and undersampling methods for an emerging pattern based classifier, over 30 imbalanced databases. We show that the accuracy is significantly improved (according to the Friedman test [6] and the Bergmann-Hommel dynamic post-hoc procedure [10]) for the minority class.

The rest of the paper has the following structure. Section 2 provides a brief introduction to emerging patterns. Section 3 reviews the most popular oversampling and undersampling methods. Section 4 presents the empirical study developed with the methods presented in section 3, it includes a description of the setup, the way we evaluate the results and some concluding remarks that arise from these results. Finally, section 5 provides conclusions and future work.

2 Emerging Patterns

A *pattern* is an expression, defined in a language, which describes a collection of objects. The objects described, or covered, by a pattern are named the pattern *support*. In a supervised classification problem, we say that a pattern is an *emerging pattern* if its support increases significantly from one class to the others [12]. Emerging patterns are usually expressed as combinations of feature values, like $(Eyes = blue, Sex = male, Age = 30)$ or as logical properties, for example $[Eyes = blue] \wedge [Sex = male] \wedge [Age > 30]$.

Extracting emerging patterns from a training sample is a challenge because the number of candidates grows exponentially with respect to the number of features. Moreover, the downward closure property, one of the most effective pruning strategies, does not hold for emerging patterns [12].

In the literature, there are several algorithms for mining emerging patterns. Special attention deserve those algorithms based on decision trees, which usually do not find all the emerging patterns, but obtain a good collection of high quality patterns [12]. In this paper, we use LCMine [11] because it is an efficient algorithm for finding discriminative regularities (patterns) for supervised classification in problems with mixed and incomplete data. LCMine induces diverse decision trees, extracts patterns from these trees, and in a filtering post-processing stage, LCMine finds a reduced set of high quality discriminative properties (emerging patterns) for each class. In [11] the authors propose a classifier (LCMine classifier), which uses several strategies to avoid over-fitting [11].

As far as we know, this is the first paper that studies the use of oversampling and undersampling methods for a classifier based on emerging patterns (LCMine classifier) in order to solve the imbalance in databases.

3 Oversampling and Undersampling Methods

Most supervised classifiers work with databases containing balanced classes. However, there are application domains that contain high imbalance among classes. Imbalanced classes bias the classifiers which tend to classify all objects into the majority class. One way to deal with this problem is applying oversampling and undersampling methods.

Oversampling methods increase the amount of objects in the minority class in order to balance the classes. On the contrary, undersampling methods adjust the class distribution by removing objects from the majority class. In the literature there are also hybrid methods which combine oversampling and undersampling. However, there is no consensus in the literature about what type of method is the best [5].

In this paper, we perform several experiments to study the impact of oversampling and undersampling methods on the LCMine classifier. The following are the methods that we use in our study:

1. Spread Subsample: This undersampling method generates a random subsample of a database. This method adjusts the class distribution through a random elimination of objects from the majority class. This distribution is computed in dependence of a *Spread* value determined by the user. The *Spread* value (a parameter) specifies the maximum ratio between the majority and minority classes.
2. Synthetic Minority Over-sampling Technique (SMOTE) [4]: This is an oversampling method that generates synthetic objects based on the nearest neighbor of each sample in the minority class. Synthetic samples are generated computing the difference between the feature vector (sample) under

consideration and its nearest neighbor, then this difference is multiplied by a random number between 0 and 1, and the result is added to the feature vector under consideration. This causes the selection of a random point along the line segment between two specific features values. This approach effectively forces the decision region of the minority class to become more general.

3. SMOTE_NEW: This method is similar to SMOTE but it determines dynamically for each class the percent of new objects to be generated depending on the ratio between classes. Since this percent depends on the ratio between classes, the higher the imbalance, the higher will be the percent to be used. In short, the goal of this algorithm is to achieve uniform distribution among classes, increasing the amount of objects in the minority class but without exceeding the majority class as occurs in SMOTE.
4. Resample: This is a hybrid method that produces a random subsample of the majority class and applies oversampling the minority class in order to obtain uniform class distribution. This method can use sampling with replacement or without replacement. The parameter B specifies the level of balance between classes; values close to one will produce more uniform class distribution.

4 Experimental Results

This section presents the empirical study developed in this paper.

4.1 Experimental Setup

For our experiments, we use 30 databases taken from the UCI Repository of Machine Learning [3]. Table 1 describes the used databases. These databases have different characteristics according to size, class distribution, feature types and percentage of objects with missing values.

Similar to other authors [17,14] we modify the databases *hypothyroid_M*, *page-blocks_M* and *postoperative_M*. In these databases, we merge into a single class (named minority class) those objects belonging to the complement of the majority class. The *iris_M* database is a modification of the original iris database where we join the two classes with higher overlapping.

For each database and each oversampling and undersampling method, we perform 10 fold cross validation averaging the classification accuracy for the minority and majority classes.

For our experiments we use the Friedman test [6] and the Bergmann-Hommel dynamic post-hoc procedure [10] to compare the accuracy results. We also use CD diagrams to show the post-hoc results. In a CD diagram, the top line is the axis where the average rank of the classifiers is plotted, the rightmost classifier is the best classifier, while two classifiers sharing a thick line have statistically similar behavior [6].

We use the implementations of Resample, Spread Subsample and SMOTE taken from Weka [13]. We modify the parameter values, as it is shown in the Table 2, to ensure an uniform distribution of classes.

Table 1. Databases used in the experiments. #Obj: number of objects; Class Distrib: objects per class; #Features: number of features; Missing values: percentage of objects with missing values; Ratio: the ratio between the majority class and its complement.

			# Features			
Database	#Obj	Class Distrib (%)	Numeric	Non-Numeric	Missing values	Ratio
sick	3772	6/94	7	22	5.54%	15.3
hypothyroid_M	3772	8/92	7	22	5.54%	12.0
page-blocks_M	5473	10/90	10	0	-	8.8
wdbc	569	37/63	30	0	-	3.2
haberman	306	26/74	2	1	-	2.8
postoperative_M	90	30/70	0	8	$< 1\%$	2.5
breast-cancer	286	30/70	0	9	$< 1\%$	2.4
credit-g	1000	30/70	7	13	-	2.3
iris_M	150	34/76	4	0	-	2.0
breast-w	699	35/65	9	0	$< 1\%$	1.9
tic-tac-toe	958	35/65	0	9	-	1.9
diabetes	768	35/65	8	0	-	1.9
labor	57	35/65	8	8	35.75%	1.9
ionosphere	351	36/64	34	0	-	1.8
heart-h	294	36/64	6	7	20.46%	1.8
colic	368	37/63	7	15	23.80%	1.7
colic.ORIG	368	37/63	7	20	19.39%	1.7
wpbc	198	24/76	33	0	$< 1\%$	1.7
vote	435	39/61	0	16	5.63%	1.6
spambase	4601	39/61	57	0	-	1.5
shuttle-landing	15	40/60	0	6	28.89%	1.5
liver-disorders	345	42/58	6	0	-	1.4
cylinder-bands	540	43/57	18	21	4.74%	1.4
heart-statlog	270	44/56	13	0	-	1.3
credit-a	690	45/55	6	9	$< 1\%$	1.2
crx	690	45/55	6	9	$< 1\%$	1.2
cleveland	303	46/54	6	7	$< 1\%$	1.2
sonar	208	46/54	60	0	-	1.1
kr-vs-kp	3196	48/52	0	36	-	1.1
mushroom	8124	48/52	0	22	1.39%	1.1

Table 2. Description of the oversampling an undersampling methods and the parameters in our experiments

Path in Weka	Parameters
weka.filters.supervised.instance.Resample	-B 1.0 -S 1 -Z 100.0
weka.filters.supervised.instance.SpreadSubsample	-M 1.2 -X 0.0 -S 1
weka.filters.supervised.instance.SMOTE	-C 0 -K 5 -P 100.0 -S 1

4.2 Accuracy Analysis

In this section, we analyze the global accuracy and the accuracy in the minority and majority classes obtained by oversampling and undersampling methods over the tested databases. We also include the plain results for LCMine.

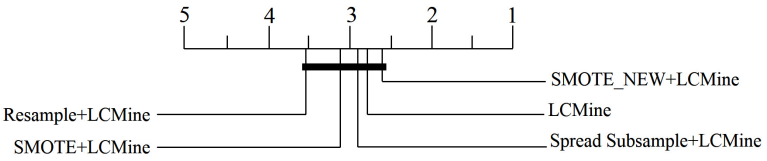


Fig. 1. CD diagram with a statistical comparison of the global accuracy of the LCMine classifier before and after using oversampling and undersampling methods over all the tested databases

Figure 1 shows that the global accuracy of SMOTE_NEW+LCMine classifier is the best. Nevertheless, there is not significant statistical difference among using or not using oversampling or undersampling methods.

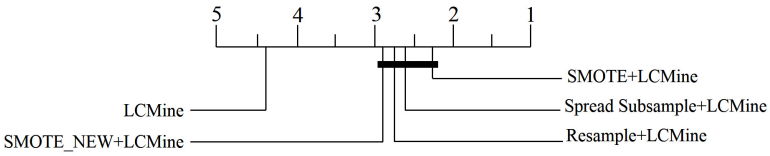


Fig. 2. CD diagram with a statistical comparison of the accuracy in the minority class of the LCMine classifier before and after using the oversampling and undersampling methods over all the tested databases

Figure 2 shows that applying oversampling or undersampling methods improves the accuracy of the LCMine classifier in the minority class. SMOTE+LCMine achieves the best results, nevertheless, notice that there is not statistical significant difference among the different results obtained by oversampling and undersampling methods.

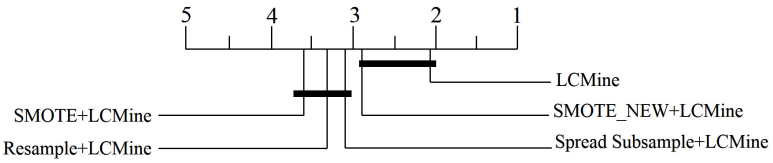


Fig. 3. CD diagram with a statistical comparison of the accuracy in the majority class of the LCMine classifier before and after using the oversampling and undersampling methods over all the tested databases

Figure 3, we can see that LCMine classifier obtained the best results in the majority class. Nevertheless, notice that there is not significant statistical difference between the results of LCMine and SMOTE_NEW+LCMine .

Table 3. Accuracy results of the compared oversampling and undersampling methods on the tested databases. We show the accuracy for the minority (*min*) and majority (*maj*) classes. The best results for each database in the minority and majority classes appear bolded.

Database	Resample		SMOTE NEW		SMOTE		No resampling		Spread Subsample	
	min	*maj*	*min*	*maj*	*min*	*maj*	*min*	*maj*	*min*	*maj*
sick	87.01	73.85	83.98	**75.01**	82.68	71.14	83.55	67.27	**94.37**	72.86
hypothyroid_M	84.88	**88.25**	31.62	68.86	95.53	75.98	86.94	14.22	**98.63**	87.04
page-blocks_M	84.11	**97.44**	54.11	83.72	84.82	91.33	83.21	84.86	**93.93**	95.91
wdbc	**72.34**	64.24	59.57	72.19	48.94	81.46	34.04	**93.38**	70.21	62.91
haberman	58.02	69.78	38.27	79.56	41.98	80.44	28.40	**83.11**	**59.26**	67.11
postoperative_M	26.92	53.13	7.69	65.63	7.69	64.06	3.85	**84.38**	**38.46**	64.06
breast-cancer	**57.65**	65.67	40.00	77.61	41.18	78.61	34.12	**86.57**	56.47	65.17
credit-g	**66.33**	71.29	54.67	84.43	53.00	83.43	41.00	**90.29**	65.67	74.57
iris_M	**100**	99.00	**100**	**100**	**100**	**100**	96.00	**100**	**100**	99.00
breast-w	**93.36**	96.51	92.95	96.51	92.95	96.29	92.53	**97.16**	92.53	96.51
tic-tac-toe	94.88	96.49	**96.08**	99.52	94.88	99.52	92.77	**100**	95.78	99.20
diabetes	74.63	75.00	70.90	76.20	73.13	75.00	59.33	**83.60**	**74.63**	76.60
labor	85.00	62.16	90.00	70.27	**100**	59.46	80.00	**78.38**	90.00	67.57
ionosphere	82.54	96.89	**83.33**	97.78	80.95	96.44	76.98	**99.11**	76.98	97.78
heart-h	86.79	69.15	86.79	62.23	**87.74**	52.13	76.42	**81.38**	84.91	70.21
colic	71.32	88.79	77.21	86.21	**80.15**	82.33	72.06	**90.95**	74.26	87.93
colic.ORIG	75.74	87.93	**76.47**	86.64	72.79	86.64	69.12	**92.24**	75.74	88.79
wpbc	91.98	96.92	**93.87**	96.08	93.40	95.24	91.51	**97.48**	93.40	96.36
vote	**94.05**	92.88	91.07	**94.01**	92.86	**94.01**	89.88	94.01	91.67	93.26
spambase	**93.33**	**90.32**	78.27	83.21	47.10	83.39	92.83	78.08	91.06	81.71
shuttle-landing	0.00	77.78	0.00	**100**	**50.00**	77.78	0.00	**100**	0.00	**100**
liver-disorders	60.69	78.00	60.69	78.50	**68.28**	68.00	60.00	**80.50**	62.76	76.00
cylinder-bands	44.30	78.53	49.56	78.53	**54.39**	73.08	32.46	**85.58**	42.11	80.77
heart-statlog	77.50	84.67	74.17	**87.33**	**80.00**	84.67	76.67	84.00	77.50	85.33
credit-a	83.71	85.38	87.95	84.07	**88.60**	**85.90**	85.34	85.12	86.64	85.12
crx	85.34	**84.60**	86.64	83.81	**87.30**	83.81	84.36	84.33	85.02	78.50
cleveland	78.42	77.44	75.54	**88.41**	**79.14**	81.71	76.98	86.59	77.70	86.59
sonar	61.86	**90.99**	75.26	45.59	**82.47**	71.17	74.23	84.68	74.23	84.68
kr-vs-kp	98.82	99.34	99.41	**99.46**	**99.61**	98.74	99.41	**99.46**	99.41	**99.46**
mushroom	99.13	**100**	99.80	**100**	**100**	**100**	99.18	**100**	99.18	**100**
Average	75.69	83.08	70.53	83.38	75.38	82.39	69.11	**86.22**	**77.42**	84.03

In the Table 3 we can see the global accuracy. This table shows that, in most of the databases, the use of oversampling and undersampling methods jointly with LCMine achieved the best average accuracy in the minority class (*min*). In this table we can also see that Spread Subsample gets the best average accuracy in the minority class; nevertheless, the original LCMine classifier gets the best average accuracy in the majority class (*maj*).

4.3 Accuracy in the Minority Class Regarding the Imbalance Ratio

In this section, we show the accuracy results in the minority class regarding the imbalance ratio. For this analysis we divide the databases in two groups depending if their imbalance ratio is lower than 2 or greater than or equal to 2. The goal of this experiment is to show the behavior of oversampling and undersampling methods with respect to imbalance ratio.

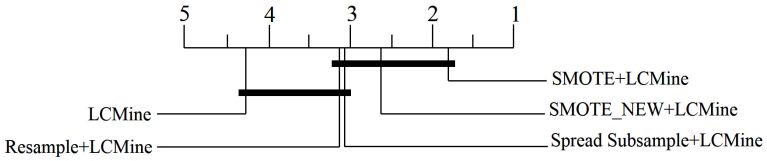


Fig. 4. CD diagram with a statistical comparison of the LCMine classifier before and after using the oversampling and undersampling methods over databases with imbalance ratio lower than 2 (see Table 1)

Figure 4 shows that, when the imbalance ratio is lower than 2, there is not statistical significant difference among the results obtained by LCMiner and those results obtained by Resample+LCMine and Spread Subsample+LCMine. While, applying SMOTE or SMOTE_NEW jointly with LCMine is statistically better than using the original LCMine classifier.

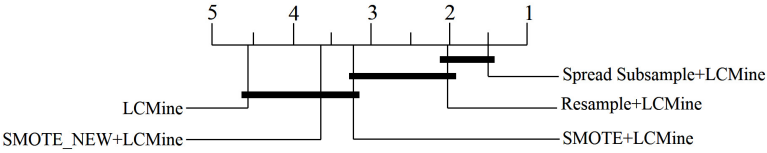


Fig. 5. CD diagram with a statistical comparison of the LCMine classifier before and after using the resampling methods over databases with imbalance ratio greater than or equal to 2 (see Table 1)

Figure 5 shows that, when the imbalance ratio is greater than or equal to 2, there is not statistical significant difference among the results obtained by the original LCMine classifier and the results obtained by SMOTE+LCMine and SMOTE_NEW+LCMine. While, applying Resample or Spread Subsample jointly with LCMine is statistically better than using the original LCMine classifier.

4.4 General Concluding Remarks

The results shown in the previous section lead us to conclude that oversampling and undersampling methods improve the accuracy of the LCMine classifier in the minority class without significantly reducing the accuracy in the majority class. Moreover, if the imbalance ratio is lower than 2 then it is better to use SMOTE; else, it is better to use Spread Subsample.

A possible explanation for this behavior is that, when the imbalance ratio is greater than or equal to 2, the oversampling methods create as many false objects as the real objects in the oversampled class. This way, the classifier cannot correctly classify new objects in the minority class if its knowledge in this class is 50% artificial.

5 Conclusions and Future Work

The classifiers based on emerging patterns are sensitive to databases containing imbalanced classes. These classifiers generate many emerging patterns for the majority class and only a few patterns (or none) for the minority class. This fact affects this type of classifiers, leading them to obtain low accuracy for the minority class.

The main contribution of this paper is an empirical study of the behavior of a classifier based on emerging patterns when using oversampling and undersampling methods in databases containing imbalanced classes. The experimental results show that there is not a method which clearly outperforms the others, but applying any oversampling or undersampling method improves the LCMine classifier accuracy.

From our experimental study we can conclude that if the classes in the database have an imbalance ratio greater than or equal to 2 (1:2) the best option is to use undersampling methods; otherwise, if the ratio is lower than 2 the best option is to use an oversampling method.

As future work, we plan to build a cascade classifier, based on emerging patterns, capable of accurately classifying imbalanced databases with more than two classes. The idea is to apply an oversampling or undersampling method in the complement of the majority class, and then applying this procedure recursively for the other classes, in order to build a cascade classifier. On the other hand, since some studies in the literature propose decision trees that are robust to imbalanced databases [16]; another future work would be studying how to modify the LCMine classifier following these ideas in order to improve its results for imbalanced databases without using oversampling or undersampling methods.

Acknowledgment. This work was partly supported by the National Council of Science and Technology of Mexico under the project grants CB2008-106443 and CB2008-106366.

References

1. Batista, G.E.A.P.A., Prati, R.C., Monard, M.C.: A Study of the Behavior of Several Methods for Balancing Machine Learning Training Data. SIGKDD Explor. Newsl. 6(1), 20–29 (2004)
2. Bhattacharyya, S., Jha, S., Tharakunnel, K., Westland, J.C.: Data mining for credit card fraud: A comparative study. Decision Support Systems 50(3), 602–613 (2011)
3. Blake, C., Merz, C.J.: {UCI} Repository of machine learning databases. Tech. rep., University of California, Irvine, School of Information and Computer Sciences (1998), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
4. Chawla, N.V., Bowyer, K.W., Hall, L.O., Kegelmeyer, W.P.: SMOTE: Synthetic Minority Over-sampling Technique. Journal of Artificial Intelligence Research 16(1), 321–357 (2002)
5. Chawla, N.: Data Mining for Imbalanced Datasets: An Overview. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook, pp. 875–886. Springer, US (2010)
6. Demšar, J.: Statistical Comparisons of Classifiers over Multiple Data Sets. Journal of Machine Learning Research 7, 1–30 (2006)
7. Dong, G.: Preliminaries. In: Dong, G., Bailey, J. (eds.) Contrast Data Mining: Concepts, Algorithms, and Applications, ch. 1. Data Mining and Knowledge Discovery Series, pp. 3–12. Chapman & Hall/CRC, United States of America (2012)
8. Estabrooks, A., Jo, T., Japkowicz, N.: A Multiple Resampling Method For Learning From Imbalanced Data Sets. Computational Intelligence 20(1), 18–36 (2004)
9. Frawley, W.J., Piatetsky-Shapiro, G., Matheus, C.J.: Knowledge Discovery in Databases: An Overview. AI Magazine 13(3), 57–70 (1992)
10. García, S., Herrera, F.: An Extension on "Statistical Comparisons of Classifiers over Multiple Data Sets" for all Pairwise Comparisons. Journal of Machine Learning Research 9, 2677–2694 (2008)
11. García-Borroto, M., Martínez-Trinidad, J.F., Carrasco-Ochoa, J.A., Medina-Pérez, M.A., Ruiz-Shulcloper, J.: LCMine: An efficient algorithm for mining discriminative regularities and its application in supervised classification. Pattern Recognition 43(9), 3025–3034 (2010)
12. García-Borroto, M., Martínez-Trinidad, J., Carrasco-Ochoa, J.: A survey of emerging patterns for supervised classification. Artificial Intelligence Review 1–17 (2012)
13. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations 11(1), 10–18 (2009)
14. Lenca, P., Lallich, S., Do, T.-N., Pham, N.-K.: A comparison of different off-centered entropies to deal with class imbalance for decision trees. In: Washio, T., Suzuki, E., Ting, K.M., Inokuchi, A. (eds.) PAKDD 2008. LNCS (LNAI), vol. 5012, pp. 634–643. Springer, Heidelberg (2008)
15. Li, D.C., Liu, C.W., Hu, S.C.: A learning method for the class imbalance problem with medical data sets. Computers in Biology and Medicine 40(5), 509–518 (2010)
16. Liu, W., Chawla, S., Cieslak, D.A., Chawla, N.V.: A Robust Decision Tree Algorithm for Imbalanced Data Sets. In: SDM 2010, pp. 766–777 (2010)
17. Prati, R.C., Batista, G.E.A.P.A., Monard, M.C.: A Study with Class Imbalance and Random Sampling for a Decision Tree Learning System. In: Bramer, M. (ed.) Artificial Intelligence in Theory and Practice II, vol. 276, pp. 131–140. Springer, Boston (2008)

Recognizing Structural Patterns on Graphs for the Efficient Computation of #2SAT

Guillermo De Ita, Pedro Bello, and Meliza Contreras

Benemérita Universidad Autónoma de Puebla, Faculty of Computer Science
Av. San Claudio y 14 sur, Puebla, México
{deita,pbello,mcontreras}@cs.buap.mx

Abstract. To count models for two conjunctive forms (#2SAT problem) is a classic #P problem. We determine different structural patterns on the underlying graph of a 2-CF F allowing the efficient computation of #2SAT(F).

We show that if the constrained graph of a formula is acyclic or the cycles on the graph can be arranged as independent and embedded cycles, then the number of models of F can be counted efficiently.

Keywords: #SAT Problem, Counting models, Structural Patterns, Graph Topologies.

1 Introduction

#SAT is of special concern to Artificial Intelligence (AI), and it has a direct relationship to Automated Theorem Proving, as well as to approximate reasoning [3,4,9].

The problem of counting models for a Boolean formula (#SAT problem) can be reduced to several different problems in approximate reasoning. For example, for estimating the degree of reliability in a communication network, computing degree of belief in propositional theories, for the generation of explanations to propositional queries, in Bayesian inference, in a truth maintenance systems, for repairing inconsistent databases [1,3,5,9,10]. The previous problems come from several AI applications such as planning, expert systems, approximate reasoning, etc.

#SAT is at least as hard as the SAT problem, but in many cases, even when SAT is solved in polynomial time, no computationally efficient method is known for #SAT. For example, 2-SAT problem (SAT restricted to consider (≤ 2)-CF's), it can be solved in linear time. However, the corresponding counting problem #2-SAT is a #P-complete problem. Earlier works on #2-SAT include papers by Dubois [6], Zhang [11] and Littman [8]. More recently, new upper bounds for exact deterministic algorithms for #2-SAT have been found by Dahllöf [2], Fürer [7], Angelsmark [1] and Jonsson [2]. And given that #2SAT is a #P-complete problem, all the above proposals are part of the class of exponential algorithms.

The maximum polynomial class recognized for #2SAT is the class ($\leq 2, 2\mu$)-CF (conjunction of binary or unary clauses where each variable appears twice at

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 274–283, 2013.

most) [9,10]. Here, we extend such class for considering the topological structure of the undirected graph induced by the restrictions (clauses) of the formula. We extend here some of the procedures presented in [5,4] for the #2-SAT problem and show how to apply them to compute the number of models in a propositional theory. Furthermore, we show different structural patterns on the constrained graph of the formula which allow the efficient computation of the number of models for some classes of 2-CF's.

2 Notation

Let $X = \{x_1, \ldots, x_n\}$ be a set of n Boolean variables. A literal is either a variable x_i or a negated variable $\overline{x_i}$. As usual, for each $x_i \in X$, $x_i^0 = x_i$ and $x_i^1 = x_i$. A clause is a disjunction of different literals (sometimes, we also consider a clause as a set of literals). For $k \in N$, a k-clause is a clause consisting of exactly k literals and, a $(\leq k)$-clause is a clause with at most k literals. A variable $x \in X$ appears in a clause c if either x or $\overline{x}$ is an element of c.

A Conjunctive Form (CF) F is a conjunction of clauses (we also consider a CF as a set of clauses). We say that F is a positive monotone CF if all of its variables appear in unnegated form. A k-CF is a CF containing only k-clauses and, $(\leq k)$- CF denotes a CF containing clauses with at most k literals. A $k\mu$-CF is a formula in which no variable occurs more than k times. A $(k, j\mu)$-CF ($(\leq k, j\mu)$-CF) is a k-CF ($(\leq k)$-CF) such that each variable appears no more than j times.

We use $\nu(X)$ to express the set of variables involved in the object X, where X could be a literal, a clause or a Boolean formula. For instance, for the clause $c = \{\overline{x_1}, x_2\}, \nu(c) = \{x_1, x_2\}$. And $Lit(F)$ is the set of literals which appear in a CF F, i.e. if $X = \nu(F)$, then $Lit(F) = X \cup \overline{X} = \{x_1, \overline{x_1}, \ldots, x_n, \overline{x_n}\}$. We also denote $\{1, 2, \ldots, n\}$ by $[[n]]$.

An assignment s for F is a Boolean function $s : \nu(F) \to \{0, 1\}$. An assignment can be also considered as a set of non complementary pairs of literals. If $l \in s$, being s an assignment, then s turns l true and $\overline{l}$ false. Considering a clause c and assignment s as a set of literals, c is satisfied by s if and only if $c \cap s \neq 0$, and if for all $l \in c$, $\overline{l} \in s$ then s falsifies c. If $F_1 \subset F$ is a formula consisting of some clauses of F, then $\nu(F1) \subset \nu(F)$, and an assignment over $\nu(F_1)$ is a partial assignment over $\nu(F)$. Assuming $n = |\nu(F)|$ and $n_1 = |\nu(F_1)|$, any assignment over $\nu(F_1)$ has 2^{n-n_1} extensions as assignments over $\nu(F)$.

Let F be a Boolean formula in Conjunctive Form (CF), F is satisfied by an assignment s if each clause in F is satisfied by s. F is contradicted by s if any clause in F is contradicted by s. A model of F is an assignment for $\nu(F)$ that satisfies F. Given F a CF, the SAT problem consists of determining if F has a model. The #SAT problem consists of counting the number of models of F defined over $\nu(F)$. #2-SAT denotes #SAT for formulas in 2-CF.

3 Computing #2SAT for Acyclic Formulas

Let F be a 2-CF F, its signed constrained undirected graph is denoted by $G_F = (V(F), E(F))$, with $V(F) = \nu(F)$ and $E(F) = \{\{\nu(x), \nu(y)\} : \{x, y\} \in F\}$, that is, the vertices of G_F are the variables of F, and for each clause $\{x, y\}$ in F there is an edge $\{\nu(x), \nu(y)\} \in E(F)$. Each edge $c = \{\nu(x), \nu(y)\} \in E$ is associated with an ordered pair (s_1, s_2) of signs, assigned as labels of the edge connecting the variables appearing in the clause. The signs s_1 and s_2 are related to the signs of the literals x and y respectively. For example, the clause $\{\overline{x} \vee y\}$ determines the labelled edge: "$x \overset{-+}{---} y$" which is equivalent to the edge "$y \overset{+-}{---} x$".

A graph with labelled edges on a set S is a pair (G, ψ), where $G = (V, E)$ is a graph, and φ is a function with domain E and range S. $\psi(e)$ is called the label of the edge $e \in E$. Let $S = \{+, -\}$ be a set of signs . Let $G = (V, E, \psi)$ be a signed graph with labelled edges on SxS. Let x and y be nodes in V . If $e = \{x, y\}$ is an edge and $\psi(e) = (s, s')$, then $s(s')$ is called the adjacent sign to $x(y)$. We say that a 2-CF F is a path, cycle, or a tree if its signed constrained graph G_F is a path, cycle, or a tree, respectively.

3.1 If the 2-CF Represents a Path

If G_F is a path, then $F = \{C_1, C_2, \ldots, C_m\} = \{\{x_1^{\epsilon_1}, x_2^{\delta_1}\}, \{x_2^{\epsilon_1}, x_3^{\delta_1}\}, \ldots, \{x_m^{\epsilon_m}, x_{m+1}^{\delta_m}\}\}$, where $\delta_i, \epsilon_i \in \{0, 1\}$, $i \in [\![m]\!]$. Let f_i be a family of clauses of the formula F, built as follows: $f_1 = \emptyset; f_i = \{C_j\}_{j<i}$, $i \in [\![m]\!]$. Notice that $n = |\upsilon(F)| = m + 1$, $f_i \subset f_{i+1}$, $i \in [\![m-1]\!]$. Let $SAT(f_i) = \{s : s \text{ satisfies } f_i\}$, $A_i = \{s \in SAT(f_i) : x_i \in s\}$, $B_i = \{s \in SAT(f_i) : \overline{x}_i \in s\}$. Let $\alpha_i = |A_i|$; $\beta_i = |B_i|$ and $\mu_i = |SAT(f_i)| = \alpha_i + \beta_i$.

For every node $x \in G_F$ a pair (α_x, β_x) is computed, where α_x indicates how many times the variable x is 'true' and β_x indicates the number of times that the variable x can take value 'false' into the set of models of F. The first pair is $(\alpha_1, \beta_1) = (1, 1)$ since x_1 can be true or false in order to satisfy f_1. The pairs (α_x, β_x) associated to each node x_i,$i = 2, \ldots, m$ are computed according to the signs (ϵ_i, δ_i) of the literals in the clause c_i by the following recurrence equation:

$$(\alpha_i, \beta_i) = \begin{cases} (\beta_{i-1}, \alpha_{i-1} + \beta_{i-1}) & \text{if } (\epsilon_i, \delta_i) = (-, -) \\ (\alpha_{i-1} + \beta_{i-1}, \beta_{i-1}) & \text{if } (\epsilon_i, \delta_i) = (-, +) \\ (\alpha_{i-1}, \alpha_{i-1} + \beta_{i-1}) & \text{if } (\epsilon_i, \delta_i) = (+, -) \\ (\alpha_{i-1} + \beta_{i-1}, \alpha_{i-1}) & \text{if } (\epsilon_i, \delta_i) = (+, +) \end{cases} \quad (1)$$

Note that, as $F = f_m$ then $\#SAT(F) = \mu_m = \alpha_m + \beta_m$. We denote with $\rightarrow$ the application of one of the four rules of the recurrence (1).

Example 1. *Let* $F = \{(x_1, x_2), (\overline{x}_2, \overline{x}_3), (\overline{x_3}, \overline{x}_4), (x_4, \overline{x}_5), (\overline{x}_5, x_6)\}$ *be a path. The series* $(\alpha_i, \beta_i), i \in [\![6]\!]$, *is computed as:* $(\alpha_1, \beta_1) = (1, 1) \rightarrow (\alpha_2, \beta_2) = (2, 1)$ *since* $(\epsilon_1, \delta_1) = (1, 1)$, *and the rule 4 has to be applied. In general, applying the corresponding rule of the recurrence (1) according to the signs expressed by* $(\epsilon_i, \delta_i), i = 3, ..., 6$, *we have* $(2, 1) \rightarrow (1, 3) \rightarrow (3, 4) \rightarrow (3, 7) \rightarrow (\alpha_6, \beta_6) = (10, 7)$, *and then,* $\#SAT(F) = \mu_6 = \alpha_6 + \beta_6 = 10 + 7 = 17$.

3.2 If the 2-CF Represents a Tree

Let F be a 2-CF where its associated constrained graph G_F is a tree. We denote with (α_v, β_v) the pair associated with the node v ($v \in G_F$). We compute $\#SAT(F)$ while we are traversing by G_F in post-order.

Algorithm Count_Models_for_trees(G_F)
Input: G_F - a tree graph.
Output: The number of models of F
Procedure:
Traversing G_F in post-order, and when a node $v \in G_F$ is left, assign:

1. $(\alpha_v, \beta_v) = (1, 1)$ if v is a leaf node in G_F.
2. If v is a parent node with a list of child nodes associated, i.e., $u_1, u_2, ..., u_k$ are the child nodes of v, as we have already visited all child nodes, then each pair $(\alpha_{u_j}, \beta_{u_j})$ $j = 1, ..., k$ has been determined based on recurrence (1). Then, let $\alpha_v = \prod_{j=1}^{k} \alpha_{v_j}$ and $\beta_v = \prod_{j=1}^{k} \beta_{v_j}$. Notice that this step includes the case when v has just one child node.
3. If v is the root node of G_F then return$(\alpha_v + \beta_v)$.

This procedure returns the number of models for F in time $O(n + m)$ which is the necessary time for traversing G_F in post-order.

Example 2. *If* $F = \{(x_1, x_2), (x_2, x_3), (x_2, x_4), (x_2, x_5), (x_4, x_6), (x_6, x_7), (x_6, x_8)\}$ *is a monotone 2-CF, we consider the post-order search starting in the node* x_1. *The number of models at each level of the tree is shown in Figure 1. The procedure Count_Models_for_trees returns for* $\alpha_{x_1} = 41$, $\beta_{x_1} = 36$ *and the total number of models is:* $\#SAT(F) = 41 + 36 = 77$.

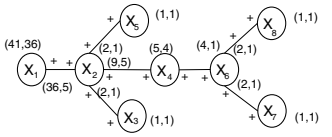


Fig. 1. Counting models over a tree

4 Processing 2-CF's Containing Cycles

Let G_F be a simple cycle with m nodes, that is, all the variables in $v(F)$ appear twice, $|V| = m = n = |E|$. Ordering the clauses in F in such a way that $| v(c_i) \cap v(c_{i+1}) | = 1$, and $c_{i_1} = c_{i_2}$ whenever $i_1 \equiv i_2 \bmod m$, hence $x_1 = x_m$, then $F = \left\{ c_i = \{x_{i-1}^{\epsilon_i}, x_i^{\delta_i}\} \right\}_{i=1}^{m}$, where $\delta_i, \epsilon_i \in \{0, 1\}$. Decomposing F as $F = F' \cup c_m$, where $F' = \{c_1, ..., c_{m-1}\}$ is a path and $c_m = (x_{m-1}^{\epsilon_m}, x_1^{\delta_m})$ is the edge which conforms with $G_{F'}$ the simple cycle: $x_1, x_2, ..., x_{m-1}, x_1$. We will call to

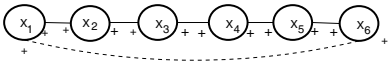
$G_{F'}$ the internal path of the cycle and to c_m the back clause of the cycle. We can apply the linear procedure described above in equation 1 for computing $\#SAT(F')$.

Every model of F' had determined logical values for the variables: x_{m-1} and x_1 since those variables appear in $v(F')$. Any model s of F' satisfies c_m if and only if ($x_{m-1}^{1-\epsilon_m} \notin s$ and $x_m^{1-\delta_m} \notin s$), that is, $SAT(F' \cup c_m) \subseteq SAT(F')$, and $SAT(F' \cup c_m) = SAT(F') - \{s \in SAT(F') : s \text{ falsifies } c_m\}$. Let $X = F' \cup \{(x_{m-1}^{1-\epsilon_m}) \wedge (x_m^{1-\delta_m})\}$, then $\#SAT(X)$ is computed as a path with two unitary clauses:

$$\#SAT(F) = \#SAT(F' \wedge C_m) = \#SAT(F') - \#SAT(F' \wedge (x_{m-1}^{1-\epsilon_m}) \wedge (x_m^{1-\delta_m})) \quad (2)$$

Example 1. Let $\Sigma = \{c_i\}_{i=1}^6 = \{\{x_1, x_2\}, \{x_2, x_3\}, \{x_3, x_4\}, \{x_4, x_5\}, \{x_5, x_6\}, \{x_6, x_1\}\}$ be a monotone 2-CF which represents a cycle: G_Σ=(V,E). Let $G' = (V, E')$ where $E = E' \cup \{c_6\}$, that is, the new graph G' is Σ minus the edge c_6. Applying equation (2), we have that $\#SAT(\Sigma) = \#SAT(F') - \#SAT(F' \wedge \overline{x}_6 \wedge \overline{x}_1) = 21 - 3 = 18$. This example is illustrated in figure 2.

When we count models over any constrained graph G_F, we use *computing threads*. A computing thread is a sequence of pairs $(\alpha_i, \beta_i), i = 1, \ldots, m$ used for computing the number of models over a path of m nodes. A main thread, denoted by L_p, is associated to a spanning tree of G_F, this thread is always active until the process of counting finishes completely. While the thread used for computing the pair associated with $\#SAT(F' \wedge (x_{m-1}^{1-\epsilon_m}) \wedge (x_m^{1-\delta_m}))$ is denoted by L_e.



$(\alpha_1, \beta_1) \rightarrow (\alpha_2, \beta_2) \rightarrow (\alpha_3, \beta_3) \rightarrow (\alpha_4, \beta_4) \rightarrow (\alpha_5, \beta_5) \rightarrow (\alpha_6, \beta_6)$
$L_p : (1,1) \rightarrow (2,1) \rightarrow (3,2) \rightarrow (5,3) \rightarrow (8,5) \rightarrow (13,8)$
$L_e : (0,1) \rightarrow (1,0) \rightarrow (1,1) \rightarrow (2,1) \rightarrow (3,2) \rightarrow (5,3)$
$\Rightarrow (13,8) - (0,3) = (13,5)$

Fig. 2. Computing $\#SAT(F)$ when G_F is a cycle

4.1 Cycles on Alternating Signed Paths

Let $G_F = (V, E, \{+, -\})$ be a signed connected graph of an input formula F in 2-CF. Let v_r be the node of minimum degree in G_F which is chosen to start a depth-first search. We obtain a spanning tree T_G with v_r as the root node and a set of fundamental cycles $C = \{C_1, C_2, ..., C_k\}$, and where each back edge $c_i \in E$ marks the beginning and the end of a fundamental cycle $C_i \in C$.

The edges in T_G are called *tree edges*. A *back edge* $e \in E$ is an edge of G_F which is not part of the spanning tree T_G but e is incident to two nodes of T_G. Each back edge holds the maximum path contained in the fundamental cycle which is part of. We will call to such maximum path, the *internal path* of a fundamental cycle. Given any pair of fundamental cycles C_i and C_j in G_F, if C_i and C_j share edges, we call them *intersecting* cycles; otherwise, they are called *independent* cycles.

In some cases, the value $\#\mathrm{SAT}(C_i)$ for a fundamental cycle $C_i \in G_F$ can be computed previously to the computation of the total graph G_F in order to determine if the cycle C_i can be reduced to a path or any other simple structure. For example, let us assume a cycle C_i whose internal path is formed by nodes with alternating signs on its edges, while the signs on the back edge e determine the different cases to analyze. Let us order the nodes into the internal path of the cycle as: $x_1 - x_2 - \ldots - x_k$, and we consider to x_1 as the initial node and x_k as the final node of the cycle. Notice that $e = \{x_1, x_k\}$ is the back edge.

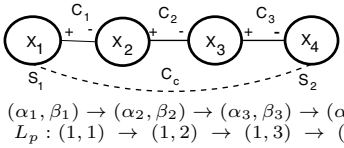
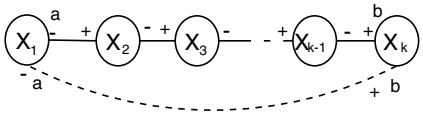


Fig. 3. G_F and the computing over the path

Figure 3 shows a cycle with an internal path whose nodes have alternating signs. Let $\psi(e) = (a, b')$ be the signs on the back edge. Assuming that the variable x_1 appears only with sign a and the variable x_k appears only with sign b, that means that the signs of the back edge coincides with the signs of its endpoints in the internal path. For this case, the back edge (its corresponding clause) can be eliminated from the cycle because the final pair obtained in the secondary thread L_e is $(0, 0)$, as it is shown in figure 4.

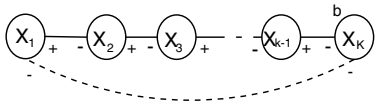


$$(\alpha, \beta) \to (\alpha + \beta, \beta) \to (\alpha + 2\beta, \beta) \ldots \to (\alpha + k \cdot \beta, \beta) \Rightarrow (\alpha + k \cdot \beta, \beta)$$
$$(\alpha, 0) \to (\alpha, 0) \quad \to (\alpha, 0) \quad \ldots \to (\alpha, 0) \quad \Rightarrow -(0, 0)$$

Fig. 4. When a back edge does not substract models to the path

The previous example shows that pre-processing the cycles appearing in a current constrained graph is relevant. In some cases, the clause corresponding

with the back edge can be eliminated or the value $\#SAT(C_i)$ for a cycle C_i can be computed using symbolic values without knowing the real values of the pairs (α_j, β_j) on the internal path of the cycle, as it is shown in figure 5.



$$(\alpha,\beta) \to (\alpha,\alpha+\beta) \to \ldots \to (\alpha, k\cdot\alpha+\beta) \Rightarrow (\alpha, k\cdot\alpha+\beta) - (\alpha,0) = (0, k\cdot\alpha+\beta)$$
$$(\alpha,0) \to (\alpha,\alpha) \quad \to \ldots \to (\alpha, k\cdot\alpha) \Rightarrow -(\alpha,0)$$

Fig. 5. Computing $\#SAT(C_i)$ using symbolic values on the pairs (α_j, β_j)

5 Processing Embedded Cycles

Let $G_F = (V, E)$ be a connected constrained graph of a 2-CF F. Given two intersecting plane cycles C_i, C_j of a graph, C_i is embedded into C_j, if
a)$V(C_i) \subset V(C_j)$: the set of nodes of C_i is a subset of the nodes of C_j.
b)$|E(C_i) - E(C_j)| = 1$: there is only one edge from C_i which is not edge of C_j.
In this case, C_i is an internal embedded cycle of C_j and C_j is an external embedded cycle of C_i.

Let us consider a graph G_F formed by a set $D = (C_1, C_2, \ldots, C_k)$ of embedded cycles, such that C_i is embedded in C_{i+1}, $i = 1, \ldots, k-1$. C_1 is the most internal embedded cycle and C_k is the most external cycle of D. For processing this class of graphs, we determine a processing order given by traversing the graph from the most internal to the most external embedded cycle.

For a graph G_F formed by a set of embedded cycles, Lp will be associated with the path formed by the nodes of G_F. Three computing threads are used for processing a current cycle, and for processing all the set of embedded cycles we require at most six computing threads.

Case 1: Processing the Most Internal Embedded Cycle
Let $e_b = \{v_s, v_f\}$ be the back edge which embraces the most internal cycle C_1 of G_F. We use three computing threads with initial values: $(\alpha_s^1, \beta_s^1) = (1, 1)$, $(\alpha_s^2, \beta_s^2) = (1, 0)$ - this thread carry on the number of models of C_1 where the

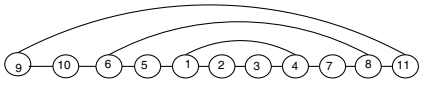
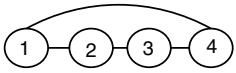


Fig. 6. An initial graph G_F with embedded cycles

variable x_s does not appear, and $(\alpha_s^3, \beta_s^3) = (0, 1)$ - this thread carry on the number of models of C_1 where the first variable x_0 appears.

We traverse by the internal path of C_1 from its initial node v_s to its end node v_f and the last visited edge is its back edge e_b. Each time that a new node on the path is visited, recurrence (1) is applied, obtaining: $(\alpha_i^j, \beta_i^j) \rightarrow (\alpha_{i+1}^j, \beta_{i+1}^j), j = 1, \ldots, 3$. When the search arrives to v_f, we have obtained the pairs (α_f^1, β_f^1), (α_f^2, β_f^2) and (α_f^3, β_f^3). The last edge processed is e_b and for this, we use two temporal variables α_{C1} and β_{C1} defined as: $\alpha_{C1} = \alpha_f^1$, $\beta_{C1} = \beta_f^1 - \beta_f^3$ for the monotone case or according to the signs associated with e_b this last rule is tuned. And, the numbers of models for the internal cycle C_1, is $\#SAT(C_1) = \alpha_{C1} + \beta_{C1}$.



$$
\begin{array}{rl}
Nodes: & Node_1 \quad Node_2 \quad Node_3 \quad Node_4 \\
Lp: & (1,1) \rightarrow (2,1) \rightarrow (3,2) \rightarrow (5,3) \;=\; (\alpha_4^1, \beta_4^1) \\
L1 \notin S: & (1,0) \rightarrow (1,1) \rightarrow (2,1) \rightarrow (3,2) \;=\; (\alpha_4^2, \beta_4^2) \\
L2 \in S: & (0,1) \rightarrow (1,0) \rightarrow (1,1) \rightarrow (2,1) \;=\; (\alpha_4^3, \beta_4^3) \\
& \Rightarrow (\alpha_{C1}, \beta_{C1}) = (5, 3-1) = (5, 2)
\end{array}
$$

Fig. 7. Processing the most internal cycle

Case 2: Processing an External Embedded Cycle
Let $C_j = C_{i+1}$ be the following external embedded cycle of the last processed cycle C_i. After processing an internal embedded cycle C_i, all the cycle is contracted into a single node Cr_{sf} where s is the number of the initial node and f is the number of the final node of the path in C_i. Then, Cr_{ij} is now a new fat node on the path formed by the nodes: $(V(C_j) - V(C_i)) \cup Cr_{ij}$.

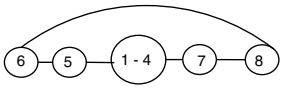
We use new three computing threads for processing the external cycle C_j according to the previous case 1. C_j is traversing as a path and appliying recurrence (1) over each node of the path until arrive to the fat node Cr_{sf}. When the computing threads cross by Cr_{sf}, each current pair $(\alpha_x^i, \beta_x^i), i = 1, 2, 3$ is updated according to the following recurrence.

$$
\begin{aligned}
\alpha_{x+1}^i &= \alpha_f^2 \cdot \alpha_x^i + \alpha_f^3 \cdot \beta_x^i \\
\beta_{x+1}^i &= \beta_f^2 \cdot \alpha_x^i + \beta_f^3 \cdot \beta_x^i, \text{for } i = 1, 2, 3.
\end{aligned}
\tag{3}
$$

Obtaining the new pairs $(\alpha_{x+1}^i, \beta_{x+1}^i)$, for $i = 1, 2, 3$. We will denote the application of the recurrence (3) as $(\alpha_x, \beta_x) \odot (\alpha_{x+1}, \beta_{x+1})$. As it exists an implicit back edge into the contracted fat node Cr_{sf} then we have to update the pair $(\alpha_{x+1}^1, \beta_{x+1}^1)$ as $(\alpha_{x+1}^1, \beta_{x+1}^1) = (\alpha_{x+1}^1, \beta_{x+1}^1 - \beta_x^1 * \beta_f^3)$. We will denote the processing of a back edge by $\hookleftarrow$, then $(\alpha_x, \beta_x) \hookleftarrow (\alpha_{x+1}, \beta_{x+1})$ meaning the application of the formula $(\alpha_{x+1}, \beta_{x+1}) = (\alpha_{x+1}, \beta_{x+1} - \beta_x * \beta_f^2)$.

We obtain new current values for the last node of the cycle C_j:(α_f^1, β_f^1),(α_f^2, β_f^2), (α_f^3, β_f^3) and the cycle C_j is contracted into a new fat node C_{kf} where k was the number of the initial node and f was the number of the final node processed in

C_j. In this way, we process any embedded cycle until arrives to the most external cycle of G_F.



$$
\begin{array}{rl}
G_2 & Node_6 \;\; Node_5 \qquad\qquad Node_{1-4} \qquad\qquad Node_7 \qquad Node_8 \\
Lp: & (1,1) \to (2,1) \to (3,2) \odot (13,8) \hookleftarrow (13,6) \to (19,13) \to (32,19) = (\alpha_8^1, \beta_8^1) \\
L1 \notin S: & (1,0) \to (1,1) \to (2,1) \odot (8,5) \;\hookleftarrow (8,4) \;\to (12,8) \;\to (20,12) = (\alpha_8^2, \beta_8^2) \\
L2 \in S: & (0,1) \to (1,0) \to (1,1) \odot (5,3) \;\hookleftarrow (5,2) \;\to (7,5) \;\to (12,7) \;= (\alpha_8^3, \beta_8^3) \\
& \Rightarrow (32,19) - (0,7) = (32,12)
\end{array}
$$

Fig. 8. Processing an external embedded cycle

Notice that our proposal for counting models on a set D of embedded cycles has a linear time complexity over the number of edges $|E(D)|$. Then, we have shown that for some restricted cases of a 2-CF F, particularly when G_F contains only independent and embedded cycles, #2-SAT(F) can be computed in polynomial time.

6 Conclusion

#SAT problem for the class of Boolean formulas in 2-CF is a classical #P-complete problem. However, However, there are several instances of 2-CF's for which #2SAT can be solved efficiently.

We have shown here, different polynomial-time procedures for counting models of Boolean formulas for subclasses of 2-CF's. For example, for formulas whose contrained graph is acyclic, its corresponding number of models is computed in linear time.

Given a formula F in 2-CF, we show that if the cycles in its constrained graph G_F can be arranged as independent and embedded cycles, then we can count efficiently the number of models of F.

Thus, the unique graph topology for the constrained graph G_F of a 2-CF F where the computation of #2SAT(F) continues being intractable is when G_F has intersected cycles and they can not be arranged as embedded cycles.

References

1. Angelsmark, O., Jonsson, P.: Improved Algorithms for Counting Solutions in Constraint Satisfaction Problems. In: Rossi, F. (ed.) CP 2003. LNCS, vol. 2833, pp. 81–95. Springer, Heidelberg (2003)
2. Dahllöf, V., Jonsonn, P., Wahlström, M.: Counting models for 2SAT and 3SAT formulae. J. Theoretical Computer Sciences 332(1-3), 265–291 (2005)
3. Darwiche, A.: On the Tractability of Counting Theory Models and its Application to Belief Revision and Truth Maintenance. J. of Applied Non-classical Logics 11(1-2), 11–34 (2001)

4. De Ita, G., Tovar, M.: Applying Counting Models of Boolean Formulas to Propositional Inference. J. Advances in Computer Science and Engineering Researching in Computing Science 19, 159–170 (2006)
5. De Ita, G., Bello, P., Contreras, M.: New Polynomial Classes for #2SAT Established via Graph-Topological Structure. Engineering Letters 15(2), 250–258 (2007)
6. Dubois, O.: Counting the number of solutions for instances of satisfiability. J. Theoretical Computer Sciences 81(1), 49–64 (1991)
7. Fürer, M., Prasad, S.K.: Algorithms for Counting 2-SAT Solutions and Coloring with Applications. Technical Report No. 33, Electronic Colloqium on Comp. Complexity (2005)
8. Littman, M.L., Pitassi, T., Impagliazzo, R.: On the Complexity of counting satisfying assignments. Technical Report Unpublished manuscript
9. Roth, D.: On the hardness of approximate reasoning. J. Artificial Intelligence 82, 273–302 (1996)
10. Russ, B.: Randomized Algorithms: Approximation, Generation, and Counting, Distingished dissertations. Springer (2001)
11. Zhang, W.: Number of models and satisfiability of set of clauses. J. Theoretical Computer Sciences 155(1), 277–288 (1996)

A Supervised Learning Approach to Construct Hyper-heuristics for Constraint Satisfaction

José Carlos Ortiz-Bayliss[1], Hugo Terashima-Marín[2], and Santiago Enrique Conant-Pablos[2]

[1] Automated Scheduling, Optimisation and Planning (ASAP)
School of Computer Science, University of Nottingham, UK
Jose.Ortiz_Bayliss@nottingham.ac.uk
[2] Tecnológico de Monterrey, Campus Monterrey, Mexico
{terashima,sconant}@itesm.mx

Abstract. Hyper-heuristics are methodologies that 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. In this paper we propose a logistic regression model to generate hyper-heuristics for variable ordering within constraint satisfaction problems. The first step in our approach requires to generate a training set that maps any given instance, expressed in terms of some of their features, to one suitable variable ordering heuristic. This set is used later to train the system and generate a hyper-heuristic that decides which heuristic to apply given the current features of the instances at hand at different steps of the search. The results suggest that hyper-heuristics generated through this methodology allow us to exploit the strengths of the heuristics to minimize the cost of the search.

Keywords: Constraint Satisfaction, Hyper-heuristics, Logistic Regression.

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 [17]. 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. 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] and [17].

Several deterministic methods to solve CSPs exist(see for example [14], 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 [24]. When using DFS to solve CSPs, every variable represents a node in the tree and the deeper we go in that tree, the larger the

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 284–293, 2013.

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 in 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 [11]. 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 [10]. Thus, every time a variable is instantiated, the values of the other variables that are not allowed due to the current instantiation are removed.

Logistic regression is a type of regression analysis used for classification problems. The idea is to map from a set of input variables to one output variable that represents a class. Logistic regression is one type of supervised machine learning because training examples where the expected output corresponding to the given input must be provided. The general idea of this investigation is to combine the strengths of some existing heuristics through a logistic regression approach to generate a method that chooses among heuristics 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], [5] and [21]) and they have achieved promising results on 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 which includes the features used to characterize the CSP instances, the set of heuristics used and the hyper-heuristic model. The experiments and main results are discussed in Sec. 4. Finally, Sec. 5 presents the conclusion and future work.

2 Background and Related Work

The idea of combining heuristics goes back to 1960s [8] and has been used in many investigations under different names [26,28,7]. Hyper-heuristics are one alternative to combine the strengths of heuristics based on the current problem features. Hyper-heuristics can be divided into two main classes: those which select from existing heuristics and those that generate new heuristics [21]. A more detailed description about the classification of hyper-heuristics can be found

in [6]. In this investigation we will focus our attention on hyper-heuristics that select from existing heuristics.

The first attempts to systematically map CSPs to algorithms and heuristics according to the features of the instances were presented in [29]. In their work, the authors presented a survey of algorithms and heuristics for solving CSPs and proposed a relation between the formulation of the CSP and the most adequate solving method for that formulation. More recently, Ortiz-Bayliss et al. [19] 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. [27], who proposed a genetic algorithm framework to produce hyper-heuristics for variable ordering; Bittle and Fox [3] who presented a hyper-heuristic approach for variable and value ordering based on a symbolic cognitive architecture augmented with case based reasoning as the machine learning mechanism for their hyper-heuristics; and recent works where neural networks are used as hyper-heuristics for variable ordering [18,20]. The differences between all these works on hyper-heuristics for CSPs lies in the set of heuristics used and the learning mechanism used to produce the hyper-heuristics.

3 Solution Model

In this section we discuss the problem state representation, the set of heuristics and the hyper-heuristic model used in this investigation.

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. [23] 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 the use of three important binary CSPs properties known as constraint density (p_1), constraint tightness (p_2) and κ [12]. 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. 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. The value of κ is suggested in the literature as a general measure of how restricted a combinatorial problem is. If κ is small, the instances usually have many solutions with respect to their size. When κ is large, instead, the instances often have few solutions or do not have any at all [12]. κ is defined as $\kappa = \frac{-\sum_{c\in C}\log_2(1-p_c)}{\sum_{x\in X}\log_2(m_x)}$, where p_c is the fraction of unfeasible tuples on constraint c and m_x is the domain size of variable x. It has been found that the most difficult instances with respect to their size occur when $\kappa \approx 1$ [12].

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, p_2 and κ change, and a sub-problem with new features appears. This is the reason why we decided to use this set of features to represent the problem state and guide the selection of the heuristics.

3.2 Variable Ordering Heuristics

A solution to any given CSP is constructed by selecting one variable at the time based on one of the five variable ordering heuristics used in this investigation: minimum domain (DOM), maximum weighted degree (WDEG), domain over weighted degree (DOM/WDEG), kappa (K) and maximum tightness (MXT). 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.

DOM. This heuristic selects the variable with the fewer available values in its domain [13,22].

WDEG. This heuristic attaches a counter, called *weight*, to every constraint of the problem [4,15]. The counters will be updated during the search whenever a dead-end occurs (no more values available for the current variable remain). This heuristic gives priority the variables with the largest weighted degrees.

DOM/WDEG This is a combination of DOM and WDEG heuristics. It selects first the variable that maximizes the quotient of the domain size over the weighted degree of the variable.

K orders the variables based on the value of the kappa factor, κ. K will select first the variable that minimizes the value of κ of the remaining instance [12].

MXT prefers the variable with the tightest constraints (the one with the highest average value of p_2 among all the constraints where it is involved).

We have also used Min-Conflicts [16] as value ordering heuristic to improve the search. When using Min-Conflicts, the next value to try for the selected variable is the one involved in the minimum number of conflicts. Min-Conflicts is not considered as part of the hyper-heuristic model because it is a value ordering heuristic and at the moment we are only using the hyper-heuristic approach for variable ordering. We expect to extend our approach to include value ordering as part of the hyper-heuristic on future developments.

3.3 Instances Used

Our set of instances includes 1000 random binary CSPs distributed among three sets. We will refer to these sets as training set, cross validation set and testing set. These sets contain 600, 200 and 200 instances, respectively. All the CSP instances used for this research were randomly generated with a modified version of model D [25]. First, a constraint graph G with n nodes is randomly constructed and then, 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 five parameters: $\langle n, m, \sigma_m, w_1, w_2 \rangle$. The number of variables is defined by n and the domain size by m; with a maximum deviation of σ in the domain size of each variable. The parameter w_1 determines the probability that a constraint exists in the CSP instance, whereas w_2 determines the probability that an unfeasible pair of values occurs on each constraint.

For each instance, the number of variables was randomly selected in the range $[15, 30]$, where each variable can contain a domain of size in the range $[10, 20]$ (the domain is not uniform among the variables within each instance because of the parameter σ_m used by the generator). The values of w_1 and w_2 were determined by independently choosing at random values in the range $[0, 1]$ for each instance.

Because sometimes more than one heuristic obtains the best result for a given instance (the one with the minimum number of consistency checks), it was necessary to perform a filtering process during the generation of the instances. All the instances where more than one heuristic obtained the best result in terms of consistency checks were discarded and another one was created.

3.4 The Hyper-heuristic Model

This investigation describes a hyper-heuristic model for variable ordering on CSPs based on a logistic regression approach. The hyper-heuristic proposed in this investigation dynamically decides which heuristic to apply as the search progreses. At each step of the search, every time a new variable is to be instantiated, the hyper-heuristic decides which heuristic to apply according to the current problem state (defined by the values of p_1, p_2 and κ).

The hyper-heuristic contains a module for multi-class logistic regression. The hyper-heuristic needs to be trained before being applied. A detailed description of the training process will be provided in the next sections. The core of the hyper-heuristic contains a sigmoid function:

$$h(\boldsymbol{\theta_h}, \boldsymbol{f}) = \frac{1}{1 + e^{\boldsymbol{\theta_h} \cdot \boldsymbol{f}}} \tag{1}$$

where $\boldsymbol{f}$ is the vector of features that characterizes the current problem instance (p_1, p_2, κ). The vector $\boldsymbol{\theta_h}$ is adjusted for each heuristic during the training phase.

In our model, each heuristic is associated a specific vector $\boldsymbol{\theta_h}$. The function $h(\boldsymbol{\theta_h}, \boldsymbol{f})$ is evaluated with the corresponding $\boldsymbol{\theta_h}$ from each heuristic and the vector of features $\boldsymbol{f}$ of the current problem state. The heuristic which $\boldsymbol{\theta_h}$ produces the largest output is selected to be applied on the instance. This is a common way to implement multi-class classification by using logistic regression. In the next section we will discuss how to obtain the vectors $\boldsymbol{\theta_h}$ associated to each heuristic.

4 Experiments and Results

In this investigation, we are using logistic regression for multi-class classification. A vector $\boldsymbol{\theta_h}$ is generated for each heuristic. A set of examples containing the

features $\boldsymbol{f}$ and the best heuristic for such features was obtained from the training set. For each instance in this set, its features (p_1, p_2 and κ) and the most suitable heuristic (the one that required the fewer consistency checks for the search) were saved. The training set was later used to produce specific training examples for each heuristic. These particular training examples contain only binary outputs: 1 when the example corresponds to problem features that made the current heuristic the best option, and 0 otherwise. We used a gradient descend procedure to obtain the values of $\boldsymbol{\theta_h}$ that minimize the cost function for each heuristic. We used 0.01 as learning rate and 1000 iterations to minimize the cost function.

There is a cost function associated to the minimization problem. In this case, the cost function is given by:

$$J(\boldsymbol{\theta_h}) = \begin{cases} \log(h(\boldsymbol{\theta_h}, \boldsymbol{f})) & y = 1 \\ \log(1 - h(\boldsymbol{\theta_h}, \boldsymbol{f})) & y = 0 \end{cases} \tag{2}$$

Then, the idea is to find a vector $\boldsymbol{\theta_h}$ that minimizes the cost function $J(\boldsymbol{\theta_h})$. With gradient descend, at each iteration we must simultaneously update all the values in $\boldsymbol{\theta_h}$ by using the following equation:

$$\boldsymbol{\theta_h} = \boldsymbol{\theta_h} - \alpha \sum_{j=1}^{l} h(\boldsymbol{\theta_h}, \boldsymbol{f}^{(j)}) \boldsymbol{f}_i^{(j)} \tag{3}$$

where l is the number of examples in the training set and $\boldsymbol{f}^{(j)}$ is the jth example of the training set and $\boldsymbol{f}_i^j$ is the feature i of the jth example of the training set (p_1, p_2 and κ, in that order).

At the end, the values of the vectors $\boldsymbol{\theta_h}$ obtained for each heuristic are:

$$\begin{gathered} \boldsymbol{\theta_{MRV}} = [-5.6922, 2.5416, 1.6944, 0.6989] \\ \boldsymbol{\theta_{WDEG}} = [1.7820, -0.6832, 1.8075, -2.8147] \\ \boldsymbol{\theta_{DOM/WDEG}} = [-5.6356, 2.7119, 3.9738, 0.3091] \\ \boldsymbol{\theta_K} = [-3.1786, 1.6747, 2.0354, 0.1376] \\ \boldsymbol{\theta_{MXT}} = [-1.6886, 0.7275, -4.9322, 0.7212] \end{gathered} \tag{4}$$

We used this set of vectors as the core of the hyper-heuristic that was tested in the following experiments. Nevertheless the vector of features $\boldsymbol{f}$ contains only three elements, $\boldsymbol{\theta_h}$ contains four because the first element in the vector corresponds to a bias. To be consistent, a fixed feature with a constant value of 1 is added at the first position of the vector of features $\boldsymbol{f}$. Thus, the first element of the vector $\boldsymbol{\theta_h}$ is always multiplied by 1 at the moment of calculating $\boldsymbol{\theta_h} \cdot \boldsymbol{f}$.

4.1 Evaluating the Hyper-heuristics

We tried the hyper-heuristic obtained with the proposed approach on the three sets. The results are shown in Table 1. It is important to stress that for these instances, all the methods are able to find a solution or to prove that none exists. Then, the only difference is the number of consistency checks which is used for comparison of the quality of the methods.

Table 1. Percentage of instances on each set where each method obtains the best result (requires the fewer consistency checks)

Method	Training set	Validation set	Test set
MRV	5.00%	3.50%	3.50%
WDEG	**57.50**%	**57.00**%	**55.00**%
DOM/WDEG	8.83%	10.50%	7.50%
K	18.83%	20.50%	22.50%
MXT	9.83%	8.50%	11.50%
HH	**62.67**%	**64.50**%	**60.50**%

We can observe that, even though WDEG is clearly the heuristic that achieves the minimum number of consistency checks on the largest fraction of instances on the three sets, the hyper-heuristic is able to overcome this proportion in all the sets. One consideration about these results is the fact that the hyper-heuristic is choosing among different heuristics during the search and then, the cost of the search (in terms of consistency checks) is not exactly the same than the best result obtained by the heuristics applied in isolation. Then, we must interpret the results from table 1 as the proportion of instances where the hyper-heuristic behaves at least as well as the best result obtained with the heuristics applied in isolation.

In Table 2 we present the average costs of each method on the three sets. The average cost for each method is calculated as the average consistency checks required by each method to solve an instance in the set (the sum of the costs on all the instances over the number of instances).

Table 2. Average cost per instance (in consistency checks) for each method on the three sets

Method	Training set	Validation set	Test set
MRV	228430	299319	79901
WDEG	252989	451341	133057
DOM/WDEG	195621	**222157**	84241
K	**160538**	222780	**74548**
MXT	286714	301132	258061
HH	190447	229294	82566

The results of the average costs per instance on the different sets show that, even though WDEG was the heuristic that most of the times obtained the best results, it is not the heuristic with the smallest average cost. This is an interesting result that makes us think that most of the instances that were best solved by using WDEG were not hard, and then, the variance in the results with respect to the other heuristics was small. On the other hand, the instances where K and DOM/DEG (on the cross validation set) were the best options, represent large reductions in the number of consistency checks.

Even though the hyper-heuristic is not able to overcome K with respect to the average cost of the search, it is a very competent solving method with respect to the other heuristics. It is important to stress the difference in the average cost of WDEG and the hyper-heuristic. The percentage of instances where both methods achieved the best results are very close to each other (around 5%), but when we evaluate the average cost, the hyper-heuristic proves its real contribution. The hyper-heuristic is, in all the sets, a best solving method than WDEG. The reductions in the average costs obtained by using the hyper-heuristic, with respect to WDEG, are of 24.72%, 49.19% and 37.94% for the training, cross validation and test set, respectively.

5 Conclusion and Future Work

We have explored the use of a logistic regression approach to produce hyper-heuristics for CSPs. The results show that it is possible to map a CSP instance to one suitable heuristic given the described features. The hyper-heuristic outperforms the best heuristic in the fraction of instances where it achieves the best results. Also, the hyper-heuristic is very competent with respect to the average cost per instance. We observed that the hyper-heuristic is able to exploit the strengths of individual heuristics to perform well on distinct sets of instances.

Even though the results are promising, more work is needed regarding the features used to characterize the instances. In this investigation we used the constraint density (p_1), constraint tightness (p_2) and kappa (κ), but we consider that more features are needed to improve the mapping from instances to heuristics. We think the most important idea to be addressed in the future is the analysis of other relevant CSP features that could lead to a better classification of the instances and the solving methods according to those features.

We also observed that there are opportunities to improve the approach in the way we select the best heuristic for a given instance. In this investigation we produced a set of examples by mapping the instance features to the heuristic that required the fewer consistency checks on that instance. By using this idea we concluded that WDEG was the best heuristic. Nevertheless WDEG was the heuristic with the largest proportion of best results, it was not the heuristic with the minimum average cost per instance. Then, it may be a good idea to explore other alternatives to create the training examples. This is left as part of the future work.

Finally, we are interested in testing our approach on other classes of instances. For example, we would like to apply it to real problems such as scheduling and timetabling, and some optimization problems from vision and biology. This will raise the question of whether one heuristic exists that dominates the others for each specific problem domain.

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: Burke, E.K., Rudová, H. (eds.) PATAT 2007. LNCS, vol. 3867, pp. 394–412. Springer, Heidelberg (2007)
2. Bitner, J.R., Reingold, E.M.: Backtrack programming techniques. Communications of the ACM 18(11), 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, pp. 2209–2212. ACM (2009)
4. Boussemart, F., Hemery, F., Lecoutre, C., Sais, L.: Boosting systematic search by weighting constraints. In: European Conference on Artificial Intelligence (ECAI 2004), pp. 146–150 (2004)
5. 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 (2003)
6. Burke, E.K., Hyde, M., Kendall, G., Ochoa, G., Özcan, E., Woodward, J.R.: A classification of hyper-heuristic approaches. In: Gendreau, M., Potvin, J.Y. (eds.) Handbook of Metaheuristics. International Series in Operations Research and Management Science, vol. 146, pp. 449–468. Springer (2010)
7. Cowling, P., Kendall, G., Soubeiga, E.: Hyperheuristics: A robust optimisation method applied to nurse scheduling. In: Guervós, J.J.M., Adamidis, P.A., Beyer, H.-G., Fernández-Villacañas, J.-L., Schwefel, H.-P. (eds.) PPSN 2002. LNCS, vol. 2439, pp. 851–860. Springer, Heidelberg (2002)
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 (1994)
10. Freuder, E.C.: Synthesizing constraint expressions. Communications of the ACM 21(11), 958–966 (1978)
11. Gaschnig, J.G.: A general backtrack algorithm that eliminates most redundant tests. In: Proceedings of the 5th International Joint Conference on Artificial Intelligence, vol. 1, p. 457. Morgan Kaufmann Publishers (1977)
12. Gent, I., MacIntyre, E., Prosser, P., Smith, B., Wals, T.: An empirical study of dynamic variable ordering heuristics for the constraint satisfaction problem. In: Freuder, E.C. (ed.) CP 1996. LNCS, vol. 1118, pp. 179–193. Springer, Heidelberg (1996)
13. Haralick, R.M., Elliott, G.L.: Increasing tree search efficiency for constraint satisfaction problems. Artificial Intelligence 14, 263–313 (1980)
14. Kumar, V.: Algorithms for constraint satisfaction: a survey. AI Magazine 13(1), 32–44 (1992)
15. Lecoutre, C., Boussemart, F., Hemery, F.: Backjump-based techniques versus conflict-directed heuristics. In: Proceedings of the 16th IEEE International Conference on Tools with Artificial Intelligence, ICTAI 2004, pp. 549–557. IEEE Computer Society, Washington, DC (2004)
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. Ortiz-Bayliss, J., Terashima-Marín, H., Conant-Pablos, S.: Neural networks to guide the selection of heuristics within constraint satisfaction problems. In: Martínez-Trinidad, J., Carrasco-Ochoa, J., Ben-Youssef Brants, C., Hancock, E. (eds.) MCPR 2011. LNCS, vol. 6718, pp. 250–259. Springer, Heidelberg (2011)
19. Ortiz-Bayliss, J.C., Özcan, E., Parkes, A.J., Terashima-Marín, H.: Mapping the performance of heuristics for constraint satisfaction. In: Proceedings of the 2010 IEEE Congress on Evolutionary Computation (CEC 2010), pp. 1–8. IEEE Press (2010)
20. Ortiz-Bayliss, J.C., Terashima-Marín, H., Conant-Pablos, S.E.: Learning vector quantization for variable ordering in constraint satisfaction problems. Pattern Recogn. Lett. 34(4), 423–432 (2013)
21. Özcan, E., Bilgin, B., Korkmaz, E.E.: A comprehensive analysis of hyper-heuristics. Intelligent Data Analysis 12(1), 3–23 (2008)
22. Purdom, P.W.: Search rearrangement backtracking and polynomial average time. Artificial Intelligence 21, 117–133 (1983)
23. 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)
24. Russell, S., Norvig, P.: Artificial Intelligence A Modern Approach. Prentice Hall (1995)
25. Smith, B.M.: Locating the phase transition in binary constraint satisfaction problems. Artificial Intelligence 81, 155–181 (1996)
26. Storer, R.H., Wu, S.D., Vaccari, R.: New search spaces for sequencing problems with application to job shop scheduling. Management Science 38(10), 1495–1509 (1992)
27. Terashima-Marín, H., Ortiz-Bayliss, J.C., Ross, P., Valenzuela-Rendón, M.: Hyper-heuristics for the dynamic variable ordering in constraint satisfaction problems. In: Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation (GECCO 2008), pp. 571–578. ACM (2008)
28. Terashima-Marín, H., Ross, P., Valenzuela-Rendón, M.: Evolution of constraint satisfaction strategies in examination timetabling. In: Proceedings of the Genetic and Evolutionary Computation Conference (GECCO 1999), pp. 635–642. Morgan Kaufmann (1999)
29. Tsang, E., Kwan, A.: Mapping constraint satisfaction problems to algorithms and heuristics. Tech. Rep. CSM-198, Department of Computer Sciences, University of Essex (1993)

Social Interaction Discovery: A Simulated Multiagent Approach

José C. Carrasco-Jiménez, José M. Celaya-Padilla, Gilberto Montes, Ramón F. Brena, and Sigfrido Iglesias

Instituto Tecnológico y de Estudios Superiores de Monterrey, Monterrey NL, México
{jc.carrasco.phd.mty,A00811434,A00808911,
ramon.brena,sigfrido}@itesm.mx

Abstract. Social interaction inference is a problem that has been of interest in the past few years. The intrinsic mobility patterns followed by humans present a number of challenges that range from interaction inference to identification of social relationships linking individuals. An intuitive approach is to focus on the similarity of mobility patterns as an indicator of possible social interaction among individuals. By recording the access points observed at each unit of time along with the strength of the signals received, individuals may be group based on similar walking patterns shared on space and time. In this paper, an implementation of a multiagent simulation of a University-like environment is tested using NetLogo and a methodology that consists of two phases: 1) Cluster Analysis and 2) Construction of Social Networks is used to discover possible interactions among individuals. The first phase consists of a number of clustering methods that are used to identify individuals that are more closely related given the characteristics that describe their mobility patterns obtained from simulated Wifi data. In the second phase, users belonging to the same cluster are linked within a social network, meaning that there is possible ongoing social interaction or tie that might link the individuals.

1 Introduction

Social interactions can be described in terms of different context variables (e.g. physical proximity, association affiliation, online profiles, etc.). Discovering physical interactions is a challenging task given the dynamic characteristic of human interactions. For this particular work, social interaction is defined as the process of influencing on each other's mobility behaviours due to physical proximity.

Analyzing interaction information from a simulated world environment allows us to control the environment and as a consequence to validate the results obtained, therefore a NetLogo simulation was designed. In this simulation environment, it is possible to configure a Univeristy-like environment with features such as access points, and software agents behaving like real people. The user interface allows us to modify the conditions of the environment in order to study grouping patterns.

By analyzing mobility patterns followed by individuals, we can gain valuable information about the context of users' social world [1]. Social networks are constructed

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 294–303, 2013.

from the empirical evidence of pattern similarity discovered by applying cluster analysis to the Wifi context data. The groups of people that are linked in the social network, are individuals who are most likely to be linked by a social interaction.

Clustering algorithms have been chosen to segment individuals that might be linked by a probable interaction due to their effectiveness in identifying distinct groups of individuals based on their characteristics (e.g. the signals received from the different access points that surround the context) [2]. In other words, objects that are grouped in the same cluster, are more closely related to one another than objects that are grouped in different clusters [3].

The k-means algorithm is a widely used method for data segmentation or grouping when a number of k groups is known; it assigns an observation to a cluster with the nearest mean. For this research project, we tested the k-means implementation supplied in R software environment (as described by Hartigan and Wong) [4] with $k = 4$, since the experiments were initialized with 4 groups. In the k-medoids clustering algorithm the centers for the clusters are restricted to be one of the observations that were assigned to the cluster, i.e., it is more robust in the presence of outliers. The 'fpc'R package [5] has an implementation of an enhanced version called pamk(), which calls pam and clara algorithms for the partitioning around medoids clustering method, as introduced by Kaufman and Rouseeuw [5].This function does not require a k value, instead, it uses the CLARA algorithm to do the partitioning around medoids to estimate an appropriate k value. The DBSCAN clustering algorithm, as introduced by Ester et al. [5], is based on the notion of density of data points. The idea of this algorithm is that a group has to contain at least a minimum number of points (minpoints) within a given radius (eps); in other words, the density of the neighborhood has to exceed some threshold [6].

This paper is organized as follows. Section 2 describes a number of related works followed by Section 3, which describes the simulation implemented in NetLogo platform . In Section 4, there is an overview of the methodology proposed to discover social group interaction followed by experimental results described in Section 5. A brief conclusion is given in Section 6, and future work is described in Section 7.

2 Related Work

The field of computing sciences has given birth to a vast number of applications that allow us to understand human dynamics. Although research in this field is very recent, some approaches to understand complex human dynamics have been made. In a study led by Eagle et al. [7], they show that it is possible to infer friendships based solely on observational data exploiting Bluetooth traces, cellular data, phone data (e.g. SMS, call records), as sources of social information. On the other hand, Mokhtar et al. [8] propose a middleware service that aims to combine both social and physical interactions in order to identify encounters, i.e. interactions between individuals. Encounters are logged from Bluetooth radio conectivity. In a research work [1], Cranshaw et al. introduce an analysis of GPS location for contextual features of human location trail data, to elicit the existance of a physical social network and to analyze the context of the

social interactions. In [9,10], Xu et al. collected and analyzed Wifi data to explore the relationship between physical proximity and social links between individuals. In this work, the physical position of the individuals was estimated from Wifi signal strength and MAC address of nearby Wifi access points. Then, proximity encounters between individuals were computed based on the distance threshold and duration threshold. The positioning of individuals was accomplished by performing a site survey that involves recording the Wifi signal strengths and access points of all floors in the building on the floor map [10]. Nevertheless, this approach requires the construction of radio maps a priori, which makes it unsuitable for highly dynamic scenarios.

3 The Simulation

A university-like environment was designed using NetLogo platform [11]. This multiagent simulation environment offers the tools to simulate, among many other types of environments, social interactions among software agents simulating individuals and access points.

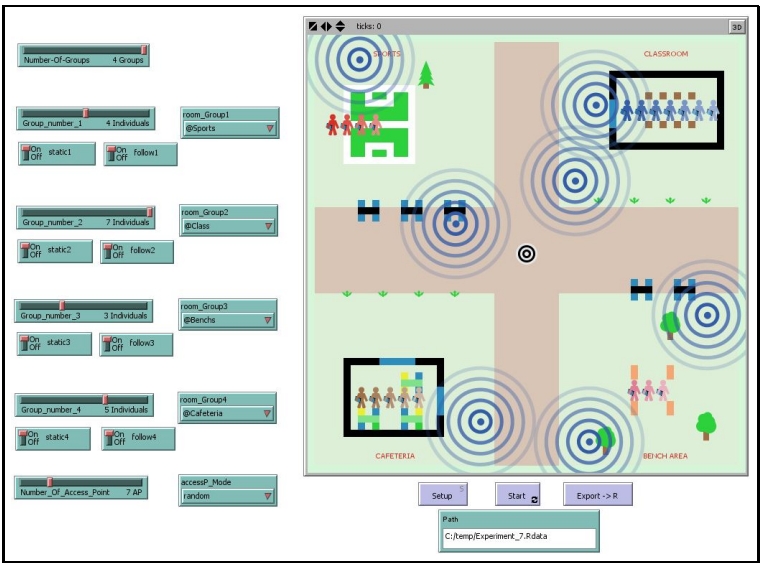
A control panel, as shown in Fig. 1, allows to modify initial settings in order to simulate different social scenarios with different mobility patterns. Among the different configurations, the user can choose the number of initial groups of individuals. On the other hand, mobility pattern generation can be adjusted for each of the initial groups, allowing the user to choose the number of individuals per group, the starting location for each group, as well as the walking patterns. The walking patterns are defined for each initial group, allowing the group to be either static or dynamic. Those groups tha are dynamic can either follow their own path (i.e. random walking), or can follow a leader (i.e. group walking).

Besides the group behaviors described previously, access point configurations can also be adjusted. The simulation control panel also allows the user to choose the number of access points (up to 25) that sorround the environment. In order to generate a more realistic set of Wifi data, the access points generate data following the theoretical signal propagation model shown in (1) and described in [12]. However, in real life scenarios, signals are constantly affected by external factors such as interference from signals, temperature, obstacles and other factors, causing signal degradation. In order to compensate for such factors of signal degradation, Gaussian noise was added to the data generated by the simulated access points, since Bose et al. [13] show that when large samples of RSS are collected over time, the noise is normally distributed among the samples.

$$RSS = -(10nlog_{10}d + A) \tag{1}$$

where:

- n: signal propagation constant (n = 3)
- d: distance from sender.
- A: received signal strength at a distance of one meter.



(a)

Fig. 1. Environment and Control Panel

4 Methodology

4.1 Social Group Discovery

Our approach discovers social group interactions in 2 phases:

1. **Cluster analysis:** in this phase, simulated Wifi data is segmented using the three clustering algorithms described in Section 1. The clustering is based on the Received Signal Strength (RSS) observed by the closest access points (APs) at each unit of time.
2. **Construction of social networks:** the groups of individuals obtained from the cluster analysis can be seen as a social network, where the individuals who belong to the same cluster are linked among themselves forming a social network. Similar movement patterns can be associated to possible physical interaction given the physical proximity through time.

4.2 Data Sets

The data sets analyzed for this project are collections of simulated context data that includes Wifi readings. The data collected contains four types of information with the following signature: wifi{User, Timestamp, MacAddress, RSS}. This signature contains information about the individual that collects the data, the time at which it was collected, the unique ID of the access points observed, and the signal strengths of each observed access point respectively.

An intuitive assumption made in this project is that if two users are close to each other at the same time, i.e. same timestamp, they will observe similar access points with the same MacAddresses and similar RSS (Signal Strength) readings. In other words, if a group of users walk together, they will observe the same movement patterns, detecting the same access points with similar signal strengths as time passes.

4.3 Experimental Settings

For the study of social interaction based on clustering methods, we performed 16 experiments with different initial configurations. All the experiments initially contain four groups. Each group was located at one of the four available initial locations. The numbers of individuals per group was regularly changed from one experiment to another. Table 1 includes the set of experiments taken for this study. In this table, Static refers to the number of groups (i.e. initial number of groups of individuals) that were stationary. Follow-Leader refers to the number of groups that moved together (i.e. following a leader), and Agents refers to the number of simulated individuals that were part of the experiment.

Table 1. Experiment configuration details

	Movement Patterns			Data Set
Experiment	Static	Follow-Leader	Agents	Instances
1	4	0	19	3800
2	4	0	14	2800
3	2	2	18	3600
4	0	4	13	2600
5	2	0	21	4200
6	0	0	12	2400
7	0	0	8	1600
8	2	2	18	3600
9	1	3	18	3600
10	0	0	4	800
11	0	0	5	1000
12	0	4	8	1600
13	0	0	4	800
14	0	0	8	1600
15	0	4	15	3000
16	0	1	15	3000

The initial setup of the simulations show different conditions under which social interactions might take place in real life. Since every single group of people simulated in the social environment has its own selective control, it is possible to change the behavior of each group. For example: it is possible to have one group with no motion at all, while the individuals of a second group may move randomly around, and the other two groups could follow the walking patterns of their leader.

5 Experimental Results

The experimental phase consists of a set of 16 experimental configurations simulating different conditions for social interaction, as described in Sect. 4. For all the experiments, three clustering algorithms were tested: k-means, k-medoids, and DBSCAN. The results obtained for the three algorithms are presented in Table 2.

The k-means algorithm [4] was tested with a parameter $k = 4$ for all the experiments since all of them consist in initial settings of 4 groups. The error rate, computed as shown in (2), was used as a measure of performance of the algorithm. The number of misclassified individuals was computed from visual inspection of the movement patterns observed on the individuals, i.e. we used observational validation. Execution time was also considered in order to see how the algorithm behaved as the number of instances varied. As a result, the error rate for the k-means algorithm, as shown in Table 2, was observed to be 0.246 in average. On the other hand, the execution time for this algorithm was 4.669s in average.

$$ER = \epsilon/\omega \tag{2}$$

where:

- ϵ is the number of misclassified individuals
- ω is the total number of individuals.

k-medoids algorithm implementation [5] showed an average error rate of 0.209 and an execution time of 4.902s, as presented in Table 2. Although k-medoids algorithms require the number of clusters (i.e. k value), the algorithm implementation provided in *R* programming environment [5] provides an enhanced way of selecting the number of clusters automatically based on the input data.

The DBSCAN algorithm implemented in [5] was also tested on the same datasets. This algorithm requires two parameters, the minimum number of points in a cluster, which was set to 2 (minpoints = 2), and the radius around which to search for data points to be added to the same cluster, which was obtained empirically for most of the data sets to be 0.22 (eps = 0.22). For this algorithm, the average error rate was 0.336 and the execution time was 4.642s, as we can see in Table 2.

Table 2 shows that all three algorithms have a similar average execution time. The average error rate shows that k-means and k-medoids algorithms outperform the density-based DBSCAN algorithm. The k-medoids shows a better average performance over the other two algorithms, although the k-means performs almost as well as the k-medoids. The k-means works well for controlled environments, but when the context is very dynamic, as the study case presented in this work, the k value has to be adjusted. In other words, there is a need for a method that estimates the appropriate k value that is adjusted to the model representing the environment.

Among the advantages of the k-medoids algorithm implementation [5] we find the fact that it does not require a predefined k value, and that it outperforms the other two algorithms when two characteristics of the context apply: 1) the environment is dynamic, 2) the distribution of Wifi access point signals cover most of the environment under study. It is important to note that when there are many uncovered regions, users

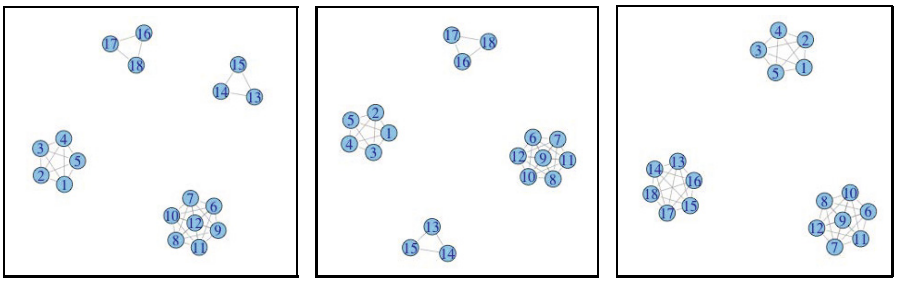
Table 2. Experimental Results

	K-Means			K-Medoids			DBSCAN		
Exp. #	k val.	Error Rate	Exec. Time (s)	k val.	Error Rate	Exec. Time (s)	k val.	Error Rate	Exec. Time (s)
1	4	0	5.152	4	0	4.998	4	0	4.759
2	4	0	3.871	4	0	4.041	4	0	3.885
3	4	0	4.758	4	0	4.917	2	0.270	4.749
4	4	0	3.863	4	0	3.811	4	0	3.953
5	4	0.333	5.510	4	0	5.739	2	0.285	5.392
6	4	0.250	3.790	4	0.333	3.942	4	0.250	3.737
7	4	0.250	2.825	8	0.875	2.877	1	1	2.939
8	4	0	4.889	4	0	5.332	3	0.166	4.714
9	4	0.277	4.780	4	0	5.077	2	0.333	4.813
10	4	0.277	4.898	4	0	5.230	2	0.333	4.973
11	4	0	1.866	1	1	2.057	1	1	1.922
12	4	0	2.354	4	0	6.107	2	0.5	2.366
13	4	0	6.084	2	0.875	3.453	4	0	6.016
14	4	0	4.554	4	0	5.936	1	1	5.120
15	4	0	6.361	4	0.266	7.423	4	0.25	6.114
16	4	0.066	9.158	4	0	7.497	4	0	8.821
Avg. Error Rate:		0.246	–		0.209	–		0.336	–
Avg. Exec. Time (s):		–	4.669		–	4.902		–	4.642

can be grouped in the same cluster even when they are far away from each other. The reason for this is that users are grouped based on the observed access points, and when no access point is in sight, a value of 0 is assigned to the user at that point in time. As a consequence, when two individuals have long trajectories in areas without any signal coverage, they will report high similarities even when they are far away from each other.

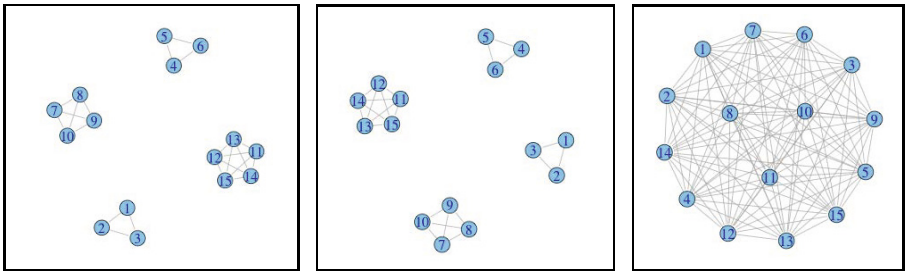
Some of the benefits of grouping individuals based on observed access points and RSS include robustness to noise in Wifi readings, and the ability to apply the methodology in real time, since the results are not affected if the distribution of access points is modified as oppose to methods that require radio maps to be constructed a priori. The results were validated using observation of trajectory traces followed by the human-like agents in the simulation.

Figure 2 shows the resulting social network after running the three algorithms, mentioned above, on the data collected in experiment number 8, described in Table 1. Experiment 8 is a controlled environment where 2 of the groups are static and 2 groups move randomly but following the traces of a leader. As a consequence there are 4 groups perfectly identifiable and separated from each other as shown in Fig. 2 (a) and (b). The image shows the social network constructed using the "igraph" R package [14], based on the clusters identified by each of the three clustering algorithms revised in this project. As it has been mentioned in the analysis of the results obtained for each clustering method, the k-medoids and k-means perform well in average, both having similar results. Figure 2(c) shows that the DBSCAN algorithm has a similar set of clusters than those identified by the other two algorithms. The only difference is that two of



(a) (b) (c)

Fig. 2. Experiment #8 a) K-means. b) K-medoids. c) DBSCAN.

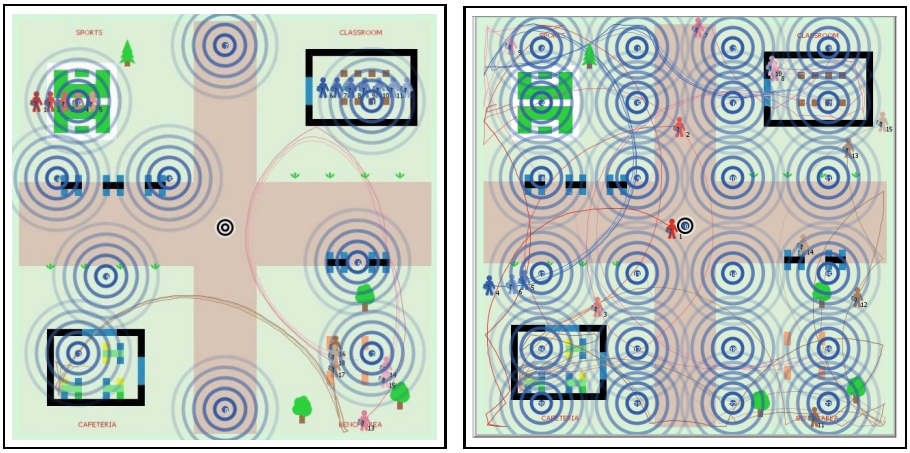


(a) (b) (c)

Fig. 3. Experiment #16 a) K-means. b) K-medoids. c) DBSCAN.

the clusters are merged into one, which means that it misclassified some of the individuals. For this specific scenario, DBSCAN's grouping is not far from the known groups. The scenario describing the patterns followed by the individuals using the settings of experiment 8 can be seen in Fig. 4 (a).

On the other hand, Fig. 3 shows the social networks constructed from the results of experiment 16 shown in Table 1. Experiment 16 consists of 4 non-fixed random groups, one of which follows a leader's trace. For this particular scenario, both k-means and k-medoids were able to find the same clusters while DBSCAN was unable to find the clusters at all, as it is shown in Fig. 3 (c). From visual inspection, we conclude that the groups of individuals found by k-means and k-medoids are the most likely groups given the movement patterns followed by the groups of individuals. Even though this experiment consists of 3 groups walking randomly, 4 groups were detected due to the amount of time they were close to each other, which is the time it took the individuals to leave the cafeteria and classroom. The scenario of experiment 16 is shown in Fig. 4 (b).



(a) (b)

Fig. 4. Experimental Scenarios a) Experiment #8. b) Experiment #16.

6 Conclusions and Future Work

The NetLogo platform contains suitable characteristics to model and simulate mobility patterns in a bidimensional space. Cluster analysis is an efficient way to group individuals based on similar mobility patterns described by contextual Wifi data. Out of the three algorithms we tested in this project, k-means and k-medoids performed fairly well on the data sets obtained from the NetLogo simulations. One of the advantages of k-medoids over k-means, as implemented in [5], is that k-medoids can estimate the number of clusters while k-means requires the user to adjust the parameter manually. The resulting clusters of individuals represent discovered social groups that are likely to be linked by a social interaction. We were also able to identify the k-medoids as the best option for grouping individuals of similar movement patterns in the simulated environment when two conditions applied: 1) dynamic environment 2) high Wifi signal coverage. Also, it is possible to tell that the clustering methodology is robust to the presence of Wifi signal noise because it does not depend on a predefined distribution of access points. The signal noise robustness should make this clustering approach suitable for applications that require real-time analysis.

Real world mobility patterns can be hard to analyze due to the difficulty in controlling the context as well as the restriction on the amount of experiments. From this perspective, having a controlled environment allow us to analyze the power of the proposed methodology and to learn about the expected behavior of the clustering strategies prior to test it in a real and more complex scenario.

As future work, we propose to apply the methodology followed in this paper on data collected on real scenarios. A deeper analysis of unsupervised learning algorithms can also be tested on Wifi data to discover group interactions (e.g. FRiS-Tax, Hierarchical, etc.). Other types of measures to estimate the error rate can also be investigated, furthermore different type of sensors (e.g. Bluetooth, GPS, Magnetic sensors) and online

social network information (e.g. Facebook, Twitter, Foursquare) can be incorporated to improve the cluster accuracy.

References

1. Cranshaw, J., Toch, E., Hong, J., Kittur, A., Sadeh, N.: Bridging the gap between physical location and online social networks. In: Proceedings of the 12th ACM International Conference on Ubiquitous Computing, Ubicomp 2010, pp. 119–128. ACM, New York (2010)
2. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning. Springer Series in Statistics. Springer New York Inc., New York (2001)
3. Witten, I., Frank, E., Hall, M.: Data Mining: Practical Machine Learning Tools and Techniques: Practical Machine Learning Tools and Techniques. The Morgan Kaufmann Series in Data Management Systems. Elsevier Science (2011)
4. R Core Team: R: A Language and Environment for Statistical Computing. R Foundation for Statistical Computing, Vienna, Austria (2012) ISBN 3-900051-07-0
5. Hennig, C.: fpc: Flexible procedures for clustering, R package version 2.1-4 (2012)
6. Patwary, M.A., Palsetia, D., Agrawal, A., Liao, W.K., Manne, F., Choudhary, A.: A new scalable parallel dbscan algorithm using the disjoint-set data structure. In: Proceedings of the International Conference on High Performance Computing, Networking, Storage and Analysis, SC 2012, pp. 62:1–62:11. IEEE Computer Society Press, Los Alamitos (2012)
7. Eagle, N., Pentland, A., Lazer, D.: From the cover: Inferring friendship network structure by using mobile phone data. Proceedings of The National Academy of Sciences 106, 15274–15278 (2009)
8. Mokhtar, S.B., McNamara, L., Capra, L.: A middleware service for pervasive social networking. In: Proceedings of the International Workshop on Middleware for Pervasive Mobile and Embedded Computing, M-PAC 2009, pp. 2:1–2:6. ACM, New York (2009)
9. Xu, B., Chin, A., Wang, H., Wang, H., Zhang, L.: Social linking and physical proximity in a mobile location-based service. In: Proceedings of the 1st International Workshop on Mobile Location-Based Service, MLBS 2011, pp. 99–108. ACM, New York (2011)
10. Zhu, L., Chin, A., Zhang, K., Xu, W., Wang, H., Zhang, L.: Managing workplace resources in office environments through ephemeral social networks. In: Yu, Z., Liscano, R., Chen, G., Zhang, D., Zhou, X. (eds.) UIC 2010. LNCS, vol. 6406, pp. 665–679. Springer, Heidelberg (2010)
11. Tisue, S., Wilensky, U.: Netlogo: A simple environment for modeling complexity. In: International Conference on Complex Systems, pp. 16–21 (2004)
12. CC2431 Location Engine
13. Bose, A., Foh, C.H.: A practical path loss model for indoor wifi positioning enhancement. In: 2007 6th International Conference on Information, Communications Signal Processing, pp. 1–5 (December 2007)
14. Csardi, G., Nepusz, T.: The igraph software package for complex network research. Inter. Journal Complex Systems 1695 (2006)

Associative Model for the Forecasting of Time Series Based on the Gamma Classifier

Itzamá López-Yáñez[1,2], Leonid Sheremetov[1], and Cornelio Yáñez-Márquez[3]

[1] Mexican Petroleum Institute (IMP),
Av. Eje Central Lázaro Cárdenas Norte, 152, Mexico City, Mexico
{itzamal,sher}@imp.mx
[2] Instituto Politécnico Nacional,
Centro de Investigación y Desarrollo Tecnológico en Cómputo (CIDETEC - IPN),
Av. Juan de Dios Bátiz s/n Edificio CIDETEC, Mexico City, Mexico
[3] Instituto Politécnico Nacional, Centro de Investigación en Computación
(CIC - IPN), Av. Juan de Dios Bátiz s/n Edificio CIC, Mexico City, Mexico
cyanez@cic.ipn.mx

Abstract. The paper describes a novel associative model for the forecasting of time series in petroleum engineering. The model is based on the Gamma classifier, which is inspired on the Alpha-Beta associative memories, taking the alpha and beta operators as basis for the gamma operator. The objective is to reproduce and predict future oil production in different scenarios in an adjustable time window. The distinctive features of the experimental data set are spikes, abrupt changes and frequent discontinuities, which considerably decrease the precision of traditional forecasting methods. As experimental results show, this classifier-based predictor exhibits competitive performance. The advantages and limitations of the model, as well as lines of improvement, are discussed.

Keywords: Time series forcasting, associative models, oil production time series, Gamma classifier.

1 Introduction

Time series (TS) analysis has become a relevant tool for understanding the behavior of different processes, both naturally ocurring and human caused [1]. One of the latter kind of processes is the study of oil production through time, more specifically in fractured oil reservoirs, given their non-homogenous nature [2]. One of the tasks involved in such TS analysis is the prediction of future values (also known as TS forecasting), which is of particular interest in the context of industrial processes.

Computational Intelligence and Machine Learning have become standard tools for modeling and prediction of industrial processes in recent years, contributing models related mainly to Artificial Neural Networks (ANN) [3,4]. On the other hand, classical approaches, such as the Box-Jenkins Auto-Regressive Moving Average (ARMA) models, are still widely in use [1]. However, TS which exhibit non-linear, complex behavior tend to pose difficulties to such methods.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 304–313, 2013.

In the current paper, a novel associative model for the forecasting of TS is proposed. This method is based on the Gamma classifier (GC) [8], which is a supervised learning pattern classifier of recent emergence, belonging to the associative approach to Pattern Recognition. The GC has been previously applied to forecast atmospheric pollution TS [8], exhibiting a quite promising performance. However, that manner of forecasting had some limitations. In particular, it was able to predict only the following sample, given a known section of the TS.

This paper extends the previous work in order to predict a whole fragment of a TS, as well as to forecast samples located towards the past of a known fragment, allowing for a more complete reconstruction. The proposed method is applied to the prediction of several TS of oil production at a Mexican fractured oil reservoir, as well as to the synthetically generated Mackey-Glass TS, commonly used as a benchmark for TS forecasting.

The rest of the paper is organized as follows. Section 2 describes the TS used, while section 3 presents the GC, which is the basis for the proposal. The method presented here is further discussed in section 4, while section 5 introduces the experimental results and their discussion, and the conclusions and future work are included in the final section.

2 Oil Production Time Series

Since the beginning of petroleum engineering history, it has been a concern to forecast how long the production will last, sparking an ongoing interest for methods to predict production [4]. Yet, forecasting oil production is not an easy task. Production TS have several distinctive features regarding their predictability, which separate them from financial TS or physical processes TS, which are usually used for forecasting competitions [5].

First, monthly production TS are rather short: for a 30 years old brown oilfield, the longest series have about 300 data points; the rest of the series are shorter covering only several years or even months of production, so the data sets may only be a few dozens points long. Moreover, many TS are discontinuous: there are periods when for some reason the well is shut. Usually, a data set is predictable on the shortest time scales, but has global events that are harder to predict (spikes). These events can be caused by workovers, change of producing intervals and formations and so on. Some of these features (e.g. spikes, abrupt changes, discontinuities, and different trend segments) are evident in the plots of the TS used for experimentation, shown in figures 1 and 2. As can be seen, such TS lack typical patterns such as periodicity, seasonality, or cycles. Even though direct measurements are possible, the sampling process for these data is noisy.

Two data sets were used for the experiments: the first consisting of 6 randomly selected TS related to the monthly oil production and 6 wells in a Mexican oilfield, while the second is a synthetically generated TS coming from Mackey-Glass delay differential equations (MG equation) [6].

Table 1 summarizes statistical characteristics of the selected TS belonging to the first data set. TS forecasting is usually made on a one-year basis. Though the

Table 1. Statistical characteristics of the 6 selected TS of the first data set

Characteristic	TS1	TS2	TS3	TS4	TS5	TS6
Data points	154	296	126	131	139	140
Linear trend	$-264.52x$ $+79625$	N/A	$-84.437x$ $+89559$	$-0.0927x$ $+9590.6$	$-645.83x$ $+96228$	$-763.69x$ $+111875$
R^2	0.27	N/A	0.02	0.00	0.67	0.43
Mean	59,366.91	46,083.38	84,440.67	6,692.23	58,381.15	55,221.67
Std. deviation	24,245.14	50,872.73	21,880.89	2,118.10	29,912.79	45,683.36
Mode	86,185.58	14,819.24	97,495.00	6,824.65	81,493.00	72,007.00
Skewness	-0.81	1.11	-0.58	0.57	-0.14	0.68
Kurtosis	-0.34	-0.31	2.64	1.67	-1.07	-0.10
Normality	0.89	0.87	0.92	0.92	0.95	0.90
Partial ACF	0.90	0.98	0.76	0.65	0.91	0.96

selected data set covered in all cases more than 10 years periods, discontinuities found almost in all TS substantially reduced the training basis.

To check for normality, the Shapiro-Wilk test was applied [7]. As can be seen from table 1, the normality test statistic (W) values for TS1 and TS2, though close but do not pass the test (W should be above 0.90 for normality). Many models used in TS analysis, including ARMA models, assume stationarity. To check for stationarity both parametric (partial autocorrelation function, ACF) and nonparametric (runs) tests were applied. Though nonparametric tests for stationarity are more applicable for our dataset since TS1 and TS2 do not meet the normality assumption, the auto-correlation does seem to agree with results of the nonparametric runs test. Partial ACF plots (calculated for 18 lags) show similar results for most of the wells. Most of the autocorrelation is actually just a lag 1 (1 month) effect (shown in table 1), with minor effects at other lags (maximum of 0.32 for lag of 6 months for TS6). The results indicate that there is actually little pattern in the data —though having a distinct decline trend, they have very little periodicity, and only very short term, month-to-month correlation between observations.

3 Gamma Classifier

This supervised learning associative model was originaly designed for the task of pattern classification, and borrows its name from the operation at its heart: the generalized Gamma operator [8,9]. This operation takes as input two binary patterns — $\mathbf{x}$ and $\mathbf{y}$— and a non-negative integer θ and gives a 1 if both vectors are similar (at most as different as indicated by θ) or 0 otherwise. Given that the Gamma operator uses some other operators (namely α, β, and u_β), they will be presented before. The rest of this section is strongly based on [8,9].

Definition 1 (Alpha and Beta operators). *Given the sets* $A = \{0, 1\}$ *and* $B = \{0, 1, 2\}$*, the alpha (*α*) and beta (*β*) operators are defined in a tabular*

form as shown in table 2. The corresponding vector versions of both operators for inputs $\mathbf{x} \in A^n$, $\mathbf{y} \in A^n$, *and* $\mathbf{z} \in B^n$ *give an n-dimensional vector as output, whose i-th component is computed as follows.*

$$\alpha\left(\mathbf{x}, \mathbf{y}\right)_i = \alpha\left(x_i, y_i\right) \quad \textit{and} \quad \beta\left(\mathbf{z}, \mathbf{y}\right)_i = \beta\left(z_i, y_i\right) \tag{1}$$

Definition 2 (u_β operator). *Considering the binary pattern* $\mathbf{x} \in A^n$ *as input, this unary operator gives the following integer as output.*

$$u_\beta(\mathbf{x}) = \sum_{i=1}^{n} \beta(x_i, x_i) \tag{2}$$

Definition 3 (Gamma operator). *The similarity Gamma operator takes two binary patterns —*$\mathbf{x} \in A^n$ *and* $\mathbf{y} \in A^m$*;* $n, m \in \mathbb{Z}^+$ $n \leq m$*— and a non-negative integer* θ *as input, and outputs a binary number, according to the following rule.*

$$\gamma_g(\mathbf{x}, \mathbf{y}, \theta) = \begin{cases} 1 & \textit{if } m - u_\beta\left[\alpha(\mathbf{x}, \mathbf{y}) \mod 2\right] \leq \theta \\ 0 & \textit{otherwise} \end{cases} \tag{3}$$

where mod *denotes the usual modulo operation.*

The GC makes use of the previous definitions, as well as that of the Modified Johnson-Möbius code [10] in order to classify a (potentially unknown) test pattern $\tilde{\mathbf{x}}$, given a fundamental set of learning or training patterns $\{(\mathbf{x}^\mu, \mathbf{y}^\mu)\}$. To do this, it follows the algorithm described below.

1. Convert the patterns in the fundamental set into binary vectors using the Modified Johnson-Möbius code.
2. Code the test pattern with the Modified Johnson-Möbius code, using the same parameters used for the fundamental set.
3. Compute the stop parameter $\rho = \bigwedge_{j=1}^{n} \bigvee_{i=1}^{p} x_j^i$.
4. Transform the index of all fundamental patterns into two indices, one for their class and another for their position in the class (e.g. $\mathbf{x}^\mu$ in class i becomes $\mathbf{x}^{i\omega}$).

Table 2. The alpha (α) and beta (β) operators

$\alpha : A \times A \to B$		
x	y	$\alpha\left(x, y\right)$
0	0	1
0	1	0
1	0	2
1	1	1

$\beta : B \times A \to A$		
x	y	$\beta\left(x, y\right)$
0	0	0
0	1	0
1	0	0
1	1	1
2	0	1
2	1	1

5. Initialize $\theta = 0$.
6. Do $\gamma_g(x_j^{i\omega}, \tilde{x}_j, \theta)$ for each component of the fundamental patterns.
7. Compute a weighted sum c_i for each class, according to this equation:

$$c_i = \frac{\sum_{\omega=1}^{k_i} \sum_{j=1}^{n} \gamma_g(x_j^{i\omega}, \tilde{x}_j, \theta)}{k_i} \tag{4}$$

   where k_i is the cardinality in the fundamental set of class i.
8. If there is more than one maximum among the different c_i, increment θ by 1 and repeat steps 6 and 7 until there is a unique maximum, or the stop condition $\theta \geq \rho$ is fulfilled.
9. If there is a unique maximum among the c_i, assign $\tilde{\mathbf{y}}$ to the class corresponding to such maximum.
10. Oterhwise, assign $\tilde{\mathbf{y}}$ to the class of the first maxima.

The main characteristic of the GC, which sets it appart from other classifiers, is that the measure of similarity between patterns (on which the classification decision is based) is computed independently on each feature, and later integrated for the whole pattern. This fundamental difference allows this model to offer better performance than more conventional models on those problems for which computing the similarity using all features give rise to ambiguity. This property is useful to address such issues as spikes and abrupt changes: if the majority of features are similar enough among two patterns, the differences between two particular features have little bearing on the final outcome. Another defining and desirable characteristic of the GC is its low computational complexity, which is $O(pn)$ for a fundamental set of p n-dimensional learning patterns [8,9].

4 Proposed Model

As mentioned before, the GC has been already applied to forecast TS, but predicting only the following sample. The current proposal takes the previous work and extends it in order to predict not only the first unknown sample in a fragment of a TS, but the whole fragment (of arbitrary length). Also, it is now possible to forecast samples located towards the past of a known fragment (i.e. towards the left of the TS, or previous to the known segment), which allows for a more complete reconstruction. In order to achieve the fomer objectives, the coding and pattern building method introduced in [8] is generalized in order to consider negative *separations* between input and output patterns, as well as separations greater than one sample away.

Definition 4 (Separation). *Given a TS D with samples $d_1 d_2 d_3 \ldots$, the* separation s *between a segment $d_i d_{i+1} \ldots d_{n-1}$ (of length n) and sample d_j is given by the distance between the closest extreme of the segment and the sample.*

Example 1. Let D be the TS with the following sample values: $D = 10, 9, 8, 7, 6, 5, 4, 3, 2, 1$. Considering the segment D_1 of size 3 as $D_1 = d_3 d_4 d_5 = 8, 7, 6$ then sample $d_6 = 5$ is at a separation $s = 1$, sample $d_7 = 4$ is at $s = 2$, sample $d_{10} = 1$ is at $s = 5$, and sample $d_1 = 10$ is at $s = -2$.

Based on this definition and the GC, the proposed TS forecasting method follows the algorithm presented below, considering a TS D of length l with a prediction (test) segment of length t, and a length for the patterns of size n.

1. Starting from the TS D, calculate the differences between succesive samples in order to work with values relative to that of the previous sample. The new time series D' has length $l-1$.

$$D_{l\times 1} \longrightarrow D'_{(l-1)\times 1} \tag{5}$$

2. Build a set of associations between TS difference segments of length n and its corresponding difference with separation s, for both positive and negative values of $s = 1, 2, \ldots, t-1, t$.
   Thus, there will be $2t$ sets of associations of the form $\{\mathbf{a}^\mu, \mathbf{b}^\mu\}$ where $\mathbf{a} \in \mathbb{R}^n$ and $\mathbf{b} \in \mathbb{R}$, and the i-th association of the s-th set is made up by $\mathbf{a}^\mu = d_i d_{i+1} \ldots d_{i+n-1}$ and:

$$\mathbf{b}^\mu = \begin{cases} d_{i+n+s-1} & \text{if } s > 1 \\ d_{i-s} & \text{if } s < 1 \end{cases} \tag{6}$$

3. Train a different GC from each association set; there will be $2t$ different classifiers, each with a distinct fundamental set $\{\mathbf{x}^\mu, \mathbf{y}^\mu\}$ for its corresponding value of s.
4. Operate each GC with all the input segments $\mathbf{a}^\mu$.
5. When multiple classifiers give different output values $\tilde{\mathbf{y}}$ for the same data point in the differences TS D', there are two prominent alternatives to integrate them into one value $\tilde{\mathbf{y}}'$.
   (a) Average the values given by the two classifiers with the same absolute separation $|s| = \{-s, s\}$; this is denoted as the *combined* method.
   (b) Average the values given by all available classifiers; this is known as the *combined average* method.
6. Convert back to absolute values by adding the forecast relative value ($\tilde{\mathbf{y}}'$) to the original value of the previous sample, taken from D.

$$\tilde{\mathbf{y}}_i = D_{i-1} + \tilde{\mathbf{y}}'_i \tag{7}$$

Thus, the proposed method uses a classifier to obtain a particular value for each fundamental set (associated to a different separation), in order to later integreate these output values into a single value, which may or may not be known. With this method, a TS with no discontinuities will be reconstructed and predicted in its entirety, except for the first sample. This is due to the differences taken as the first step, which decreases the length of TS D' in one with respect to the original series D.

Also, given the guaranteed correct recall of known patterns exhibited by the GC [8], it is most likely that the known segments of the TS will be exactly reconstructed (i.e. error of 0). There will be a difference between the original and forecast values only if the same input pattern ($\mathbf{a}^i = \mathbf{a}^j$) appears more than once, associated to different output values ($\mathbf{b}^i \neq \mathbf{b}^j$).

$$\exists \left(\mathbf{a}^i, \mathbf{b}^i\right) \wedge \left(\mathbf{a}^j, \mathbf{b}^j\right), i \neq j \text{ such that } \mathbf{a}^i = \mathbf{a}^j \wedge \mathbf{b}^i \neq \mathbf{b}^j \tag{8}$$

5 Experimental Results

In order to test the proposed prediction model, it was applied to two data sets: one made up by several TS related to oil production, which were previously described, and the other a synthetically generated TS coming from the MG equation [6].

For comparison purposes, two optimality measures were used: the Mean Square Error (MSE) and the adjusted Mean Absolute Percent Error introduced by Armstrong, also known as symmetric MAPE (SMAPE, though it is not symmetric since over- and under-forecasts are biased). These error metrics are computed as shown in equations 9 and 10, where y_j is the actual and $\hat{y}_j$ is the predicted value, with N being the amount of samples considered.

$$E_{MSE} = \frac{1}{N}\sum_{i=1}^{N}(y_i - \hat{y}_i)^2 \tag{9}$$

$$E_{SMAPE} = \frac{1}{N}\sum_{i=1}^{N}\left|\frac{y_i - \hat{y}_i}{y_i + \hat{y}_i}\right| \tag{10}$$

5.1 Oil Production TS Experiment

For the first data set, three scenarios were considered, with test segments of 12, 18, and 24 samples of length, in order to find out how the length of the test segment affects prediction. Meanwhile, the experimetal results are summarized in table 3, and for illustrating purposes, the resulting monthly oil production predictions for two TS are shown in figures 1 and 2.

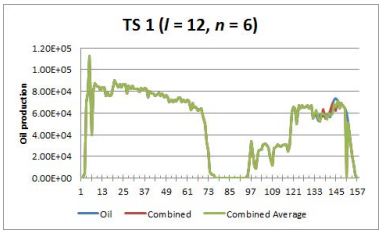


Fig. 1. Prediction of TS 1

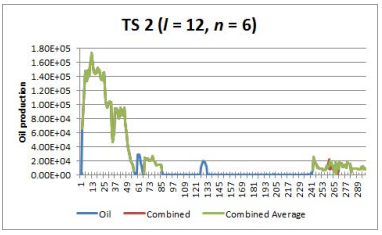


Fig. 2. Prediction of TS 2

As can be seen, the performance of the proposed method varies throughout the different experiments, as was expected given the differences presented by the TS used. TS 1 and 3 show very good results on both metrics, while TS 4 has very good MSE and reasonably good SMAPE. On the other hand, the metrics for TS 5 are large for both metrics. When looking at the plots, the reasons for such behaviors start to emerge. TS 1 has no discontinuities, while out of the 296 samples of TS 2, only 146 are valid data points grouped in 5 segments. Under

Table 3. Experimental results of oil production data for Mean Square Error in millions (MSE × 1.0E+06) and Symmetric Mean Absolute Percentage Error (SMAPE)

Mean Squared Error						
Time	Combined			Combined Avg.		
Series	l =12	l =18	l =24	l =12	l =18	l =24
TS 1	24.01	27.50	28.82	20.87	20.96	36.08
TS 2	33.76	38.11	105.28	48.39	64.32	60.49
TS 3	8.71	20.30	35.59	8.71	28.78	36.55
TS 4	0.99	3.43	4.06	1.19	2.60	5.62
TS 5	297.13	195.10	206.62	297.13	291.22	202.31
TS 6	28.05	64.07	4.12	29.48	82.83	15.08
Symmetric Mean Absolute Percentage Error						
Time	Combined			Combined Avg.		
Series	l =12	l =18	l =24	l =12	l =18	l =24
TS 1	0.0304	0.0349	0.0323	0.0304	0.0272	0.0366
TS 2	0.2252	0.3172	0.2908	0.3162	0.3172	0.2908
TS 3	0.0182	0.0248	0.0284	0.0182	0.0264	0.0328
TS 4	0.0508	0.1376	0.1046	0.0488	0.1140	0.1609
TS 5	0.1858	0.1321	0.1334	0.1858	0.1699	0.1215
TS 6	0.4329	0.3893	0.0711	0.3616	0.3574	0.1516

such conditions, it can only be expected that any predictor will have a hard time obtaining good results.

It is noteworthy that for all six TS, on both variants (*Combined* and *Combined average*) and both performance metrics, the best results were obtained with test segments of length $l = 12$. It is well-known in TS forecasting that the farther away from the learning samples, the greater the uncertainty in the prediction [1]. However, given the characteristics and properties of the GC (i.e. a segment and a sample with arbitrary separation could be associated and learned) the effect of l was previously unknown.

Another interesting finding is that some TS yield quite good results under some performance metrics, while the results for the other metric are surprisingly poor. Such is the case of TS 6, whose MSE (1.51E+07 best, 3.73E+07 average) is below the mean of the six TS (6.61E+07), while its SMAPE (0.2940 average, 0.4329 worst) is considerably above the mean (0.1502, respectively). Yet, each metric emphasizes different aspects of the prediction error, thus giving rise to such apparent discrepancies. What remains to be found is why such large inconsistencies in performance arise.

5.2 Mackey-Glass TS Experiment

On the other hand, the TS for the second data set is generated by the MG equation, which has been extensively used by different authors to perform comparisons between different techniques for forecasting and regression models. The MG equation is explained by the time delay differential equation defined as:

Table 4. Comparison of performance on the Mackey-Glass benchmark [6,11]

Model	RMSE
Gamma Classifier	**0.001502**
MLMVN	0.0063
CNNE	0.009
SuPFuNIS	0.014
GEFREX	0.0061
EPNet	0.02
GFPE	0.026
Classical BP NN	0.02
ANFIS	0.0074

$$y(t) = \frac{\alpha y(t-\tau)}{1+y^c(t-\tau)} - \beta y(t) \tag{11}$$

where α, β, and c are parameters and τ is the delay time. According as τ increases, the solution turns from periodic to chaotic. To make the comparisons with earlier work, one thousand data points are generated with an initial condition $x(0) = 1.2$ and $\tau = 17$ based on the fourth-order Runge-Kutta method with time step 0.1. The following parameters (taken from consulted literature) have been used as a benchmark: $\alpha = 0.2$, $\beta = 0.1$, $c = 10$ [6]. From these 1000 samples, the first 500 were used for training and the latter 500 points were used for testing. Rooted MSE (RMSE, the square root of the MSE) was used given that it is the performance measure of choice for this TS through the consulted literature, obtaining a RMSE of 0.001502. Table 4 includes a comparison against other models [6,11].

As can be seen, the proposed model based on the GC clearly outperforms the other techniques, giving an RMSE which is approximately four times smaller than that of GEFREX (0.001502 against 0.0061), its closest competition.

6 Conclusions and Future Work

In this paper the forecasting capabilities of an associative model are tested. The main result is that a pattern classifier is successfully applied to the task of TS prediction, though it is not the kind of tasks envisioned while designing and developing the GC. Moreover, the proposed method uses only one variable when there are more available, and the phenomenon under study is considered to be a rather complex one. Experiments with the GC-based predictor under multivariate settings for oil-production forecasting are on-going. One of such scenarios is the development of a hierarchichal model, which begins by detecting and classifying specific events, ir order to use a specific predictor.

On a different matter, the dependencies of forecasting uncertainty on separation between predicted data point and known data have been confirmed, despite

the independency offered by the proposed model. This indicates a likely dependency in the actual data sampled from the natural phenomenon, which remains to be tested.

Acknowledgements. This research was partially supported by the CONACYT-SENER (project 146515), as well as the Instituto Politécnico Nacional (Secretaría Académica, COFAA, SIP, CIDETEC and CIC), CONACyT, and SNI.

References

1. Schelter, B., Winterhalder, M., Timmer, J. (eds.): Handbook of Time Series Analysis. Wiley, Weinheim (2006)
2. van Golf-Racht, T.D.: Fundamentals of Fractured Reservoir Engineering, Developments in Petroleum Science, vol. 12. Elsevier, Amsterdam (1982)
3. Palit, A.K., Popovic, D.: Computational Intelligence in Time Series Forecasting. Springer, London (2005)
4. Sheremetov, L., Alvarado, M., Bañares-Alcántara, R., Anminzadeh, F.: Intelligent Computing in Petroleum Engineering (Editorial). J. of Petroleum Science and Engineering 47(1-2), 1–3 (2005)
5. He, Z., Yang, L., Yen, J., Wu, C.: Neural-Network Approach to Predict Well Performance Using Available Field Data. In: SPE Western Regional Meeting, Bakersfield, California, March 26-30, SPE 68801 (2001)
6. Kim, D., Kim, C.: Forecasting Time Series with Genetic Fuzzy Predictor Ensemble. IEEE Tr. on Fuzzy Systems 5(4), 523–535 (1997)
7. Johnson, R.A., Wichern, D.W.: Applied Multivariate Statistical Analysis (Prentice Hall series in statistics), 5th edn. Prentice Hall (2001)
8. López-Yáñez, I., Argüelles-Cruz, A.J., Camacho-Nieto, O., Yáñez-Márquez, C.: Pollutants Time-Series Prediction Using the Gamma classifier. Int. J. of Computational Intelligence Systems 4(4), 680–711 (2011)
9. Acevedo-Mosqueda, M.E., Yáñez-Márquez, C., López-Yáñez, I.: Alpha-Beta Bidirectional Associative Memories: Theory and Applications. Neural Processing Letters 26(1), 1–40 (2007)
10. Yáñez, C., Felipe-Riveron, E., López-Yáñez, I., Flores-Carapia, R.: A Novel Approach to Automatic Color Matching. In: Martínez-Trinidad, J.F., Carrasco Ochoa, J.A., Kittler, J. (eds.) CIARP 2006. LNCS, vol. 4225, pp. 529–538. Springer, Heidelberg (2006)
11. Aizenberg, I., Moraga, C.: Multilayer feedforward neural network based on multivalued neurons (MLMVN) and a backpropagation learning algorithm. Soft Computing 11, 169–183 (2007)

Modified Dendrite Morphological Neural Network Applied to 3D Object Recognition

Humberto Sossa and Elizabeth Guevara

Instituto Politécnico Nacional - CIC, Av. Juan de Dios Batiz, S/N, Col. Nva. Industrial Vallejo, Mexico, D. F. 07738, Mexico

Abstract. In this paper a modified dendrite morphological neural network (DMNN) is applied for recognition and classification of 3D objects. For feature extraction, the first two Hu's moment invariants are calculated based on 2D binary images, as well as the mean and the standard deviation obtained on 2D grayscale images. These four features were fed into a DMNN for classification of 3D objects. For testing, COIL-20 image database and a generated dataset were used. A comparative analysis of the proposed method with MLP and SVM is presented and the results reveal the advantages of the modified DMNN. An important characteristic of the proposed recognition method is that because of the simplicity of calculation of the extracted features and the DMNN, this method can be used in real applications.

Keywords: Dendrite morphological neural network, efficient training, 3D object recognition, classification.

1 Introduction

Recognizing an object from an image is an important task in computer vision due to it having a variety of applications in many areas of artificial intelligence including, for example, content-based image retrieval, industrial automation or object identification for robots [1].

In recognition systems, objects are represented in a suitable way for the processing; such representations (features) can be considered as patterns, so the correct classification of these patterns is an essential part of these systems.

Besides the pattern recognition approach that uses low-level appearance information for recognizing an object, there are feature-based geometric approaches that construct a model for the object to be recognized and match the model against the image [2]. In this paper, we describe a method to recognize 3D objects from 2D images through a pattern recognition approach using a modified DMNN for classification [3].

The proposed method of object recognition system has two phases: training and testing phase. During the training phase, the images are given as input to the system, the image is preprocessed and the feature vector is generated. The feature vector is stored with the image label and the DMNN is trained. During testing phase, the test image is given to the system; the features are extracted

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 314–324, 2013.

from the preprocessed image and then the classifier is employed to recognize the object.

The system was evaluated on the COIL-20 dataset [4] and on a set of images captured without controlled lighting condition. The system was trained using a small number of images of each object class, and as we show experimentally, it is then able to identify objects under other non trained transformations.

The rest of the paper is organized as follows. Section 2 describes the feature extraction process. Section 3 explains the modified DMNN. Section 4 presents the proposed object recognition method. Section 5 is focused to present the experimental results where the classification method used in the recognition process is tested and compared with other classifiers. Finally, Section 6 is oriented to provide the conclusions and directions for further research.

2 Feature Extraction Process

Feature extraction is a process for converting the input data into a set of features that extract the relevant information from the data in order to perform a task, in this case, recognize an object. In this paper, we use Hu's moment invariants and the mean and standard deviation of the distribution of the pixels to represent the object.

2.1 Moment Invariants

The geometric moment invariant was introduced by Hu based on the theory of algebraic invariants [5]. Image or shape feature invariants remain unchanged if the image undergoes any combination of the following transformations: translation, rotation and scaling. Therefore, the moment invariants can be used to recognize the object even if the object has changed in certain transformations.

The 2D moment of order $(p+q)$ of a digital image $f(x,y)$ of size $M \times N$ is defined as [6]

$$m_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} x^p y^q f(x,y) \tag{1}$$

where $p, q = 0, 1, 2, ..$ are integers. The corresponding central moment of order $(p+q)$ is defined as

$$\mu_{pq} = \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} (x-\bar{x})^p (y-\bar{y})^q f(x,y) \tag{2}$$

for $p, q = 0, 1, 2, ..$ where $\bar{x} = \frac{m_{10}}{m_{00}}$ and $\bar{y} = \frac{m_{01}}{m_{00}}$.

The normalized central moments, denoted η_{pq}, are defined as

$$\eta_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}} \tag{3}$$

where $\gamma = \frac{p+q}{2} + 1$ for $p + q = 2, 3, ...$

A set of seven invariant moments can be derived from the second and third moments

$$\phi_1 = \eta_{20} + \eta_{02} \tag{4}$$

$$\phi_2 = (\eta_{20} - \eta_{02})^2 + 4\eta_{11}^2 \tag{5}$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - 3\eta_{03})^2 \tag{6}$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2 \tag{7}$$

$$\begin{aligned}\phi_5 = (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + \\ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]\end{aligned} \tag{8}$$

$$\phi_6 = (\eta_{20} - \eta_{02})\left[(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right] + 4\eta_{11}(\eta_{30} + \eta_{12})(\eta_{21} + \eta_{03}) \tag{9}$$

$$\begin{aligned}\phi_7 = (3\eta_{21} - \eta_{30})(\eta_{30} + \eta_{12})\left[(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2\right] + \\ (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})\left[3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2\right]\end{aligned} \tag{10}$$

In this paper we only use the first and second invariant moments to represent the object.

3 Dendrite Morphological Neural Networks

The dendrite morphological neural networks (DMNN) were first described by Ritter and colleagues in [7] and [8]. DMNN emerge as an improvement of classical morphological neural networks (MNN), originally introduced by Davidson in [9] and then re-discussed by Ritter and Sussner in [10]. Algorithms and applications of MNN can be found in [11], [12], [13], [14], [15], [16], [17], [18] and [19]. Morphological perceptrons with competitive learning, a variation of standard morphological perceptrons are discussed in [20]. Processing at the level of dendrites and not only at the level of the cell body allows neurons to power their processing capacities [21]. This fact is taken into account by Ritter and colleagues in the DMNN proposal.

A key issue in the design of a DMNN is its training; this is in the selection of the number of dendrites and the values of synaptic weights for each dendrite. Diverse algorithms to automatically train a DMNN can be found in [7], [8], [22], [23], [24], [25] and [26].

A novel algorithm for the automatic training of a DMNN was proposed in [3] and it is applied in this work for object recognition.

3.1 Basics on Dendrite Morphological Neural Networks

Morphological neural networks are closely related to Lattice Computing [27], which can be defined as the collection of Computational Intelligence tools and techniques that either make use of lattice operators *inf* and *sup* for the construction of the computational algorithms and replace the multiplication operator of the algebra of the real numbers $(\mathbb{R}, +, \cdot)$ by the addition. Therefore, morphological neural networks use lattice operations $\vee$ (maximum), or $\wedge$ (minimum), and $+$ from the semirings $(\mathbb{R}_{-\infty}, \vee, +)$ or $(\mathbb{R}_{\infty}, \wedge, +)$ where $\mathbb{R}_{-\infty} = \mathbb{R} \cup \{-\infty\}$ and $\mathbb{R}_{\infty} = \mathbb{R} \cup \{\infty\}$. The computation at a neuron in a MNN for input $\mathbf{x} = (x_1, x_2, \ldots, x_n)$ is given by

$$\tau_j(\mathbf{x}) = a_j \bigvee_{i=1}^{n} b_{ij}(x_i + w_{ij}) \tag{11}$$

or

$$\tau_j(\mathbf{x}) = a_j \bigwedge_{i=1}^{n} b_{ij}(x_i + w_{ij}) \tag{12}$$

where $b_{ij} = \pm 1$ denotes if the ith neuron causes excitation or inhibition on the jth neuron, $a_j = \pm 1$ denotes the output response (excitation or inhibition) of the jth neuron to the neurons whose axons contact the jth neuron and w_{ij} denotes the synaptic strength between the ith neuron and the jth neuron. Parameters b_{ij} and a_j take $+1$ or -1 value if the ith input neuron causes excitation or inhibition to the jth neuron.

The computation performed by the kth dendrite can be expressed by the formula:

$$D_k(\mathbf{x}) = a_k \bigwedge_{i \in I} \bigwedge_{l \in L} (-1)^{1-l} \left(x_i + w_{ik}^l\right) \tag{13}$$

where $\mathbf{x} = (x_1, x_2, \ldots, x_n) \in \mathbb{R}^n$ corresponds to the input neurons, $I \subseteq \{1, \ldots, n\}$ denotes to the set of all input neurons N_i with terminal fibers that synapse on the kth dendrite of a morphological neuron N, $L \subseteq \{0, 1\}$ corresponds to the set of terminal fibers of the ith neuron that synapse on the kth dendrite of N, and $a_k \in \{-1, 1\}$ denotes the excitatory or inhibitory response of the kth dendrite.

Clearly, $I \neq 0$ y $L \neq 0$ since there is at least one axonal fiber coming from at least one of the input neurons with synapse dendrite k. The activation function used in a MNN is the hard limiter function that assigns 1 if the input is greater or equal to 0 and assigns 0 if the input is lesser than 0. A more detailed explanation can be found in [7] and [25].

Figure 1 shows a dendrite morphological neural network with an input layer that separates two classes: C^1 and C^2. The neurons of the input layer are connected to the next layer via the dendrites. The black and white circles denote excitatory and inhibitory connection respectively. The geometrical interpretation of the computation performed by a dendrite is that every single dendrite defines a hyperbox which can be defined by a single dendrite via its weight values w_{ij} as the example shows.

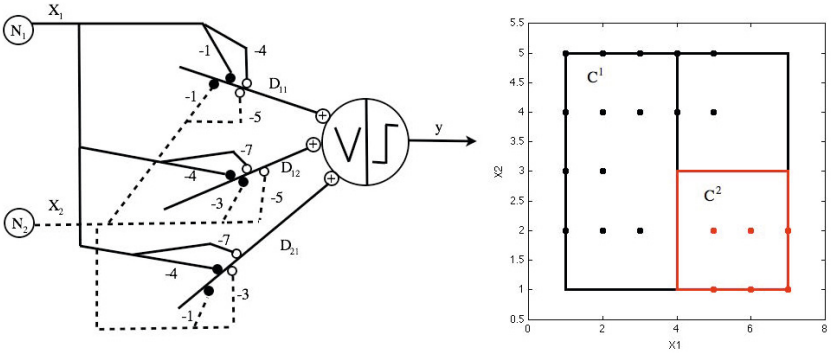


Fig. 1. Morphological neural network for solving the two classes separation problem that appears on the right side of the figure. Points of C^1 are shown as black solid dots and points of C^2 are shown as red solid dots. Points of class C^1 are enclosed by the two black boxes and C^2 by the red box generated by the dendrites.

3.2 The Training Algorithm

The proposed training algorithm for a DMNN in [3] is summarized below. Given p classes of patterns, C^k, $k = 1, 2, ..., p$, each with n attributes, the algorithm applies the following steps:

Step 1) Select the patterns of all the classes and open a hyper-cube HC^n (with n the number of attributes) with a size such that all the elements of the classes remain inside HC^n. The hyper-cube can be one whose coordinates match the patterns of class boundaries; it can be called the minimum hyper-cube *MHC*. For having better tolerance to noise at the time of classification, add a margin M on each side of the *MHC*. This margin is a number greater or equal to zero and is estimated as a function of the size T of the *MHC*. If $M = 0.1T$ then the new hyper-cube will extend that ratio to the four sides of the *MHC*.

Step 2) Divide the global hyper-cube into 2^n smaller hyper-cubes. Verify if each generated hyper-cube encloses patterns from only one class. If this is the case, label the hyper-cube with the name of the corresponding class, stop the learning process and proceed to step 4.

Step 3) If at least one of the generated hyper-cubes (HC^n) has patterns of more than one class, then divide HC^n into 2^n smaller hyper-cubes. Iteratively repeat the verification process onto each smaller hyper-cube until the stopping criterion is satisfied.

Step 4) Based on the coordinates on each axis, calculate the weights for each hyper-cube that encloses patterns belonging to C^k. By taking into account only those hyper-cubes that enclose items C^k and at this moment the DMNN is designed.

To improve the classification results, an evolutive algorithm was used to determine the optimal value of M and the hard limiter activation function of the original DMNN was changed by the maximum function.

4 Object Recognition System Overview

The flowchart of the framework for object recognition is shown in Fig. 2. For recognition purpose, a modified DMNN is used as classifier supported by features extracted from the given images. Hu's moment invariants and the mean and standard deviation of the distribution of the pixels are the features that represent the object. The original images are in grayscale, therefore the preprocessing phase converts the gray image into binary image necessary to calculate the moment invariants. The binarization is performed using the Otsu algorithm. The mean and the standard deviation are obtained directly from the grayscale images.

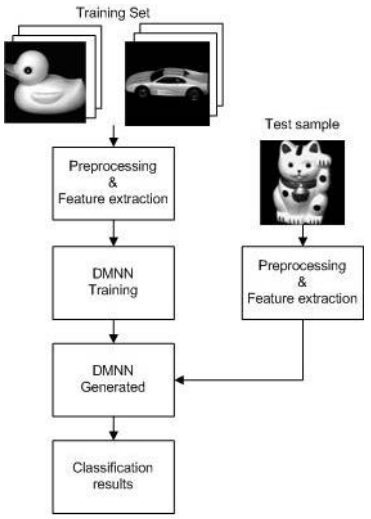


Fig. 2. Object recognition process

5 Experimental Results and Discussion

To test the performance of the object recognition approach, we use a set of images captured without controlled lighting condition and the COIL-20 dataset. The experimental dataset consists of 5 objects with 10 images each one. The images were captured using a RGB camera with a resolution of 320x240 but the region of interest of the image was of 250x200 pixels. Figure 3 a) shows the objects of the dataset.

The COIL-20 dataset [4] consists of grayscale images of 20 different objects with black background, each one is rotated with 5 degree angle interval in vertical axis. Hence for every object there are 72 images. The size of the images is of 128x128 pixels. A part of the gallery used is shown in Fig. 3 b).

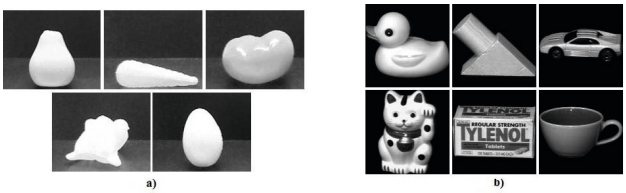


Fig. 3. a) Set of 5 objects, b) Sample objects from COIL-20 dataset

The proposed object recognition process (Fig. 2) was implemented for training and testing online using a platform developed in C#. All the algorithms were implemented on a desktop computer with Intel i7 2.2 GHz processor, with 8GB in RAM. Figure 4 shows an image of the graphic interface of the application. The 5 objects dataset was generated with this interface.

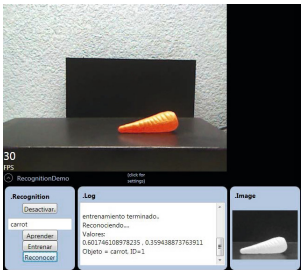


Fig. 4. Graphic interface for online training and testing for object recognition

Figure 5 presents some taken images of the carrot; on these images it is obvious that the carrot has shadows because these images were not taken under strict illumination controlled condition; in spite of this, results were satisfactory.

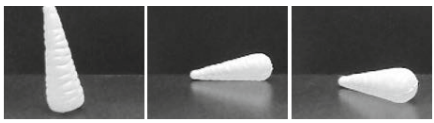


Fig. 5. Sample images of the carrot

For object representation, we use four features: the first and second Hu's moments obtained from binary images and the mean and standard deviation of

the distribution of the pixels calculated from gray images. The binary images were generated in the preprocessing stage by the Otsu algorithm.

The system was trained with 3 random images of each object and the rest of the images (7) were used for testing. To display some results MATLAB 7.11 was used; Figure 6 presents a graphic of attributes from the 5 objects, it only displays three features (first moment, second moment and mean), in this way, the patterns are in the 3D space and the results can be viewed graphically.

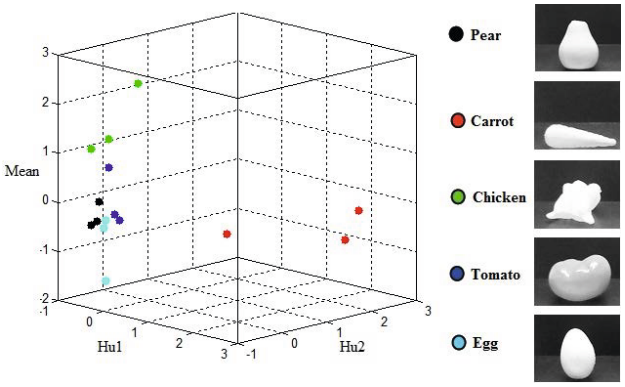


Fig. 6. Three features from the 5 objects dataset

Figure 7 a) presents the result of the training and Figure 7 b) shows the classification results. In these figures, solid dots represent the training points for the classes. Empty diamonds represent the test samples and asterisks represent the test samples classified in each class. The algorithm achieved a recognition rate of 88.57% with M=0.2861 and 10 dendrites. These results were compared with a Multilayer Perceptron (MLP) and Support Vector Machine (SVM).

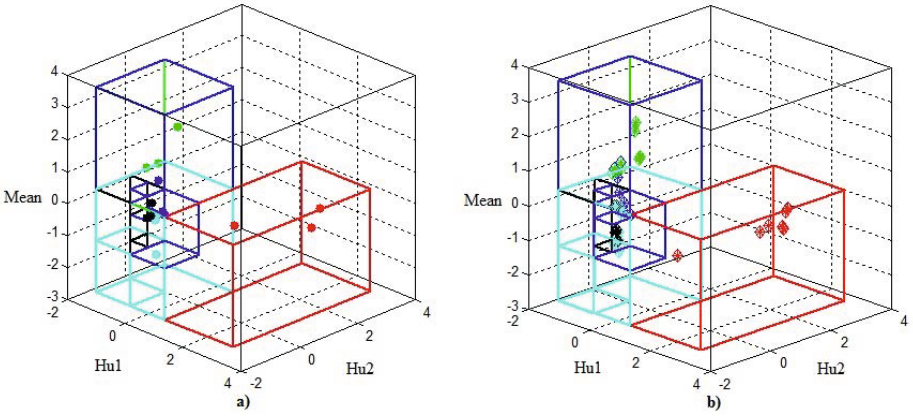


Fig. 7. a) Training result for 3 features, b) Classification results

The MLP and SVM were applied using the software Weka 3-6-8. For the MLP with one hidden layer, the training parameters were established as: learning rate=0.3 and momentum=0.2, the activation function used was the sigmoid. For the SVM, a Polynomial Kernel of degree 2 was used. Table 1 presents the results of the comparison, as it can be seen, the modified DMNN improves the results of the MLP and the SVM, so the performance of the proposed algorithm [3] is satisfactory for object recognition.

Table 1. Comparison table of the MLP, SVM and modified DMNN for the 5 objects dataset

MLP		**SVM**	**DMNN**		
# Neurons	% of Recognition	% of Recognition	# Dendrites	M	% of Recognition
9	80	85.71	10	0.2861	88.57

For the COIL-20 dataset, we used the images at 0, 45, 90, 135, 180, 225, 270 and 315 degrees for training on each of the 20 classes. Testing was done on the remaining images. Table 2 presents the recognition results for the COIL-20 dataset obtained by each one of the algorithms. As can be seen, also for this problem, the error obtained with the modified DMNN improves the results of the MLP and minimally to the SVM.

Table 2. Comparison table of the MLP, SVM and modified DMNN for the COIL-20 dataset

MLP		**SVM**	**DMNN**		
# Neurons	% of Recognition	% of Recognition	# Dendrites	M	% of Recognition
32	79.76	82.10	75	0.2876	82.27

To have an estimated error generalization, 10 experiments were realized with training and testing samples randomly selected for both datasets. Table 3 shows on the left, the average of the percentage of recognition for the 5 objects dataset, and on the right, the average for the COIL-20 dataset is presented. For the 5 objects dataset, 3 images were randomly selected for training and the rest were used for testing; and for the COIL-20 dataset, 8 images of each object were randomly selected (160 samples) for training and the others 1280 samples were used for testing. For the COIL-20 dataset, the results show that is convenient to include images of specific views of the object in the training dataset to improve the recognition percentage.

These results reveal that the performance of the modified DMNN for object recognition improves the results obtained with the MLP and SVM; however, it is necessary to find better features that describe to objects for improving these recognition rates.

Table 3. Average percentage of recognition of the MLP, SVM and modified DMNN for both datasets

	MLP	SVM	DMNN
5 objects dataset			
% of Recognition	80.57	83.42	86.85

	MLP	SVM	DMNN
COIL-20 dataset			
% of Recognition	75.77	79.16	79.36

6 Conclusions

In this paper a modified DMNN was applied for 3D object recognition using 2D images. By using 2D moments and characteristics of the pixel distribution, the proposed method does not require complex features calculation, thus reduces processing time in feature extraction stage. Comparisons of the modified DMNN with the MLP and SVM demonstrated the advantages of the DMNN. The characteristics of the proposed method allowed its implementation in a real application with good results.

Future work will be focused on the improvement of the results using features more adequate for 3D images as color and depth characteristics. Furthermore, it is necessary the implementation of DMNN training algorithm on a parallel architecture for evaluating the method in larger databases with more describing features.

Acknowledgments. H. Sossa would like to thank SIP-IPN and CONACYT under grants 20121311, 20131182 and 155014 for the economical support to carry out this research. E. Guevara thanks CONACYT for the scholarship granted to pursuit her doctoral studies.

References

1. Wöhler, C.: 3D Computer Vision: Efficient Methods and Applications. Springer (2012)
2. Grauman, K., Leibe, B.: Visual Object Recognition. Morgan & Claypool (2011)
3. Sossa, H., Guevara, E.: Efficient training for dendrite morphological neural networks. Submitted to Neurocomputing - Elsevier Journal
4. Nene, D., Nayar, S., Murase, H.: Columbia object image library: COIL (1996)
5. Hu, M.K.: Visual pattern recognition by moment invariants. IRE Transactions on Information Theory 8, 179–187 (1962)
6. González, R., Woods, R.: Digital Image Processing. Pearson (2007)
7. Ritter, G.X., Iancu, L., Urcid, G.: Morphological perceptrons with dendritic structure. In: 12th IEEE International Conference in Fuzzy Systems (FUZZ 2003), vol. 2, pp. 1296–1301 (2003)
8. Ritter, G.X., Urcid, G.: Lattice algebra approach to single-neuron computation. IEEE Transactions on Neural Networks 14(2), 282–295 (2003)
9. Davidson, J.L., Hummer, F.: Morphology neural networks: An introduction with applications. Circuits Systems Signal Process 12(2), 177–210 (1993)
10. Ritter, G.X., Sussner, P.: An introduction to morphological neural networks. In: Proceedings of the 13th International Conference on Pattern Recognition, vol. 4, pp. 709–717 (1996)

11. Sussner, P.: Morphological perceptron learning. In: IEEE ISIC/CIRA/ISAS Joint Conference, pp. 477–482 (1998)
12. Piñeiro Colón, R.C., Ortiz, J.L.: Evolutionary Training of Morphological Neural Networks. PhD thesis, Electrical and Computer Engineering Department. University of Puerto Rico, Mayagüez Campus
13. Lima, C.A.M., Coelho, A.L.V., Silva, M.E.S., Gudwin, R.R., Von Zuben, F.J.: Hybrid training of morphological neural networks: A comparative study. Technical report, DCA-FEEC-Unicamp
14. De Aráujo, R., Madeiro, F., De Sousa, R.P., Pessoa, L.F.C.: Modular morphological neural network training via adaptive genetic algorithm for designing translation invariant operators. In: IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2006), vol. 2, pp. 873–876 (2006)
15. Raducanu, B., Graña, M., Sussner, P.: Morphological neural networks for vision based self-localization. In: IEEE International Conference on Robotics and Automation, pp. 2059–2064 (2001)
16. Nong, Y., Hao, W., Changyong, W., Fanming, L., Lide, W.: Morphological neural networks for automatic target detection by simulated annealing learning algorithm. Science in China (Series F) 46, 262–278 (2003)
17. Villaverde, I., Graña, M., D'Anjou, A.: Morphological neural networks for localization and mapping. In: IEEE International Conference on Computational Intelligence for Measurement Systems and Applications (CIMSA 2006), pp. 9–14 (2006)
18. Roberson, C., Dankel II, D.D.: A morphological neural network approach to information retrieval. In: Twentieth International Florida Artificial Intelligence Research Society Conference, pp. 184–185 (2007)
19. Cheng-Tian, S., Ke-Yong, W.: Image target detection using morphological neural network. In: International Conference on Computational Intelligence and Security, pp. 234–236 (2009)
20. Sussner, P., Esmi, E.L.: Morphological perceptrons with competitive learning: Lattice-theoretical framework and constructive learning algorithm. Information Sciences 181, 1929–1950 (2011)
21. Segev, I.: Dendritic processing. In: The Handbook of Brain Theory and Neural Networks, pp. 282–289 (1998)
22. Barrón, R., Sossa, H., Cortés, H.: Morphological neural networks with dendrite computation: A geometrical approach. In: Sanfeliu, A., Ruiz-Shulcloper, J. (eds.) CIARP 2003. LNCS, vol. 2905, pp. 588–595. Springer, Heidelberg (2003)
23. Valle, M.E.R., Pimenta, M.A.: Perceptrón morfológico de camada única. Technical report, Department of Mathematics of the State University of Londrina, Brazil (2005)
24. Ritter, G.X., Schmalz, M.S.: Learning in lattice neural networks that employ dendritic computing. In: IEEE International Conference on Fuzzy Systems, Vancouver, BC, Canada, pp. 7–13 (2006)
25. Ritter, G.X., Urcid, G.: Learning in lattice neural networks that employ dendritic computing. Computational Intelligence Based on Lattice Theory 67, 25–44 (2007)
26. Chyzhyk, D., Graña, M.: Optimal hyperbox shrinking in dendritic computing applied to Alzheimer's disease detection in MRI. In: Corchado, E., Snášel, V., Sedano, J., Hassanien, A.E., Calvo, J.L., Ślęzak, D. (eds.) SOCO 2011. AISC, vol. 87, pp. 543–550. Springer, Heidelberg (2011)
27. Graña, M.: Special issue on: Lattice computing and natural computing. Neurocomputing 72(10-12), 2065–2066 (2009)

Hybrid Associative Memories for Imbalanced Data Classification: An Experimental Study

L. Cleofas-Sánchez[1], V. García[2], R. Martín-Félez[2], R.M. Valdovinos[3], J.S. Sánchez[2], and O. Camacho-Nieto[4]

[1] Centro de Investigación en Computación, Instituto Politécnico Nacional, Av. Juan de Dios Bátiz s/n, Col. Nueva Industrial Vallejo, 07738 México D.F., México
[2] Institute of New Imaging Technologies, Department of Computer Languages and Systems, Universitat Jaume I, Av. Vicent Sos Baynat s/n, 12071 Castellón de la Plana, Spain
[3] Centro Universitario Valle de Chalco, Universidad Autónoma del Estado de México, Hermenegildo Galena 3, 56615 Valle de Chalco, México
[4] Centro de Innovación y Desarrollo Tecnológico en Cómputo, Instituto Politécnico Nacional, Av. Juan de Dios Bátiz s/n, Col. Nueva Industrial Vallejo, 07700 México D.F., México

Abstract. Hybrid associative memories are based on the combination of two well-known associative networks, the lernmatrix and the linear associator, with the aim of taking advantage of their merits and overcoming their limitations. While these models have extensively been applied to information retrieval problems, they have not been properly studied in the framework of classification and even less with imbalanced data. Accordingly, this work intends to give a comprehensive response to some issues regarding imbalanced data classification: (i) Are the hybrid associative models suitable for dealing with this sort of data? and, (ii) Does the degree of imbalance affect the performance of these neural classifiers? Experiments on real-world data sets demonstrate that independently of the imbalance ratio, the hybrid associative memories perform poorly in terms of area under the ROC curve, but the hybrid associative classifier with translation appears to be the best solution when assessing the true positive rate.

Keywords: Class Imbalance, Associative Memory, Neural Network.

1 Introduction

An associative memory [1] is a type of neural network that allows to recall the previously stored training example $\mathbf{x}^i$ that most closely resembles the one presented to the network. This connectionist model has demonstrated to be very effective for information storage and retrieval [2–4], but it has not been much studied in the framework of classification. Among the simplest and first studied associative memory models are the lernmatrix [5] and the linear associator [6,7], which are considered as hetero-associative memories capable of producing exact recall. Both these models can also work as classifiers, but they present some drawbacks that make difficult their application to many real-life problems: the lernmatrix needs to be provided with binary input vectors $\mathbf{x}^i \in \{0,1\}$, whereas the linear associator requires the input vectors to be orthonormal and linearly independent.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 325–334, 2013.

In order to benefit from the advantages of these associative memories and overcome their shortcomings, several extensions have been developed. These include the hybrid associative classifier (HAC) and the hybrid associative classifier with translation (HACT) [8], which combine the procedure used by the linear associator in the learning phase with the recall stage of the lernmatrix. While these two classification models have been used with some success in a number of applications, there still exist open questions regarding their limitations that deserve a more thorough investigation. For example, the present paper addresses the issue of imbalanced data classification [9], which appears as a much more challenging task for this type of associative memories.

Many complex pattern recognition and data mining problems are characterized by imbalanced data, where at least one class is heavily under-represented as compared to others. Following the common practice in the area [10, 11], we will here consider only binary classification problems where the examples from the majority class are often referred to as the negative examples and those from the minority class as the positive examples, since these usually represent the concept of most interest.

The importance of the class imbalance problem comes from the fact that in general, it hinders the performance of most standard learning algorithms because they are often biased towards the majority class and have a poor performance on the minority class. Besides the classifiers are commonly built with the aim of reducing the overall error, what may lead to erroneous conclusions; for example, an algorithm that achieves an accuracy of 99% will be worthless if it fails on classifying all positive examples.

Many classifiers have been investigated in the context of class imbalance, ranging from the nearest neighbor rule and decision trees to support vector machines and various topologies of neural networks [11–15]. However, to the best of our knowledge, the use of associative memory models has not received adequate attention from researchers on this topic. In fact, we have found only a recent work [16] that analyzes the performance of the HACT approach after under-sampling the imbalanced data set, but it presents several limitations such as the reduced number of databases used in the experiments, the lack of comparisons with other state-of-the-art classifiers and especially, the fact that it does not take care of the imbalance ratio (i.e. the ratio of the majority to the minority instances) and its effect on the HACT performance.

The purpose of this paper is to gain insight into the behavior of the HAC and HACT associative models when these are used for the classification of imbalanced data, pursuing to fully understand how the class imbalance affects the performance of these classifiers. To this end, we provide a large pool of experiments on 58 real-world benchmarking data sets that have different degrees of imbalance, comparing those hybrid associative memories with other well-known artificial neural networks: a Bayesian network (BNet), a multilayer perceptron (MLP) and a radial basis function (RBF). We conducted our experiments by evaluating three performance metrics: the area under the ROC curve, the true positive rate and the true negative rate.

2 Two Hybrid Associative Memories

In this section we provide a brief introduction to the associative memory models that will be further experimented with, covering only the general concepts and notation

needed to understand their foundations. For a complete description of associative memories, the reader may review any of the many books on this subject (e.g. [17, 18]).

In general, an associative memory can be defined as a mapping matrix $\mathbf{M}$ so that an input vector $\mathbf{x}^i \in \mathbb{R}^n$ (with n components) will be transformed into an output vector $\mathbf{y}^i \in \mathbb{R}^m$ (with m components), that is

$$\mathbf{y}^i = \mathbf{M}\mathbf{x}^i \qquad i = 1, \ldots, p \tag{1}$$

where p denotes the number of input vectors.

The stored samples will be represented in the form of pairs of associations $(\mathbf{x}^i, \mathbf{y}^i)$ between the input and output vectors, $\mathbf{x}^i$ and $\mathbf{y}^i$, and are often called fundamental pattern. The set of p pairs (fundamental patterns) constitutes the fundamental set of associations.

The matrix $\mathbf{M}$ has to be determined through an iterative procedure in the learning phase. Afterwards, during the recall or recovery phase, an unknown pattern $\mathbf{x}^0$ will be applied to the input of the matrix in order to produce the vector $\mathbf{y}^0$, which is expected to be a good approximation of the true output $\mathbf{y}$.

Hybrid Associative Classifier (HAC). As previously pointed out, the HAC model [8] arises from the combination of the lernmatrix and the linear associator with the aim of overcoming the practical drawbacks of these associative neural networks. Apart from these obvious advantages, it is worth remarking that the HAC model presents some other interesting properties such as simplicity, requirements of low computational cost and the ability to support real-valued input vectors [8].

During the learning phase, the HAC memory imitates the process of the linear associator: each sample that belongs to class k is represented by a vector with zeros in all components except the k'th element that equals 1. In this way, the outer product of vectors $\mathbf{x}^i$ and $\mathbf{y}^i$ gives the corresponding associations between them. Then the matrix $\mathbf{M}$ of size $n \times m$ will be obtained as the sum of all p outer products as

$$\mathbf{M} = \sum_{i=1}^{p} (\mathbf{y}^i)(\mathbf{x}^i)^{\mathbf{T}} \tag{2}$$

After computing the mapping matrix $\mathbf{M}$, the recovery of a given input sample will be performed following the process of the lernmatrix model in order to estimate its class label.

It has to be pointed out, however, that a practical drawback of the HAC model comes from the possible large differences in the magnitude of the input vectors because in such a case, the vectors with a lower magnitude will be assigned to the class of the vectors with a larger magnitude.

Hybrid Associative Classifier with Translation (HACT). This is a modification of the HAC model that tries to face several of its limitations. More specifically, if the input samples are clustered in the same quadrant, the performance of the HAC memory will

be affected negatively. Thus the HACT approach [8] starts with a translation of the coordinate axes whose origin is taken to lie in the mean vector of all the input vectors as computed by

$$\overline{\mathbf{x}} = \frac{1}{p}\sum_{i=1}^{p}\mathbf{x}^i \tag{3}$$

In this way, the new coordinate axes are parallel to the original coordinate axes, but eliminates the clustering of samples in a unique quadrant. Then the input and test vectors in the new coordinate system will be obtained as $\mathbf{x}^i = \mathbf{x}^i - \overline{\mathbf{x}}$. After the corresponding translation of axes, the learning and recovery phases will be the same as those described for the HAC model.

3 Experimental Set-Up

As already discussed, the aim of this work and the experiments conducted here is to investigate whether two models of hybrid associative memories, which are based on the lernmatrix and the linear associator, are suitable or not for imbalanced data classification, and to what extent the degree of imbalance may affect their performance.

Table 1. Description of the data sets used in the experiments

Data sets	Features	Samples	IR	Data sets	Features	Samples	IR
Glass1	9	214	1.82	Ecoli-0-3-4-6_vs_5	7	205	9.25
Pima	8	768	1.87	Ecoli-0-3-4-7_vs_5-6	7	257	9.28
Iris0	4	150	2.00	Yeast-0-5-6-7-9_vs_4	8	528	9.35
Glass0	9	214	2.06	Vowel0	13	988	9.98
Yeast1	8	1484	2.46	Ecoli-0-6-7_vs_5	6	220	10.00
Haberman	3	306	2.78	Glass-0-1-6_vs_2	9	192	10.29
Vehicle3	18	846	3.00	Ecoli-0-1-4-7_vs_2-3-5-6	7	336	10.59
Glass-0-1-2-3_vs_4-5-6	9	214	3.20	Led-0-2-4-5-6-7-8-9_vs_1	7	443	10.97
Vehicle0	18	846	3.25	Ecoli-0-1_vs_5	6	240	11.00
Ecoli1	7	336	3.36	Glass-0-6_vs_5	9	108	11.00
New-thyroid2	5	215	5.14	Glass-0-1-4-6_vs_2	9	205	11.06
Ecoli2	7	336	5.46	Glass2	9	214	11.59
Segment0	19	2308	6.02	Ecoli-0-1-4-7_vs_5-6	6	332	12.28
Glass6	9	214	6.38	Cleveland-0_vs_4	13	177	12.62
Yeast3	8	1484	8.10	Ecoli-0-1-4-6_vs_5	6	280	13.00
Ecoli3	7	336	8.60	Shuttle-0_vs_4	9	1829	13.87
Page-blocks0	10	5472	8.79	Yeast-1_vs_7	7	459	14.30
Ecoli-0-3-4_vs_5	7	200	9.00	Glass4	9	214	15.47
Yeast-2_vs_4	8	514	9.08	Ecoli4	7	336	15.80
Ecoli-0-6-7_vs_3-5	7	222	9.09	Page-blocks-1-3_vs_4	10	472	15.86
Ecoli-0-2-3-4_vs_5	7	202	9.10	Glass-0-1-6_vs_5	9	184	19.44
Glass-0-1-5_vs_2	9	172	9.12	Yeast-1-4-5-8_vs_7	8	693	22.10
Yeast-0-3-5-9_vs_7-8	8	506	9.12	Glass5	9	214	22.78
Yeast-0-2-5-6_vs_3-7-8-9	8	1004	9.14	Yeast-2_vs_8	8	482	23.10
Yeast-0-2-5-7-9_vs_3-6-8	8	1004	9.14	Yeast4	8	1484	28.10
Ecoli-0-4-6_vs_5	6	203	9.15	Yeast-1-2-8-9_vs_7	8	947	30.57
Ecoli-0-1_vs_2-3-5	7	244	9.17	Yeast5	8	1484	32.73
Ecoli-0-2-6-7_vs_3-5	7	224	9.18	Ecoli-0-1-3-7_vs_2-6	7	281	39.14
Glass-0-4_vs_5	9	92	9.22	Yeast6	8	1484	41.40

The empirical analysis has been performed over a total of 58 benchmarking data sets taken from the KEEL Data Set Repository (`http://www.keel.es/dataset.php`) [19]. Note that all the original multi-class databases have firstly been transformed into two-class problems. Table 1 summarizes the main characteristics of the data sets, including the imbalance ratio (IR), i.e. the number of negative examples divided by the number of positive examples. As can be seen, the databases chosen for the experiments go from a low imbalance of 1.82 in Glass1 to a high/moderate imbalance of 41.40 in the case of Yeast6.

In order to gain sufficient insight into the behavior of the associative memory models, three other neural networks (BNet, MLP, RBF) have been used as baselines for comparison purposes. These were taken from the Weka toolkit [20] with their default parameter values. For the experiments here carried out, we have adopted a 5-fold cross-validation method to estimate three classification performance measures commonly used in skewed domains: the area under the ROC curve (AUC), the true positive rate (TPrate) and the true negative rate (TNrate). Each data set has been divided into five stratified blocks of size $N/5$ (where N denotes the total number of samples in the database), using four folds for training the connectionist classifiers and the remaining block as an independent test set. Therefore the results reported in tables of Section 4 correspond to those three measures averaged over the five runs.

Table 2. Confusion matrix for a two-class problem

	Predicted positive	Predicted negative
Positive class	True Positive (TP)	False Negative (FN)
Negative class	False Positive (FP)	True Negative (TN)

Given a 2×2 confusion matrix as that illustrated in Table 2, the performance measures used in the experiments can be calculated as follows: $TPrate = \frac{TP}{TP+FN}$, $TNrate = \frac{TN}{TN+FP}$, and $AUC = \frac{TPrate+TNrate}{2}$. Note that the latter corresponds to the AUC defined by a single point on the ROC curve.

4 Experimental Results

Table 3 reports the AUC values obtained by the neural network models on each database, along with the average across the whole collection of data sets. From these results, the first observation is that the HAC memory yields a 50% of AUC, which indicates that all samples of one class have been misclassified while all of the other have been correctly classified. This effect has not been found in the case of the HACT model, but its performance in terms of AUC is lower than that achieved by the three other neural networks on most databases. When paying attention of the average values, the MLP model clearly performs the best (80.70% of AUC), but the results of the Bayesian network and the RBF are not too far from that of the HACT approach.

Table 3. Experimental results using the AUC

Data set	HAC	HACT	BNet	MLP	RBF	Data set	HAC	HACT	BNet	MLP	RBF
Glass1	50.00	56.02	67.51	68.6	62.24	Ecoli-0-3-4-6_vs_5	50.00	79.12	83.11	88.65	91.96
Pima	50.00	57.58	69.01	74.69	70.30	Ecoli-0-3-4-7_vs_5-6	50.00	79.05	73.78	88.92	84.06
Iris0	50.00	95.50	100	100	100	Yeast-0-5-6-7-9_vs_4	50.00	74.94	56.91	72.79	53.36
Glass0	50.00	71.53	79.93	77.01	67.63	Vowel0	50.00	77.39	88.43	99.44	86.78
Yeast1	50.00	66.92	67.59	66.94	60.74	Ecoli-0-6-7_vs_5	50.00	79.75	82.25	86.50	87.25
Haberman	50.00	62.74	55.42	58.10	55.11	Glass-0-1-6_vs_2	50.00	63.14	50.00	47.71	48.00
Vehicle3	50.00	65.10	67.63	74.26	63.63	Ecoli-0-1-4-7_vs_2-3-5-6	50.00	76.81	80.51	87.03	79.01
Glass-0-1-2-3_vs_4-5-6	50.00	92.69	88.26	92.03	89.41	Led-0-2-4-5-6-7-8-9_vs_1	51.25	81.66	88.24	89.30	83.06
Vehicle0	50.00	74.64	81.74	94.95	84.51	Ecoli-0-1_vs_5	50.00	77.72	87.04	89.54	89.54
Ecoli1	50.00	87.36	85.01	85.83	88.35	Glass-0-6_vs_5	50.00	86.34	78.39	100	94.50
New-thyroid2	50.00	75.71	92.85	95.15	98.01	Glass-0-1-4-6_vs_2	50.00	64.62	50.00	48.67	49.74
Ecoli2	50.00	82.34	86.08	89.24	90.72	Glass2	50.00	65.49	50.00	51.03	48.97
Segment0	50.00	75.82	98.78	99.39	97.71	Ecoli-0-1-4-7_vs_5-6	50.00	79.30	51.84	84.87	83.19
Glass6	50.00	89.41	91.17	84.92	87.44	Cleveland-0_vs_4	50.00	47.92	62.63	87.22	84.90
Yeast3	50.00	78.92	85.42	85.85	87.06	Ecoli-0-1-4-6_vs_5	50.00	77.31	86.93	79.05	89.23
Ecoli3	50.00	81.96	84.01	78.34	66.82	Shuttle-0_vs_4	50.00	91.19	100	99.60	99.11
Page-blocks0	50.00	48.70	89.73	87.59	74.52	Yeast-1_vs_7	50.00	65.25	46.43	62.61	54.53
Ecoli-0-3-4_vs_5	50.00	80.00	84.44	88.60	91.66	Glass4	50.00	82.57	64.92	87.34	86.59
Yeast-2_vs_4	50.00	74.67	87.40	82.50	87.89	Ecoli4	50.00	81.51	82.34	89.21	89.05
Ecoli-0-6-7_vs_3-5	50.00	77.00	89.00	82.50	68.50	Page-blocks-1-3_vs_4	50.00	80.17	96.56	97.89	91.99
Ecoli-0-2-3-4_vs_5	50.00	80.22	86.40	89.17	89.20	Glass-0-1-6_vs_5	50.00	88.29	90.43	79.14	89.71
Glass-0-1-5_vs_2	50.00	63.63	50.00	52.48	50.24	Yeast-1-4-5-8_vs_7	50.00	59.65	50.00	51.37	50.00
Yeast-0-3-5-9_vs_7-8	50.00	69.43	59.78	64.69	61.45	Glass5	50.00	88.05	91.34	89.51	84.02
Yeast-0-2-5-6_vs_3-7-8-9	50.00	69.89	75.08	73.38	67.66	Yeast-2_vs_8	50.00	77.32	77.39	77.06	79.78
Yeast-0-2-5-7-9_vs_3-6-8	50.00	75.75	83.89	86.22	88.86	Yeast4	50.00	73.32	62.84	64.39	50.00
Ecoli-0-4-6_vs_5	50.00	78.97	89.18	88.92	86.69	Yeast-1-2-8-9_vs_7	50.00	65.03	57.96	56.46	51.67
Ecoli-0-1_vs_2-3-5	50.00	77.54	50.56	80.67	79.21	Yeast5	50.00	78.65	91.77	83.60	63.30
Ecoli-0-2-6-7_vs_3-5	50.00	77.95	80.01	81.01	81.01	Ecoli-0-1-3-7_vs_2-6	50.00	80.85	84.63	84.81	84.63
Glass-0-4_vs_5	50.00	90.81	99.41	100	94.41	Yeast6	50.00	74.89	83.30	73.85	50.00
Average							50.02	75.45	77.16	80.70	77.05

By the analysis of the behavior of these classifiers as a function of the imbalance ratio, one can guess that there is not necessarily a direct relationship between the classification performance and the degree of imbalance. For instance, the balanced accuracies for the Ecoli-0-1-3-7_vs_2-6 database, which has an imbalance ratio of 39.14, are significantly higher than those for Glass1, even though this presents a very low imbalance ratio of 1.82. In some sense, it appears that databases may also suffer from other intrinsic problems such as class overlapping, small disjuncts, feature noise and lack of representative data, which in turn may affect classification performance much more strongly than the presence of class imbalance.

In order to accomplish a better understanding of the performance of these neural network models, Tables 4 and 5 report the true positive and true negative rates, respectively. These measures allow to analyze the behavior of a classifier on each individual class, thus drawing out whether it is biased towards one class or another. This is especially important in the context of imbalanced data because the examples from the minority class, which usually correspond to the most interesting cases, are more likely to be misclassified. In addition, it is often preferable to achieve a higher true positive rate rather than a higher true negative rate and consequently, the AUC by itself is not sufficiently informative when evaluating the performance of a set of classifiers.

Table 4. Experimental results using the true positive rate

Data set	HAC	HACT	BNet	MLP	RBF	Data set	HAC	HACT	BNet	MLP	RBF
Glass1	0	80.17	47.34	59.60	50.00	Ecoli-0-3-4-6_vs_5	0	95.00	70	80	85.00
Pima	0	44.36	58.22	67.18	55.20	Ecoli-0-3-4-7_vs_5-6	0	96.00	48.00	80.00	72.00
Iris0	0	100	100	100	100	Yeast-0-5-6-7-9_vs_4	0	86.36	18.00	48.72	8.00
Glass0	0	100	80.00	70.00	42.84	Vowel0	0	97.78	78.86	98.88	75.56
Yeast1	0	76.68	46.14	43.84	27.28	Ecoli-0-6-7_vs_5	0	95.00	65.00	75.00	75.00
Haberman	0	59.26	17.52	28.20	15.98	Glass-0-1-6_vs_2	0	100	0	0	0
Vehicle3	0	60.33	63.64	58.94	41.92	Ecoli-0-1-4-7_vs_2-3-5-6	0	93.33	62.66	76.00	58.66
Glass-0-1-2-3_vs_4-5-6	0	94.00	80.18	87.74	84.36	Led-0-2-4-5-6-7-8-9_vs_1	2.50	100	78.20	81.06	67.84
Vehicle0	0	100	95.94	90.98	80.92	Ecoli-0-1_vs_5	0	100	75.00	80.00	80.00
Ecoli1	0	94.83	83.16	76.68	91.02	Glass-0-6_vs_5	0	100	70.00	100	90.00
New-thyroid2	0	91.43	85.70	91.42	97.14	Glass-0-1-4-6_vs_2	0	100	0	0	0
Ecoli2	0	96.36	77.44	82.72	87.08	Glass2	0	100	0	6.66	0
Segment0	0	100	98.20	99.10	97.90	Ecoli-0-1-4-7_vs_5-6	0	100	44.00	72.00	68.00
Glass6	0	96.67	86.66	72.00	78.66	Cleveland-0_vs_4	0	33.50	26.04	78.18	71.52
Yeast3	0	98.79	72.94	74.28	77.32	Ecoli-0-1-4-6_vs_5	0	100	75.00	60.00	80.00
Ecoli3	0	97.14	79.98	59.98	34.30	Shuttle-0_vs_4	0	99.20	100	99.20	98.40
Page-blocks0	0	19.15	85.32	76.92	50.84	Yeast-1_vs_7	0	76.67	13.34	26.64	10.00
Ecoli-0-3-4_vs_5	0	100	70.00	80.00	85.00	Glass4	0	90.00	33.32	76.68	76.68
Yeast-2_vs_4	0	90.18	76.54	66.54	78.36	Ecoli4	0	100	65.00	80.00	80.00
Ecoli-0-6-7_vs_3-5	0	88.00	80.00	67.00	41.00	Page-blocks-1-3_vs_4	0	68.67	100	96.00	86.00
Ecoli-0-2-3-4_vs_5	0	100	75.00	80.00	80.00	Glass-0-1-6_vs_5	0	100	90.00	60.00	80.00
Glass-0-1-5_vs_2	0	95.00	0	13.34	5.00	Yeast-1-4-5-8_vs_7	0	66.67	0	3.34	0
Yeast-0-3-5-9_vs_7-8	0	86.00	20.00	34.00	24.00	Glass5	0	100	90.00	80.00	70.00
Yeast-0-2-5-6_vs_3-7-8-9	0	77.68	54.36	49.42	37.32	Yeast-2_vs_8	0	70.00	55.00	55.00	60.00
Yeast-0-2-5-7-9_vs_3-6-8	0	89.95	70.68	73.78	79.94	Yeast4	0	90.18	29.28	29.46	0
Ecoli-0-4-6_vs_5	0	95.00	80.00	80.00	75.00	Yeast-1-2-8-9_vs_7	0	80.00	16.68	13.34	3.34
Ecoli-0-1_vs_2-3-5	0	96.00	10.00	65.00	65.00	Yeast5	0	100	86.4	68.08	26.92
Ecoli-0-2-6-7_vs_3-5	0	90.00	63.00	64.00	64.00	Ecoli-0-1-3-7_vs_2-6	0	100	70.00	70.00	70.00
Glass-0-4_vs_5	0	100	100	100	90.00	Yeast6	0	94.29	71.42	48.58	0
Average							0.04	88.96	60.16	64.75	57.42

For instance, the results of HAC in Table 4 demonstrate that this hybrid associative model is of no value at all because it fails on the classification of all positive examples. This makes clear that the AUC of 50% reported in Table 3 is due to the awful true positive rate of this classifier and the very high rate achieved on the negative class (see Table 5). On the contrary, the true positive rate of HACT suggests that this can be a good tool for the classification of data with class imbalance because it yields a true positive rate of close to 89% in average, that is, even higher than that of the best performing algorithm (MLP) in terms of AUC.

It is also interesting to note that in general, the highest differences between HACT and MLP are found in the most strongly imbalanced data sets. Unfortunately, in these cases, the true negative rate of the HACT model is lower than that of the MLP, but we should recall that there exist numerous real-world applications in which the minority class represents the concept of most interest and therefore, it will be crucial to correctly classify the positive examples even if this might entail a certain degradation of the true negative rate.

Table 5. Experimental results using the true negative rate

Data set	HAC	HACT	BNet	MLP	RBF	Data set	HAC	HACT	BNet	MLP	RBF
Glass1	100	31.88	87.68	77.6	74.48	Ecoli-0-3-4-6_vs_5	100	63.24	96.22	97.30	98.92
Pima	100	70.80	79.80	82.20	85.40	Ecoli-0-3-4-7_vs_5-6	100	62.11	99.56	97.84	96.12
Iris0	100	91.00	100	100	100	Yeast-0-5-6-7-9_vs_4	100	63.53	95.82	96.86	98.72
Glass0	100	43.05	79.86	84.02	92.42	Vowel0	100	57.01	98.00	100	98.00
Yeast1	100	57.16	89.04	90.04	94.20	Ecoli-0-6-7_vs_5	100	64.50	99.50	98.00	99.50
Haberman	100	66.22	93.32	88.00	94.24	Glass-0-1-6_vs_2	100	26.29	100	95.42	96.00
Vehicle3	100	69.87	71.62	89.58	85.34	Ecoli-0-1-4-7_vs_2-3-5-6	100	60.28	98.36	98.06	99.36
Glass-0-1-2-3_vs_4-5-6	100	91.38	96.34	96.32	94.46	Led-0-2-4-5-6-7-8-9_vs_1	100	63.33	98.28	97.54	98.28
Vehicle0	100	49.29	67.54	98.92	88.10	Ecoli-0-1_vs_5	100	55.45	99.08	99.08	99.08
Ecoli1	100	79.88	86.86	94.98	85.68	Glass-0-6_vs_5	100	72.68	86.78	100	99.00
New-thyroid2	100	60.00	100	98.88	98.88	Glass-0-1-4-6_vs_2	100	29.25	100	97.34	99.48
Ecoli2	100	68.31	94.72	95.76	94.36	Glass2	100	30.99	100	95.40	97.94
Segment0	100	51.64	99.36	99.68	97.52	Ecoli-0-1-4-7_vs_5-6	100	58.59	59.68	97.74	98.38
Glass6	100	82.16	95.68	97.84	96.22	Cleveland-0_vs_4	100	52.50	99.22	96.26	98.28
Yeast3	100	59.05	97.90	97.42	96.80	Ecoli-0-1-4-6_vs_5	100	54.62	98.86	98.10	98.46
Ecoli3	100	66.78	88.04	96.70	99.34	Shuttle-0_vs_4	100	83.18	100	100	99.82
Page-blocks0	100	78.24	94.14	98.26	98.20	Yeast-1_vs_7	100	53.84	79.52	98.58	99.06
Ecoli-0-3-4_vs_5	100	60.00	98.88	97.20	98.32	Glass4	100	75.13	96.52	98.00	96.50
Yeast-2_vs_4	100	59.17	98.26	98.46	97.42	Ecoli4	100	63.03	99.68	98.42	98.10
Ecoli-0-6-7_vs_3-5	100	66.00	98.00	98.00	96.00	Page-blocks-1-3_vs_4	100	91.67	93.12	99.78	97.98
Ecoli-0-2-3-4_vs_5	100	60.44	97.80	98.34	98.40	Glass-0-1-6_vs_5	100	76.57	90.86	98.28	99.42
Glass-0-1-5_vs_2	100	32.26	100	91.62	95.48	Yeast-1-4-5-8_vs_7	100	52.63	100	99.40	100
Yeast-0-3-5-9_vs_7-8	100	52.85	99.56	95.38	98.90	Glass5	100	76.10	92.68	99.02	98.04
Yeast-0-2-5-6_vs_3-7-8-9	100	62.10	95.80	97.34	98.00	Yeast-2_vs_8	100	84.65	99.78	99.12	99.56
Yeast-0-2-5-7-9_vs_3-6-8	100	61.55	97.10	98.66	97.78	Yeast4	100	56.46	96.40	99.32	100
Ecoli-0-4-6_vs_5	100	62.94	98.36	97.84	98.38	Yeast-1-2-8-9_vs_7	100	50.06	99.24	99.58	100
Ecoli-0-1_vs_2-3-5	100	59.09	91.12	96.34	93.42	Yeast5	100	57.29	97.14	99.12	99.68
Ecoli-0-2-6-7_vs_3-5	100	65.90	97.02	98.02	98.02	Ecoli-0-1-3-7_vs_2-6	100	61.69	99.26	99.62	99.26
Glass-0-4_vs_5	100	81.62	98.82	100	98.82	Yeast6	100	55.49	95.18	99.12	100
Average							100	61.94	94.16	96.65	96.68

5 Conclusions and Further Research

This paper pursues to investigate the suitability of associative memories for the classification of data in imbalanced domains. In particular, the present work has concentrated on two hybrid models, the hybrid associative classifier (HAC) and the hybrid associative classifier with translation (HACT), which come from the combination of the learning phase of the linear associator with the recall phase of the lernmatrix.

In contrast to the lernmatrix and the linear associator, two of the main characteristics of HAC and HACT refer to the potential of using real-valued input vectors that do not require to be orthonormal and linearly independent. These appealing properties allow the application of the hybrid associative models to a huge number of real-life problems. However, they have not been thoroughly studied in the context of imbalanced data classification and therefore, it is not possible to fully assert their suitability in domains where it is common to find such a complexity in data (i.e., credit risk evaluation, fraud detection in mobile telephone communications and prediction of rare diseases).

The experiments carried out over a collection of 58 real-world databases with the two hybrid associative models and three classical neural networks (Bayesian network, MLP and RBF) have demonstrated both the non-suitability of the HAC approach and

the good performance of the HACT memory. In fact, this model has achieved the highest true positive rate, which is often the most important measure when working on imbalanced data because it denotes the percentage of examples from the minority class that have been correctly classified.

This work has to be viewed just as a preliminary analysis of the hybrid associative memories in classification problems with skewed class distributions. Other avenues for further research still remain open and therefore, it will be necessary a more exhaustive experimentation that will allow to give response to a number of important issues: (i) How do other complexities in data affect the classification performance on imbalanced data sets?, (ii) Are the gains in the true positive rate of the HACT model statistically significant?, (iii) Does the HACT memory outperform other non-neural classifiers such as support vector machines and decision trees?, and (iv) How do other associative memory models perform in the presence of class imbalance?

Acknowledgment. This work has partially been supported by the Spanish Ministry of Education and Science and the Generalitat Valenciana under grants TIN2009–14205 and PROMETEO/2010/028 and Secretaría de Investigación y Posgrado del Instituto Politécnico Nacional , respectively.

References

1. Anderson, J.A.: A simple neural network generating an interactive memory. Mathematical Biosciences 14(3-4), 197–220 (1972)
2. Deng, T., Zhao, L., Feng, L., Xue, W.: Information re-finding by context: A brain memory inspired approach. In: Proc. 20th ACM International Conference on Information and Knowledge Management, Glasgow, Scotland, UK, pp. 1553–1558 (2011)
3. Pantazi, S.V.: Methodological review: Unsupervised grammar induction and similarity retrieval in medical language processing using the Deterministic Dynamic Associative Memory (DDAM) model. Journal of Biomedical Informatics 43(5), 844–857 (2010)
4. Yaakobi, E., Bruck, J.: On the uncertainty of information retrieval in associative memories. In: Proc. IEEE International Symposium on Information Theory, Cambridge, MA, pp. 106–110 (2012)
5. Steinbuch, K.: Die Lernmatrix. Kybernetik 1, 36–45 (1961)
6. Anderson, J.A., Rosenfeld, E. (eds.): Neurocomputing: Foundations of Research. MIT Press, Cambridge (1988)
7. Kohonen, T.: Self-Organization and Associative Memory. Springer, New York (1989)
8. Santiago-Montero, R.: Clasificador Híbrido de Patrones basado en la Lernmatrix de Steinbuch y el Linear Associator de Anderson-Kohonen. Master's thesis, Centro de Investigación en Computación, Instituto Politécnico Nacional, México D.F. (2003)
9. Japkowicz, N., Shaju, S.: The class imbalance problem: A systematic study. Intelligent Data Analysis 6(5), 429–449 (2002)
10. Galar, M., Fernández, A., Barrenechea, E., Bustince, H., Herrera, F.: A review on ensembles for the class imbalance problems: Bagging, boosting, and hybrid-based approaches. IEEE Trans. on Systems, Man, and Cybernetics – Part C: Applications and Reviews 42(4), 463–484 (2012)
11. García, V., Sánchez, J.S., Mollineda, R.A.: On the effectiveness of preprocessing methods when dealing with different levels of class imbalance. Knowledge-Based Systems 25(1), 13–21 (2012)

12. Akbani, R., Kwek, S., Japkowicz, N.: Applying support vector machines to imbalanced datasets. In: Proc. 15th European Conference on Machine Learning, Pisa, Italy, pp. 39–50 (2004)
13. Cieslak, D.A., Chawla, N.V.: Learning decision trees for unbalanced data. In: Proc. 19th European Conference on Machine Learning, Antwerp, Belgium, pp. 241–256 (2008)
14. Khoshgoftaar, T.M., Van Hulse, J., Napolitano, A.: Supervised neural network modeling: An empirical investigation into learning from imbalanced data with labeling errors. IEEE Trans. on Neural Networks 21(5), 813–830 (2010)
15. Hoens, T.R., Qian, Q., Chawla, N.V., Zhou, Z.-H.: Building decision trees for the multi-class imbalance problem. In: Tan, P.-N., Chawla, S., Ho, C.K., Bailey, J. (eds.) PAKDD 2012, Part I. LNCS, vol. 7301, pp. 122–134. Springer, Heidelberg (2012)
16. Cleofas-Sánchez, L., Guzmán Escobedo, M., Valdovinos, R.M., Yáñez Márquez, C., Camacho-Nieto, O.: Using hybrid associative classifier with traslation (HACT) for studying imbalance data sets. Engineering and Research 32(1), 53–57 (2012)
17. Hasson, M.H.: Associative Neural Memories: Theory and Implementation. Oxford University Press (1993)
18. Yáñez Márquez, C., Díaz de León, J.L.: Introducción a las Memorias Asociativas. Instituto Politécnico Nacional, México D.F. (2003)
19. Alcalá-Fernández, J., Fernández, A., Luengo, J., Derrac, J., García, S., Sánchez, L., Herrera, F.: KEEL data-mining software tool: Data set repository, integration of algorithms and experimental analysis framework. Journal of Multiple-Value Logic and Soft Computing 17(2-3), 255–287 (2011)
20. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA data mining software: an update. SIGKDD Explorations Newsletter 11(1), 10–18 (2009)

Assessments Metrics for Multi-class Imbalance Learning: A Preliminary Study*

R. Alejo[1], J.A. Antonio[1], R.M. Valdovinos[2], and J.H. Pacheco-Sánchez[3]

[1] Tecnológico de Estudios Superiores de Jocotitlán, Carretera Toluca-Atlacomulco KM. 44.8, Col. Ejido de San Juan y San Agustín, 50700 Jocotitlán, México
[2] Centro Universitario UAEM Valle de Chalco, Universidad Autónoma del Estado de México Hermenegildo Galena No.3, Col. Ma. Isabel, 56615 Valle de Chalco, México
[3] Instituto Tecnológico de Toluca
Av. Tecnolgico s/n Ex-Rancho La Virgen, 52140, Metepec, México

Abstract. In this paper we study some of the most common global measures employed to measure the classifier performance on the multi-class imbalanced problems. The aim of this work consists of showing the relationship between global classifier performance (measure by global measures) and partial classifier performance, i.e., to determine if the results of global metrics match with the improved classifier performance over the minority classes. We have used five strategies to deal with the class imbalance problem over five real multi-class datasets on neural networks context.

Keywords: Multi-class imbalance, global measures, accuracy by class.

1 Introduction

Nowadays, the multi-class classification methods are increasingly required by modern applications (p.e. protein function classification, music categorization and semantic scene classification) [1]. However, while two-class classification problem has been widely studied, multi-class classification problem has not received much attention [2].

In addition, many multi-class classification applications suffer the class imbalance problem [3]. Class imbalance learning refers to a type of classification problems, where some classes are highly underrepresented compared to other classes. Several studies have shown that the class imbalance problem causes seriously negative effects on the classification performance [2,4,3], since the classifier algorithms are often biased towards the majority classes [5].

Much research has been done in addressing the class imbalance problem [4], and many new algorithms, methods and techniques have been presented [6]. Often the researchers use global measures to compare the classifier performance.

On two-class imbalance problems, global metrics to assessments of the classifier performance have been amply studied [7], and in the multi-class imbalance problems some of them have been modified to be adapted at the multi-class imbalance context

* This work has been partially supported under grants of: Projects 3072/2011 from the UAEM, PROMEP/103.5/12/4783 from the Mexican SEP and SDMAIA-010 of the TESJO.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 335–343, 2013.

[8], for example the geometric mean, F-measure or measures of the area under curve family.

The use of global measures in order to evaluate the classifier performance on multi-class imbalance problem lead to some open interesting questions: a) the global measures can measure if the classifier performance over minority and majority classes is improved? b) Global measures can give us useful information about the true behavior of the classifier over minority and majority classes? c) How could we measure efficiently whether the archive classifiers allow a tradeoff between minority and majority classes performance? Or what measure can give us real information about of true behavior of the classifier?

In this paper, we presented a preliminary study about of some of the most often global metrics applied to evaluate the classifier performance over multi-class imbalance problems on the neural networks context. Our main contribution consists in to show the relationship between the results obtained by global metrics applied to evaluate the classifier performance and the classifier behavior over the minority classes, i.e., to determine if the results of global metrics match with the improve classifier performance over the minority classes.

2 Assessments Metrics for Multi-class Imbalance Learning

The most studied metrics for assessment the classifier performing in class imbalance domains have been focused a two class imbalance problems and some of them have been modified to accommodate them at the multi-class imbalanced learning problems [6]. In this section we present some of the most common two-class imbalance metrics adapted at multi-class imbalance scenarios.

Macro average geometric (MAvG): This is defined as the geometric average of the partial accuracy of each class.

$$MAvG = (\prod_{i=1}^{J} ACC_i)^{\frac{1}{J}}, \tag{1}$$

where $ACC_j = (\ correctly\ classified\ of\ class\ j)/(\ total\ of\ samples\ of\ class j)$, i.e., the accuracy on the class j. J is the number of classes.

Mean F-measure (MFM): This measure has been widely employed in information retrieval

$$F - \text{measure}(j) = \frac{2 \cdot recall(j) \cdot precision(j)}{recall(j) + precision(j)}, \tag{2}$$

where $recall(j) = (correctly\ classified\ positives)/(total\ positives)$ and $precision\ (j) = (correctly\ classified\ positives)/(total\ predicted\ as\ positives)$; j is the index of the class considered as $positive$. Finally, mean F-measure is defined for multi-class in Reference [8] as follow:

$$MFM = \sum_{j=1}^{J} \frac{FMeasure(j)}{J}. \tag{3}$$

Macro average arithmetic (MAvA): This is defined as the arithmetic average of the partial accuracies of each class.

$$MAvA = \frac{\sum_{i=1}^{J} ACC_i}{J} . \tag{4}$$

One the most widely used techniques for the evaluation of binary classifiers in imbalanced domains is the Receiver Operating Characteristic curve (ROC), which is a tool for visualizing, organizing and selecting classifiers based on their trade-offs between true positive rates and false positive rates. Furthermore, a quantitative representation of a ROC curve is the area under it, which is known as AUC [9]. The AUC measure has been adapted at multi-class problems [8] and can be defined as follow.

AUC of each class against each other, using the uniform class distribution (AU1U):

$$AU1U = \frac{2}{\|J\|(\|J\|-1)} \sum_{j_i, j_k \epsilon J} AUC_R(j_i, j_k) \ , \tag{5}$$

where $\text{AUC}_R(j_i, j_k)$ is the AUC for each pair of classes j_i and j_k.

AUC of each class against each other, using the a priori class distribution (AU1P):

$$AU1P = \frac{2}{\|J\|(\|J\|-1)} \sum_{j_i, j_k \epsilon J} p(j) AUC_R(j_i, j_k) \ , \tag{6}$$

where $p(j)$ is a priori class distribution.

AUC of each class against the rest, using the uniform class distribution (AUNU):

$$AUNU = \frac{1}{J} \sum_{j \epsilon J} AUC_R(j, rest_j) \ , \tag{7}$$

where $rest_j$ gathers together all classes different from class j, i.e., the area under the ROC curve is computed in the approach one against all.

AUC of each class against the rest, using the a priori class distribution (AUNP):

$$AUNP = \frac{1}{J} \sum_{j \epsilon J} p(j) AUC_R(j, rest_j) \ , \tag{8}$$

this measure takes into account the prior probability of each class $(p(j))$.

3 Experimental Protocols

3.1 Database Description

We have used in our experiments five remote sensing datasets: Cayo, Feltwell Satimage, Segment and 92AV3C. Feltwell is related to an agricultural region near Felt Ville, Feltwell (UK) [10], Cayo represents a particular region in the Gulf of Mexico, and Satimage consists of the multi-spectral values of pixels in 3x3 neighborhoods in a

satellital image [11]. The Segment contains instances drawn randomly from a dataset of seven outdoor images [11]. 92AV3C dataset[1] corresponds to a hyperspectral image (145x145 pixels) taken over Northwestern Indianas Indian Pines by the AVIRIS sensor.

In order to cover Cayo in a highly imbalanced dataset some of their classes were merged as follows: join classes 1, 3, 6, 7 and 10 to integrate class 1; join classes 8, 9 and 11 to integrate class 3, finally, the rest of classes (2, 4 and 5) were obtained from the original dataset. M92AV3C is a subset of 92AV3C, it contains six classes (2, 3, 4, 6, 7 and 8) and 38 attributes. The attributes were selected using a common features selection algorithm (Best-First Search [12]) implemented in WEKA[2].

Feltwell, Satimage, Segment and 92AV3C were random under-sampled with the goal of generating severe class imbalanced datasets. A brief summary of these multi-class imbalance datasets is shown in the Table 1. Note that they are highly imbalanced datasets. For each database, a $10-$fold cross$-$validation was applied. The datasets were divided into ten equal parts, using nine folds as training set and the remaining block as test set.

Table 1. A brief summary of some basic characteristics of the datasets. The bold numbers represent the samples of minority classes. Observe that in these datasets is very easy to identify the minority and majority classes.

Dataset	Size	Attr.	Class	Class distribution
MCayo	6019	4	5	2941/**293**/2283/**322/133**
MFelt	10944	15	5	3531/2441/**91**/2295/**178**
MSat	6430	36	6	1508/1531/**104**/1356/**93/101**
MSeg	1470	19	7	330/**50**/330/330/**50/50**/330
M92AV3C	5062	38	6	**190/117**/1434/2468/747/**106**

3.2 Resampling Methods

SMOTE and random under sampling (RUS) are used in the empirical study, because they are popular approaches to deal with the class imbalance problem. However, these methods have an internal parameter that enables the user to set up the resulting class distribution obtained after the application of these methods. In this paper, we decided to add or remove examples until a balanced distribution was reached. This decision was motivated by two reasons: a) simplicity and b) effectiveness. Results obtained with the other classifiers [13], have shown that when AUC is used as a performance measure, the best class distribution for learning tends to be near the balanced class distribution.

Another common re-sampling method is the Gabriel Graph Editing (GGE). The GGE consists of applying the general idea of Wilson's algorithm [14], but using the graph neighbors of each sample instead of either the Euclidean or the norm-based distance neighborhood. The original GGE was proposed to improve the k-NN accuracy [15]. However, in Reference [16] the original GGE was adapted to do it effective in the back-propagation context.

[1] `engineering.purdue.edu/biehl/MultiSpec/hyperspectral.html`
[2] `www.cs.waikato.ac.nz/ml/weka/`

3.3 Modified Back-Propagation Algorithm (MBP)

The most popular training procedure for the MLP neural network is the back-propagation algorithm, which uses a set of training instances for the learning process. Several works have shown that the class imbalance problem generates unequal contributions to the mean square error (MSE) during the training phase in the MLP, where the major contribution to the MSE is produced by the majority class.

Given a training data set with two classes ($J = 2$), such that the size is denoted by $N = \sum_j^J n_j$ and n_j is the number of samples that belong to class j, the MSE by class can be expressed as

$$E_j(U) = \frac{1}{N} \sum_{n=1}^{n_j} \sum_{p=1}^{J} (t_p^n - z_p^n)^2 \,, \tag{9}$$

where t_p^n is the desired output and z_p^n is the actual output of the network for the sample n.

Then the overall MSE can be written in terms of $E_j(U)$ as follows:

$$E(U) = \sum_{j=1}^{J} E_j(U) = E_1(U) + E_2(U) \,. \tag{10}$$

When $n_1 << n_2$, the $E_1(U) << E_2(U)$ and $\|\nabla E_1(U)\| << \|\nabla E_2(U)\|$. Consequently, $\nabla E(U) \approx \nabla E_2(U)$. So, $-\nabla E(U)$ is not always the best direction to minimize the MSE in both classes.

The unequal contribution to the MSE can be compensated by introducing a cost function (γ) in order to avoid that the MLP ignores the minority class:

$$\begin{aligned} E(U) &= \textstyle\sum_{j=1}^{J} \gamma(j) E_j = \gamma(1) E_1(U) + \gamma(2) E_2(U) \\ &= \tfrac{1}{N} \textstyle\sum_{j=1}^{J} \gamma(j) \sum_{i=1}^{n_j} \sum_{p=1}^{J} (t_p^i - z_p^i)^2 \,, \end{aligned} \tag{11}$$

where $\gamma(1)\|\nabla E_1(U)\| \approx \gamma(2)\|\nabla E_2(U)\|$.

In this work, we define a cost function γ as $\gamma(j) = \|\nabla E_{max}(U)\| / \|\nabla E_j(U)\|$, where $\|\nabla E_{max}(U)\|$ corresponds to the majority class.

3.4 Neural Network Configuration

The MLP was trained with the standard back-propagation (SBP) and modified back-propagation (MBP) algorithm in batch mode. For each training data set, MLP was initialized ten times with different weights, i.e., the MLP was run ten times with the same training dataset. The results here included correspond to the average of those accomplished in the ten different initialization and of ten partitions. The learning rate (η) was set at 0.1 and only one hidden layer was used. The stop criterion was established at 25000 epoch or an MSE below to 0.001. The number of neurons for the hidden layer was obtained from the trial and error strategy. So, the number of neurons was 7, 6, 12, 10 and 10, for MCayo, MFelt, MSat, MSeg and M92AV3C datasets respectively.

4 Experimental Results

In order to assessment the multi-class imbalance measures (sec. 2) we have carried out an experimental comparison over five popular strategies to deal with class imbalance problem: (*i*) Modified Back-Propagation Algorithm (MBP), (*ii*) Standard Back-Propagation with Gabriel Graph Editing (SBP+GGE), (*iii*) Modified Back-Propagation with Gabriel Graph Editing (MBP + GGE), (*iv*) SMOTE and (*vi*) Random Under Sampling (RUS).

The Standard Back-Propagation Algorithm (SBP) is presented as baseline method. The datasets that were preprocessed by the SMOTE and RUS strategies were applied to the SBP algorithm. In addition, we have used seven global measures: $MAvG$, MFM, $MAvA$, $AU1U$, $AU1P$, $AUNU$ and $AUNP$ (see sec. 2) to evaluate the global classifier performance.

In the Table 2 the experimental results are presented. The columns represent a strategies used to deal with class imbalance problem, the rows show values obtained for different measures. We use the average accuracy for majority and minority classes (acc^- and acc^+ respectively), because they can give us useful information about of the classifier performance.

The values between parentheses are the average rank (AR), they provide a useful tool to compare algorithms [17]. In the AR the best performing algorithm should have the rank of 1, the second best rank 2, the third best rank 3, etc. The values of AR in Table 2 might seem wrong but not so, consider that they are average of ten folds and ten run of the MLP for each dataset (see sec. 3).

AR-GM represents the average rank for the seven global measures ($MAvG$, MFM, $MAvA$, $AU1U$, $AU1P$, $AUNU$ and $AUNP$), i.e., it is the average of the ranks of the seven global measures. We make this because the global measures tend to give similar ranks. For example see M92AV3C with RUS strategy in Table 2. With other datasets and strategies it is not as clear as with the last one. However, in general terms the global measures show similar ranks. This is consistent with the presented in Reference [8].

Table 2 shows some interesting results. For example in M92AV3C the best AR-GM is obtained by RUS and the best AR for the partial measure acc^+ is presented by RUS too, i.e., the best ranks at classes level and at global level are showed for the same strategy (RUS). In contrast to M92AV3C, MSegment shows that the best AR-GM and the worst AR are obtained for MBP, i.e., the best ranks at classes level and at global level are showed for different strategies. This imply that global measures to indicate a good classifier performance but a classes level do not performance well. This might address at wrong conclusions.

In the rest of datasets MCayo, MFeltwell, and MSatimage, the behavior of the global and partial measures follow the same tendency (see Table 2), i.e., the best AR-GM and AR for the $acc+$ are not presented by the same strategy. The results presented in Table 2 pose some interesting facts: MSegment presents a best AR-GM but the worst performance at minority class's level. Some strategies improve the minority classes perform but damage the majority classes' performance and the global measures not always reflect this situation (for example, see the MBP+GGE results).

The global measures have been introduced for very different applications and, supposedly, measure quite different things [8] but in many researches they are applied to

Table 2. Performance on five datasets measured using $MAvG$, $MAvA$, $AU1U$, $AU1P$, $AUNU$, $AUNP$ and average rank (AR)

	Measure	GGE[1]	MBP+GGE	Imbalanced[1]	MBP	RUS[1]	SMOTE[1]
	acc^-	0.918 (2.8)	0.850 (5.0)	0.924 (1.8)	0.911 (3.5)	0.899 (3.0)	0.699 (3.0)
	acc^+	0.448 (5.0)	0.814 (2.0)	0.179 (6.0)	0.585 (4.0)	0.605 (3.0)	0.968 (1.0)
	$MAvG$	0.484 (5.0)	0.820 (2.0)	0.000 (6.0)	0.692 (4.0)	0.702 (3.0)	0.822 (1.0)
	MFM	0.466 (4.0)	0.503 (1.0)	0.386 (60)	0.501 (2.0)	0.491 (3.0)	0.393 (5.0)
	$MAvA$	0.636 (5.0)	0.828 (2.0)	0.477 (6.0)	0.715 (4.0)	0.722 (3.0)	0.860 (1.0)
MCayo	$AU1U$	0.636 (5.0)	0.828 (2.0)	0.477 (6.0)	0.715 (4.0)	0.722 (3.0)	0.860 (1.0)
	$AU1P$	0.840 (4.0)	0.841 (3.0)	0.804 (5.0)	0.853 (1.0)	0.844 (2.0)	0.738 (6.0)
	$AUNU$	0.743 (5.0)	0.833 (1.0)	0.649 (6.0)	0.788 (3.0)	0.786 (4.0)	0.790 (2.0)
	$AUNP$	0.840 (4.0)	0.841 (3.0)	0.804 (5.0)	0.853 (1.0)	0.844 (2.0)	0.738 (6.0)
	AR-GM	**4.6**	**2**	**5.7**	**2.7**	**2.8**	**3.1**
	Measure	GGE[1]	MBP+GGE	Imbalanced[1]	MBP	RUS[1]	SMOTE[1]
	acc^-	0.964 (2.7)	0.926 (5.3)	0.970 (2.0)	0.954 (2.8)	0.965 (2.2)	0.908 (5.0)
	acc^+	0.305 (5.2)	0.811 (2.0)	0.188 (5.8)	0.668 (3.0)	0.424 (4.0)	0.876 (1.0)
	$MAvG$	0.000 (5.5)	0.875 (2.0)	0.000 (5.5)	0.823 (3.0)	0.530 (4.0)	0.890 (1.0)
	MFM	0.614 (5.0)	0.631 (3.0)	0.598 (6.0)	0.661 (1.0)	0.641 (2.0)	0.617 (4.0)
	$MAvA$	0.700 (5.0)	0.880 (2.0)	0.658 (6.0)	0.839 (3.0)	0.749 (4.0)	0.895 (1.0)
MFeltwel	$AU1U$	0.700 (5.0)	0.880 (2.0)	0.658 (6.0)	0.839 (3.0)	0.749 (4.0)	0.895 (1.0)
	$AU1P$	0.942 (3.5)	0.923 (4.0)	0.942 (3.5)	0.944 (2.0)	0.947 (1.0)	0.909 (5.0)
	$AUNU$	0.822 (5.0)	0.902 (2.0)	0.801 (6.0)	0.892 (3.0)	0.849 (4.0)	0.903 (1.0)
	$AUNP$	0.942 (3.5)	0.923 (4.0)	0.942 (3.5)	0.944 (2.0)	0.947 (1.0)	0.909 (5.0)
	AR-GM	**4.6**	**2.7**	**5.2**	**2.4**	**2.9**	**2.6**
	Measure	GGE[1]	MBP+GGE	Imbalanced[1]	MBP	RUS[1]	SMOTE[1]
	acc^-	0.699 (5.0)	0.649 (5.3)	0.956 (1.5)	0.954 (1.8)	0.954 (2.3)	0.885 (3.7)
	acc^+	0.849 (2.0)	0.866 (1.2)	0.369 (4.8)	0.550 (4.3)	0.498 (4.8)	0.768 (2.5)
	$MAvG$	0.739 (2.0)	0.723 (3.0)	0.000 (5.5)	0.494 (4.0)	0.000 (5.5)	0.801 (1.0)
	MFM	0.346 (4.0)	0.317 (5.0)	0.512 (3.5)	0.563 (1.0)	0.546 (2.0)	0.512 (3.5)
	$MAvA$	0.774 (2.0)	0.757 (3.0)	0.663 (6.0)	0.752 (4.0)	0.726 (5.0)	0.826 (1.0)
MSatimage	$AU1U$	0.774 (2.0)	0.757 (3.0)	0.663 (6.0)	0.752 (4.0)	0.726 (5.0)	0.826 (1.0)
	$AU1P$	0.709 (5.0)	0.664 (6.0)	0.912 (3.0)	0.924 (1.0)	0.920 (2.0)	0.875 (4.0)
	$AUNU$	0.741 (5.0)	0.709 (6.0)	0.790 (4.0)	0.839 (2.0)	0.825 (3.0)	0.851 (1.0)
	$AUNP$	0.709 (5.0)	0.664 (6.0)	0.912 (3.0)	0.924 (1.0)	0.920 (2.0)	0.875 (4.0)
	AR-GM	**3.6**	**4.6**	**4.4**	**2.4**	**3.5**	**2.2**
	Measure	GGE[1]	MBP+GGE	Imbalanced[1]	MBP	RUS[1]	SMOTE[1]
	acc^-	0.905 (3.2)	0.890 (5.0)	0.973 (1.4)	0.961 (2.9)	0.921 (3.4)	0.914 (3.9)
	acc^+	0.905 (2.0)	0.958 (2.7)	0.736 (4.3)	0.857 (5.0)	0.905 (2.7)	0.835 (3.3)
	$MAvG$	0.892 (4.0)	0.913 (1.0)	0.666 (6.0)	0.901 (3.0)	0.904 (2.0)	0.788 (5.0)
	MFM	0.625 (6.0)	0.630 (4.0)	0.706 (2.0)	0.764 (1.0)	0.656 (3.0)	0.627 (5.0)
	$MAvA$	0.905 (4.0)	0.919 (1.0)	0.871 (6.0)	0.917 (2.0)	0.914 (3.0)	0.880 (5.0)
MSegment	$AU1U$	0.905 (4.0)	0.919 (1.0)	0.871 (6.0)	0.917 (2.0)	0.914 (3.0)	0.880 (5.0)
	$AU1P$	0.906 (5.0)	0.900 (6.0)	0.947 (2.0)	0.951 (1.0)	0.920 (3.0)	0.907 (4.0)
	$AUNU$	0.906 (4.0)	0.910 (3.5)	0.910 (3.5)	0.935 (1.0)	0.917 (2.0)	0.894 (5.0)
	$AUNP$	0.906 (5.0)	0.900 (6.0)	0.947 (2.0)	0.951 (1.0)	0.920 (3.0)	0.907 (4.0)
	AR-GM	**4.6**	**3.2**	**3.9**	**1.6**	**2.7**	**4.7**
	Measure	GGE[1]	MBP+GGE	Imbalanced[1]	MBP	RUS[1]	SMOTE[1]
	acc^-	0.764 (2.0)	0.723 (3.3)	0.773 (2.3)	0.688 (4.7)	0.732 (4.0)	0.634 (4.7)
	acc^+	0.467 (4.3)	0.837 (2.0)	0.252 (6.0)	0.493 (4.7)	0.860 (1.7)	0.773 (2.3)
	$MAvG$	0.322 (5.0)	0.733 (2.0)	0.000 (6.0)	0.494 (4.0)	0.778 (1.0)	0.555 (3.0)
	MFM	0.340 (3.0)	0.356 (2.0)	0.316 (4.0)	0.285 (5.0)	0.363 (1.0)	0.274 (6.0)
	$MAvA$	0.615 (4.0)	0.780 (2.0)	0.512 (6.0)	0.590 (5.0)	0.796 (1.0)	0.703 (3.0)
M92AV3C	$AU1U$	0.615 (4.0)	0.780 (2.0)	0.512 (6.0)	0.590 (5.0)	0.796 (1.0)	0.703 (3.0)
	$AU1P$	0.720 (2.0)	0.718 (3.0)	0.702 (4.0)	0.653 (5.0)	0.726 (1.0)	0.599 (6.0)
	$AUNU$	0.677 (3.0)	0.754 (2.0)	0.615 (6.0)	0.630 (5.0)	0.759 (1.0)	0.649 (4.0)
	$AUNP$	0.720 (2.0)	0.718 (3.0)	0.702 (4.0)	0.653 (5.0)	0.726 (1.0)	0.599 (6.0)
	AR-GM	**3.3**	**2.3**	**5.1**	**4.9**	**1**	**4.4**

[1] Classification using SBP.

compare the effects of new algorithms over the classifier performance. We think it is not enough to evaluate the global performance whether it is not necessary to consider the classifier performance at classes' level too. Table 2 shows that not always the classifier performance over minority or majority classes are favorable when the global measures present a good performance of the classifier.

5 Conclusion

In this paper we study some commons global metrics ($MAvG$, $MAvA$, $AU1U$, $AU1P$, $AUNU$ and $AUNP$) used to assessment classifier performance over multiclass imbalance datasets. So we employ average accuracy of minority and majority classes (partial metrics) to contrast the results, and average rank to facility the algorithms comparison. The classifier used was a multilayer perceptron trained with the back-propagation algorithm. The study was made over five multi-class imbalanced datasets and five popular strategies to deal with the class imbalance problem.

The results obtained from five datasets used in this paper, show that not always is enough to use the global metrics to compare algorithms (over these datasets), because we observed that by one hand in some datasets these global metrics show well results and at the same time the partial metrics exhibit a bad classifier performance over minority classes. This implies that global metrics indicates a good classifier performance (on some of the datasets used in this work), but in some classes the classifier does not perform well and this might to address us at wrong conclusions. On the other hand, in some other datasets the global and partial metrics present good results, i.e., global and partial metrics agree in results. These differences in their results suggest that global metrics no always reflect the improving or damage of the strategies applied to deal with the class imbalance problem on the classifier performance over minority classes, so it is necessary to study other alternatives to assessment classifier performance over multi-class imbalance datasets.

Future work will be primarily addressed to get in depth in this topic. Priority is expand the research using more datasets and applies a significance statistical test to give better confidence to the conclusions. Also it is necessary the study of new metrics which help to assessment classifier performance over multi-class imbalance datasets and that they reflect the changes caused for the strategies used to deal with class imbalance problem on the classifier performance over the majority and minority classes.

References

1. Tsoumakas, G., Katakis, I.: Multi-label classification: An overview. Int. J. Data Warehousing and Mining, 1–13 (2007)
2. Ou, G., Murphey, Y.L.: Multi-class pattern classification using neural networks. Pattern Recognition 40(1), 4–18 (2007)
3. Wang, S., Yao, X.: Multi-class imbalance problems: Analysis and potential solutions. IEEE Transactions on IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics (99), 1–12 (2012)

4. Zhou, Z.H., Liu, X.Y.: Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Transactions on Knowledge and Data Engineering 18, 63–77 (2006)
5. Pérez-Godoy, M.D., Fernández, A., Rivera, A.J., del Jesus, M.J.: Analysis of an evolutionary rbfn design algorithm, co2rbfn, for imbalanced data sets. Pattern Recogn. Lett. 31(15), 2375–2388 (2010)
6. He, H., Garcia, E.: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering 21(9), 1263–1284 (2009)
7. García, V., Mollineda, R.A., Sánchez, J.S.: Theoretical analysis of a performance measure for imbalanced data. In: ICPR, pp. 617–620 (2010)
8. Ferri, C., Hernández-Orallo, J., Modroiu, R.: An experimental comparison of performance measures for classification. Pattern Recognition Letter 30(1), 27–38 (2009)
9. Fawcett, T.: An introduction to roc analysis. Pattern Recogn. Lett. 27, 861–874 (2006)
10. Bruzzone, L., Serpico, S.: Classification of imbalanced remote-sensing data by neural networks. Pattern Recognition Letters 18, 1323–1328 (1997)
11. A. Asuncion, D.N.: UCI machine learning repository (2007)
12. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artif. Intell. 97(1-2), 273–324 (1997)
13. Weiss, G.M., Provost, F.J.: Learning when training data are costly: The effect of class distribution on tree induction. J. Artif. Intell. Res. (JAIR) 19, 315–354 (2003)
14. Wilson, D.: Asymptotic properties of nearest neighbor rules using edited data. IEEE Transactions on Systems, Man and Cybernetics 2(4), 408–420 (1972)
15. Sánchez, J.S., Pla, F., Ferri, F.J.: Prototype selection for the nearest neighbour rule through proximity graphs. Pattern Recognition Letters 18(6), 507–513 (1997)
16. Alejo, R., Valdovinos, R., García, V., Pacheco-Sanchez, J.: A hybrid method to face class overlap and class imbalance on neural networks and multi-class scenarios. Pattern Recognition Letters 34(4), 380–388 (2012)
17. García, S., Herrera, F.: Evolutionary undersampling for classification with imbalanced datasets: Proposals and taxonomy. Evolutionary Computation 17, 275–306 (2009)

Non-conventional Control and Implementation of an Electric Wheelchair Designed to Climb Up Stairs, Controlled via Electromyography and Supported by Artificial Neural Network Processing

Martín L. Guzmán, Juan P. Pinto, Luis F. Reina, and Carlos A. Esquit

Department of Electronics Engineering, Del Valle de Guatemala University, Guatemala, Central America
{guz08041,pin08359,lfreina,caesquit}@uvg.edu.gt

Abstract. This work is part of a project which main objective is to design and implement an electric wheelchair capable of climbing up stairs. The wheelchair is operated thru silent speech non-conventional control presented on this paper using electromyographic (EMG) signals which are fed to a multiple artificial neural network (ANN) system for pattern recognition. Mechanical design of the wheelchair, powering and sensing aspects are presented on additional papers. Electromyographic signals are captured from the patient's anterior triangle of the neck muscle area using Surface Electromyography (SEMG) with electrodes in bipolar configuration. Silent speech commands from patient's gestural movements of three, four and five phonetically different words are used to interact with a Graphic User Interface (GUI) for wheelchair navigation. The system was tested on five patients, achieving an overall recognition accuracy of 95% with 0.6s maximum response time.

Keywords: Artificial Neural Network (ANN), Electromyography (EMG), IIR Digital Filtering, Digital Signal Processing, Discrete Wavelet Transform, Probabilistic Inference, Silent Speech.

1 Introduction

Bioelectric signal recognition using Electroencephalography (EEG), Electrocardiography (ECG) and Electromyography (EMG) has allowed the development of interfaces between machines and human's electrical nature. Most of these interfaces have found different applications like robot control and language recognition/translation. Nowadays, one of the main intentions behind developing these kinds of non-conventional interfaces is to allow disabled people, especially paraplegic and quadriplegic people, to control applications by which they can overcome their physical limitations.

In the case of non-conventional interfaces based on Electromyography, some research has been done at the University of Hawaii in association with the NASA Ames Research Center, where researchers developed a sub-auditory speech recognition

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 344–353, 2013.

interface using electromyogram sensors [1]. Additional research has been carried out at the Karlsruhe Institute of Technology (KIT), where they have developed a speech recognition system based on myoelectric signals using seven EMG channels [2]. Related work using bioelectric signals was also developed at Del Valle de Guatemala University as part of the ANIMA project [3]. ANIMA implements several non-conventional interfaces, one of them using Electrooculography to control a robot.

Electromyographic signals can be obtained through Surface Electromyography (SEMG) using surface electrodes over the muscles of interest or Needle Electromyography (NEMG) using needles that reach the muscles of interest penetrating the skin. The EMG signals are produced by ion exchange activity in the relaxation and contraction movements of muscle fibers that are activated by motoneurons (central nervous system cells) [4].

To be able to observe the potentials from a specific muscular region in human body (micro-volt scale), a comparison between two electrodes placed at the muscle region of interest and a reference electrode placed at a reference muscle area is needed.

The usable frequency range of surface electromyograhic signals is 20 Hz to 500 Hz for most applications. Frequencies outside this range need to be filtered out to keep only the frequencies of interest. Amplification and digitization stages are required in our application in order to send information of the EMG signal (produced by the silent speech gestural movements of the patient) to a computer for further processing.

An artificial neural network is a mathematical or computational model inspired by various aspects of the structure and function of biological neural networks in order to emulate the behavior of the human brain. Artificial neural networks have a great ability to extract patterns from complex and rich data. The adaptive learning, tolerance fault and flexibility are the main advantages behind the idea of using artificial neural networks to perform pattern recognition [5]. On this work a single EMG channel was used to recognize what the patient is trying to say via a silent speech recognition system. For the recognition system multiple feedforward multilayer artificial neural networks were used. Probabilistic inference based on a multiple ANN system was used in order to add robustness to the application.

2 Experimental Setup

This work uses silent speech gestural movements from a specific set of phonetically different words in order to generate movement commands for wheelchair navigation. The set is composed of the Spanish words “Eco”, “Noviembre”, “Omega”, “Sigma”, “Teléfono” and “Uniforme”. The system implemented on this work consists of two main modules: Electromyographic Signal Module and Pattern Recognition Module.

2.1 Electromyographic Signal Module

Fig. 1 shows all the necessary stages for electromyographic signal capturing, conditioning and transmission to the Pattern Recognition module for further processing.

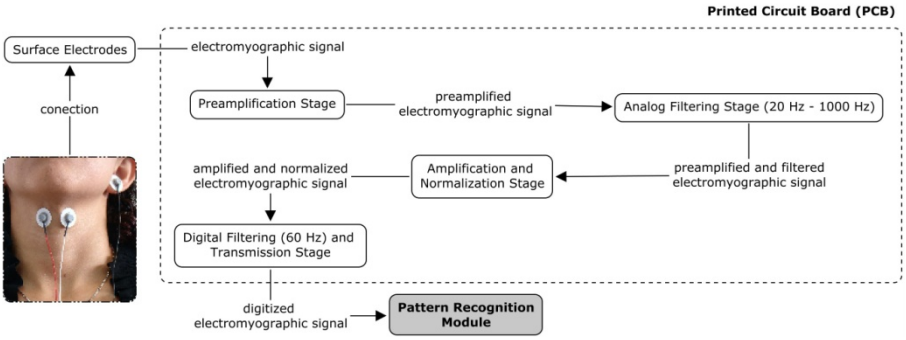


Fig. 1. Electromyographic Signal Module

This module used three Ag/AgCl F-E9M-40 surface electrodes (from Grass Technologies) connected to the printed circuit board implemented on this work (Fig. 4). Surface Electromyography was chosen for simplicity and safety. Two surface electrodes were placed in bipolar configuration on the anterior triangle of the neck muscle area and a reference electrode was placed on the left ear lobule (Fig. 2).

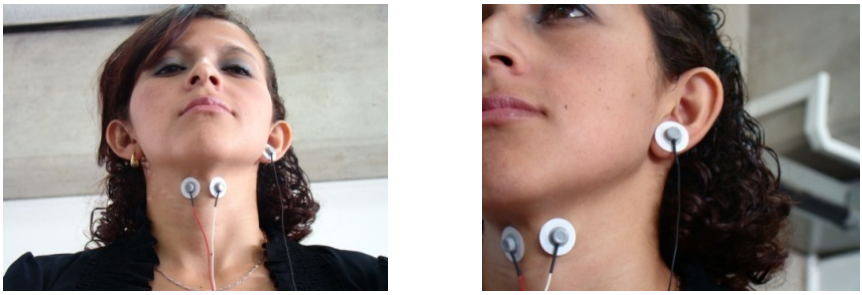


Fig. 2. Surface electrodes placed on the area of interest for electromyographic module

Preamplification Stage. Proper amplitude is needed before any analog or digital signal processing. Instrumental amplifiers offered by different manufacturers are commonly used to achieve this goal. For this stage an AD620 (from Analog Devices) instrumental amplifier was used. This IC offers up to 1000X adjustable differential gain, high-input impedance and high CMR (Common-Mode Rejection). The gain at this stage is adjusted using a precision potentiometer located on the implemented printed circuit board.

Analog Filtering Stage. Analog filtering ensures that the EMG signal will not contain any noise due to motion artifacts nor any unnecessary frequency components and guarantees a correct sampling for the digitizing stage. This filtering consists of an eight-pole anti-aliasing Butterworth filter with 1000 Hz cut-off frequency and a

fourth-pole high-pass Butterworth filter with 20 Hz cut-off frequency so that the signal's DC component is eliminated. The output of this stage is an EMG signal in the hundreds of mili-volt range and still contaminated by the 60-Hz AC power-line noise.

Amplification and Normalization Stage. Proper amplitude and voltage limits need to be guaranteed prior to digitization. An operational amplifier was used in order to add a DC offset voltage and to amplify the EMG signal. The output of this circuit is an EMG signal centered in the 0–5 V range so that it is compatible with the digital signal processor's ADC (Analog to Digital Converter). Both the DC offset voltage and the gain are adjusted using precision potentiometers located on the implemented printed circuit board. A voltage limiter circuit built with two high-speed diodes was used to keep the EMG signal in the 0–5 V range.

Digital Filtering (60-Hz) and Transmission Stage. Digitization and filtering was implemented on a dsPIC30F4013 running at 20 MIPS with 2.6 kHz sampling frequency. Samples serve as input to a difference equation that implements a 60-Hz notch second order IIR digital filter. Two bytes are used to store each 12-bit sample of the EMG signal and a third byte is used as a checksum for the computer to detect corrupt data. The dsPIC microcontroller transfers data to the computer using a UART interface at 115,200 bps.

2.2 Pattern Recognition Module

Fig. 3 shows all the necessary stages for discretization, detection and recognition of silent speech gestural movements in order to generate commands to be sent to the Wheelchair Control Module (not addressed on this paper).

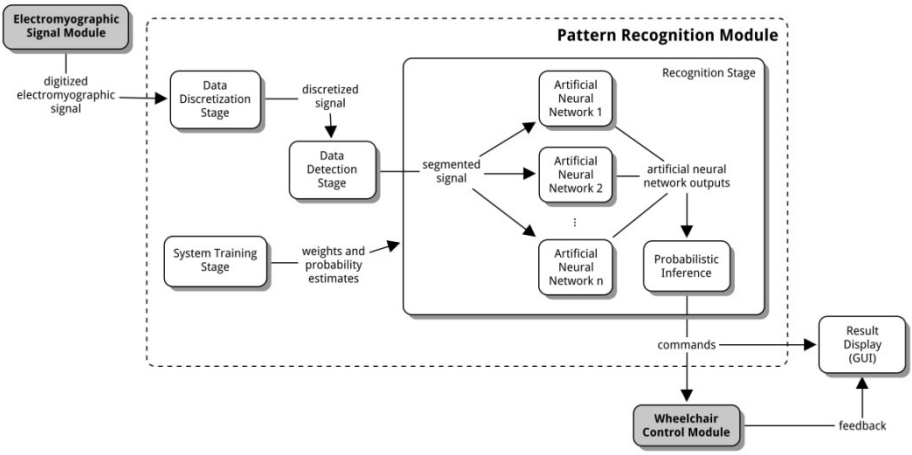


Fig. 3. Pattern Recognition Module

Discretization Stage. Data discretization is the first step in which the EMG signal is modified once entered into the computer. The stage begins by performing data downscaling for graphical representation purposes and offset adjusting for further processing. Discretization seeks to separate the EMG data and divide it into data blocks in order to accelerate the detection process. On this work the block size was defined as 0.05 seconds of data, which was useful for distinguishing consecutive gestural commands from consecutive syllables within a single gestural command.

Data Detection Stage. Once the signal is divided into a set of blocks, the system performs a data detection method. The aim of this method is to classify all blocks as useful blocks (gestural data present) and not useful blocks (none gestural data present). This data detection method was implemented using confidence intervals based on signal amplitude. These confidence intervals were previously defined by preliminary tests and are calibrated according to the current signal data when the patient is not performing any gestural movements. This stage provides segments (a group of blocks) of a size corresponding to one second (the approximate time it takes a patient to gesticulate a word). Zero-filling is used to complete segments with less of one second of gestural data.

Recognition Stage. This stage receives block segments provided by the data detection stage. These block segments are analyzed by multiple feedforward multilayer artificial neural networks (initial tests were performed using a single feedforward multilayer artificial neural network and final tests were performed using 25 different artificial neural networks). A fifth-order Daubechies Discrete Wavelet Transform was repeatedly applied to the data as a data reduction technique, in order to reduce the amount of data in the EMG signal. The output of this transform was fed into the multiple artificial neural network system. All artificial neural networks design (number of hidden layers and artificial neurons per layer) was based on heuristic methods.

Artificial neural networks provide outputs which are analyzed by a probabilistic inference method based on probability estimates of success and failure of each neural network for each command. This method seeks the probability for a sample to be or not to be a specific command, so the Bayes Theorem was applied to the specific case of a binary partition. The model used to determine the probability for a sample being a specific command is given by (1).

$$\mathrm{P}(C|E_1, E_2, \ldots, E_n) = \frac{P(C)\prod_{i=1}^{n} P(E_i|C)}{P(C)\prod_{i=1}^{n} P(E_i|C) + P(\neg C)\prod_{i=1}^{n} P(E_i|\neg C)} \tag{1}$$

Where, $P(C)$ represents the probability that a sample is a command (depends solely on the number of commands) and $P(E_i|C)$ represents the observations obtained by the $i\text{-}th$ artificial neural network for the command C. These observations consist of a group of independent events. Once the method has computed the probability of each command, a rule indicating the acceptance is carried out to determine whether the sample corresponds to a valid command or to an unidentified command. The rule of acceptance is based on individual probability magnitude.

System Training Stage. One of the main tasks is the system training. This stage is used to train the artificial neural networks which perform pattern recognition as explained in the previous stage. Supervised learning was chosen, which required the recording of patient samples. System training is personalized, so it was necessary to record samples from all patients individually. Several samples were recorded for each gestural command, between 250 and 400 samples per command depending on the patient's difficulty to produce gestural movements. A session-independent training method as explained in [6] was chosen to obtain robust pattern recognition.

2.3 Graphic User Interface

The set of gestural commands is used to interact with a Graphic User Interface (GUI) in order to navigate through a menu of options representing predefined instructions to move the wheelchair. This GUI was implemented using Python language and consists of several screens with either three, four or five options.

The option set allows tasks like rotating the wheelchair at a desired angle and moving forward, backwards and sideways. The option set also allows the use of a camera so the patient can select a destination point for the wheelchair to move with obstacle avoidance (camera navigation algorithm not presented on this paper).

3 Results

Fig. 4 shows the implemented printed circuit board for the Electromyographic Signal Module. All the important components and terminals are properly identified over the PCB image.

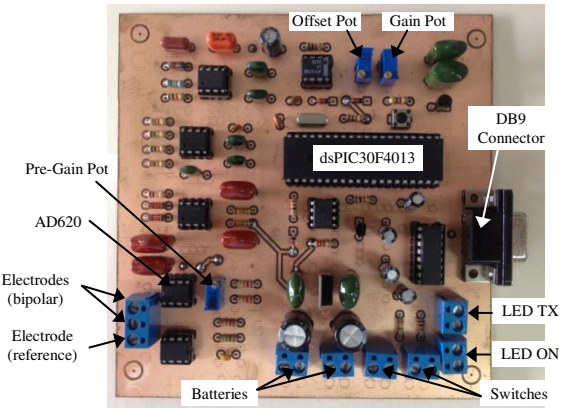


Fig. 4. Electromyographic Signal printed circuit board

Fig. 5 shows oscilloscope images for electromyographic signals after the preamplification stage (in blue) and the normalization and amplification stage (in yellow) for two different gestural commands: "Eco" (left image) and "Noviembre" (right image).

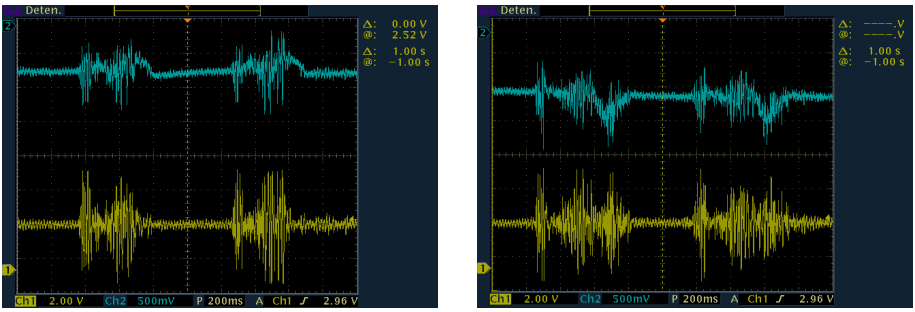


Fig. 5. Electromyographic signals for “Eco” (left) and “Noviembre” (right)

The transfer function in z-domain that describes the 60-Hz notch second order IIR digital filter implemented as a difference equation in a dsPIC microcontroller is given by (2).

$$H(z) = \frac{0.96436171 - 1.90848416z^{-1} + 0.96436171z^{-2}}{1 - 1.90848416z^{-1} + 0.92872341z^{-2}} \tag{2}$$

Fig. 6 and Fig. 7 show examples of sampled Spanish words “Eco”, “Sigma”, “Noviembre” and “Omega”. These samples are the result of the data detection stage.

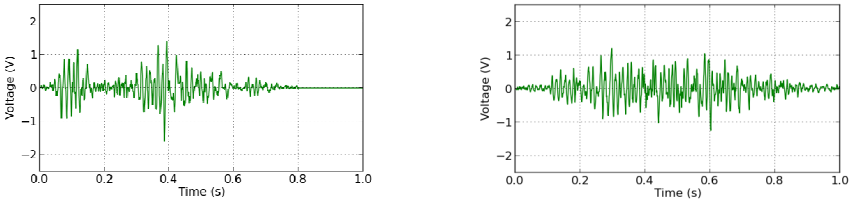


Fig. 6. Silent speech for the words “Eco” (left) and “Sigma” (right)

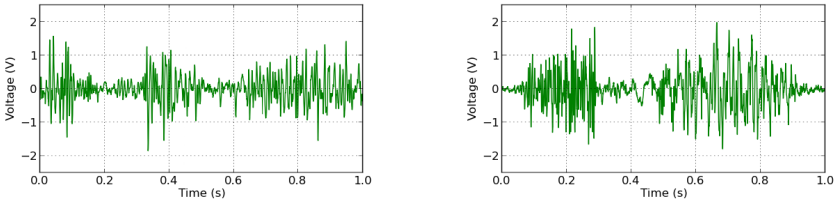


Fig. 7. Silent speech for the words “Noviembre” (left) and “Omega” (right)

Fig. 8 shows a 10-second silent speech recording in which the patient gestured the Spanish words “Uniforme”, “Eco”, “Teléfono”, “Sigma” and “Noviembre”. The data detection line was plotted (in red) alongside the EMG signal (in green) with a positive value (always a 1V value) for useful data and a negative value for not useful data (always a -1V value).

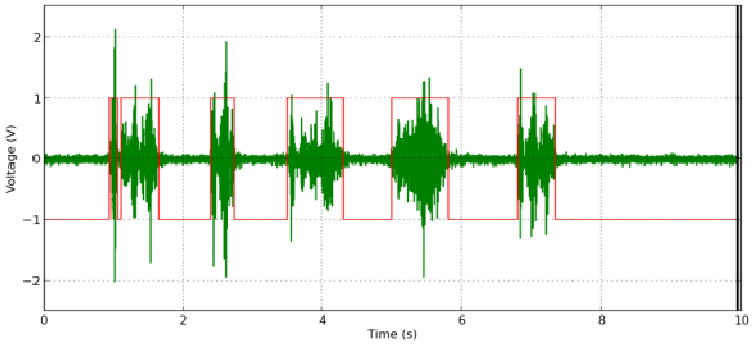


Fig. 8. Silent speech for the Spanish words “Uniforme”, “Eco”, “Teléfono”, “Sigma” and “Noviembre”

Table 1 shows a summary of the results obtained for four different gestural commands. The system was tested on five patients. The worst-case response time for the ANN gestural pattern recognition was 0.6 seconds. The ANN system was programmed on Python and tested on a 32-bit architecture 4GB RAM Intel Core 2 Duo 1.5 GHz processor running Ubuntu, Linux. Single ANN system showed 86% of accuracy (recognized). Results on Table 1 correspond to the final multiple ANN system with probabilistic inference.

Table 1. Summary of gestural command recognition for all tested patients

Patient	Recognized	Misrecognized	Unrecognized	Number of tests
1	94.17%	3.33%	2.50%	120
2	97.50%	1.67%	0.83%	120
3	95.00%	2.50%	2.50%	120
4	93.33%	3.33%	3.33%	120
5	95.00%	0.83%	4.17%	120
Average	**95.00%**	**2.33%**	**2.67%**	

Fig. 9 shows the GUI implemented on Python so that the patient can interact with the wheelchair by gestural commands. Fig. 10 shows one of the patients while testing the system.

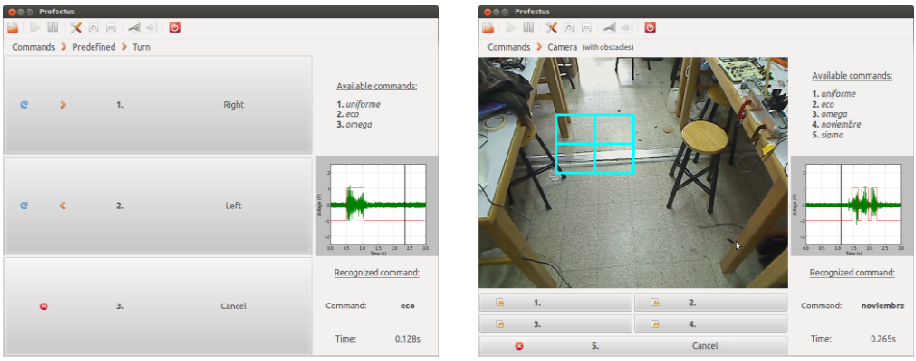


Fig. 9. Graphic User Interface for interaction between patient and wheelchair

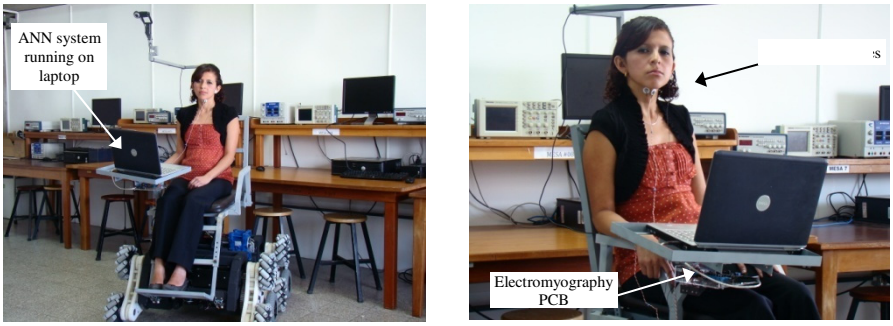


Fig. 10. Electric wheelchair using the implemented non-conventional control

4 Conclusions

A system capable of acquiring electromyographic signals from five different subject's gestural movements and recognizing a specific set of words using artificial neural networks was successfully achieved with an accuracy of 95% for four commands. Tests for three and five commands showed accuracy over 96%. All tests showed a 0.6-second maximum response time.

Tests showed that location and separation of surface electrodes play an important role on proper electromyography signal capturing.

IIR filtering was chosen over FIR filtering due to its fewer required coefficients which implies faster processing and/or the use of additional channels on the same hardware.

Repetitive use of Wavelet Transform applied to electromyographic data showed to be useful at achieving high accuracy while speeding up the whole ANN system.

The use of multiple artificial neural networks and probabilistic inference offered an average 9% improvement on accuracy from the original 86% for single artificial neural network system.

5 Future Work

Future research has to be done to test the system on handicapped subjects (paraplegic and quadriplegic people). Additional gestural commands can be tested so that the system might be useful for more complex interaction applications. Analyzing the possibility for migrating the ANN system running on the laptop to an FPGA-based system would be of interest in terms of cost, area, weight, response time and portability. The Electromyographic Signal Module could be redesigned to include additional signal channels so that there would be more useful data for further processing.

Satisfactory results were obtained for a small vocabulary as used on this work, but exploring similar systems using other approaches like HMM or SVM for EMG pattern recognition would be interesting in order to evaluate the performance for extensive vocabularies in the case of other applications which may require much more than a few words.

Genetic algorithms can be used instead of heuristic methods for ANN design in order to evaluate possible accuracy improvement.

References

1. Jorgensen, C., Binsted, K.: Sub-Auditory Speech Recognition (2006), http://www2.hawaii.edu/~binsted/papers/BinstedJorgensenHICSS2006.pdf
2. Maier-Hein, L., Metze, F., Schultz, T., Waibel, A.: Session Independent Non-Audible Speech Recognition using Surface Electromyography (2006), http://mbi.dkfz-heidelberg.de/mbi/TR/Papers/P14-05.pdf
3. Valdeavellano Wurmser, M.: ANIMA: Métodos no convencionales de interfaz en el control de robots a través de la electroencefalografía y la electrooculografía: Módulo Ocular. Del Valle de Guatemala University, Guatemala (2009)
4. Konrad, P.: The ABC of EMG: A Practical Introduction to Kinesiological Electromyography (April 2005), http://demotu.org/aulas/controle/ABCofEMG.pdf
5. Gestal Pose, M.: Introducción a las Redes de Neuronas Artificiales, http://sabia.tic.udc.es/mgestal/cv/RNATutorial/TutorialRNA.pdf
6. Wand, M., Schultz, T.: Session Independent EMG-Based Speech Recognition (2011), http://csl.anthropomatik.kit.edu/downloads/WandSchultz_Biosignals2011.pdf

A Question Answering System for Reading Comprehension Tests

Helena Gómez-Adorno, David Pinto, and Darnes Vilariño

Faculty of Computer Science
Benemérita Universidad Autónoma de Puebla
Av. San Claudio y 14 Sur, C.P. 72570, Puebla, Mexico
{helena.gomez,dpinto,darnes}@cs.buap.mx
http://www.cs.buap.mx/

Abstract. In this paper it is presented a methodology for tackling the problem of question answering for reading comprehension tests. The implemented system accepts a document as input and it answers multiple choice questions about it. It uses the Lucene information retrieval engine for carrying out information extraction employing additional automated linguistic processing such as stemming, anaphora resolution and part-of-speech tagging. The proposed approach validates the answers, by comparing the text retrieved by Lucene for each question with respect to its candidate answers. For this purpose, a validation based on textual entailment is executed. We have evaluated the experiments carried out in order to verify the quality of the methodology proposed using two corpora widely used in international forums. The obtained results show that the proposed system selects the correct answer to a given question with a percentage of 33-37%, a result that overcomes the average of all the runs submitted in the QA4MRE task of the CLEF 2011 and 2012.

Keywords: Question answering system, reading comprehension, information retrieval, textual entailment.

1 Introduction

Reading comprehension is a task associated with the ability of a reader to understand the main ideas written in a given text. This understanding comes basically from the knowledge that is triggered to the reader by observing the different words that appear in the text. Analyzing a text is quite different than just reading it. The goal of reading comprehension is to understand the main ideas implied in the text. With the aim of evaluate the level of reading comprehension, there exist tests that ask the students to read a story or article and answer a list of questions about it. See Figure 1 in which an example of reading comprehension test is presented.

Answering a question about a given text in an automatic way to evaluate the understanding of that text, is a very difficult task that oftenly has been tackled in the literature through some Natural Language Processing (NLP) techniques,

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 354–363, 2013.

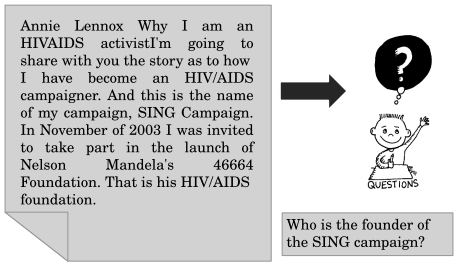


Fig. 1. Reading Comprehension test document and question

such as Question Answering (QA). Information retrieval and QA are related, however, QA assumes that given a query, the result must be the correct answer of that question, instead of a number of references to documents that contain the answer.

In this paper we present some experiments for exploring question answering architectures that can be applied to reading comprehension tests as an evaluation method for language understanding systems (machine reading systems). Such tests take the form of standardized multiple-choice diagnostic reading skill tests.

The main idea behind QA systems for reading comprehension tests is to answer questions based on a single document. This approach is different from that of traditional QA systems, in which they have a very large corpus for searching the requested information, which implies in some cases a very different system architecture.

There exist a seminal work on QA for reading comprehension tests written in the late 90's by *Hirschman* which describes an automatic reading comprehension system, **Deep Read**. The system receives as input a document, and answers questions about it. The authors designed a corpus with 60 test documents, and each document contained five associated questions, along with their correct answers. In a first approach, the system used basic techniques based on pattern matching (bag of words), enriched with automated linguistic processing (stemming, named entity identification, semantic class identification, and pronoun resolution) for extracting the sentence containing the answer. The system reported between 30-40% of precision [1].

Other studies were conducted on this research line using the same corpus. The work presented by *Charniak* [2], for instance, apply techniques that achieve small improvements over the *Hirschman's* research work. These techniques range from simple (give higher weights to verbs in the answer selection) to more complex (use specific techniques to answer specific types of questions). In the *Riloff's* work [3], a rule-based system is presented, **Quarc**. It uses heuristic rules to look for lexical and semantic clues in the question and the answer. This system finds the correct answer 40% of the time. Finally, *Hwee Tou Ng* [4] presents an approach based on machine learning techniques, in particular decision trees algorithms,

with a set of features extracted from the corpus. This was the first work that achieved competitive results in the task of QA for reading comprehension tests using these techniques (39% accuracy).

The QA for reading comprehension tests field has been inactive for a long time, due to the lack of agreement in the way the systems evaluation should be done [5] . In 2011, and later in the 2012, the CLEF conference[1] proposed a QA task for *Machine Reading (MR) systems* evaluation called QA4MRE. The task consists of reading a document and identifying answers for a set of questions about the information that is expressed or implied in the text. The questions are written in the form of multiple choices; each question has 5 different options, and only one option is the correct answer. The detection of the correct answer is specifically designed to require various types of inference, and the consideration of prior knowledge acquired from a collection of reference documents [6,7].

The QA4MRE task encourage the interest in this research line, because it provides a single evaluation platform for the experimentation with new techniques and methodologies towards giving a solution to this problem. In this sense we can take the systems presented in this conference as state-of-the-art work for this research field.

The rest of the paper is organized as follows. Section 2 describes the System Architecture. Section 3 presents the evaluation results in a collection of documents of the QA4MRE task at CLEF 2011 and 2012, and compares these results against those reported in literature. Finally, Section 4 presents the conclusions obtained, so that it outlines some future work directions.

2 System Architecture

The proposed architecture is made up of three main modules. Each of these modules are described in the following subsections.

2.1 Document Processing

The document processing module consists of three sub-modules: an XML Interpreter, a Query Analyzer, and a Document Pre-Processor. A detailed description of each submodule follows.

XML Parser: The XML parser receives as input a corpus structured in XML format which contains all the documents of the reading comprehension test, along with their respective questions and multiple choice answers. The XML parser extracts the documents, questions and associated answers. It stores the questions and answers identifying them according to the document to which they belong in order to be used in the following processes.

[1] The Cross-Lingual Evaluation Forum: `http://www.clef-initiative.eu`

Query Analyzer: This module receives as input the question set associated with the documents. A Part-Of-Speech (POS) tagger is applied to the questions in order to identify the "question keywords" (what, where, when, who, etc.), and the result is passed to the *hypothesis generation* module (this module will be explained more into detail in Section 2.2).

Document Pre-Processing: This module has the task of performing anaphora resolution for the documents associated with the questions. It has been observed that applying anaphora resolution in QA systems improves the results obtained, in terms of precision [8]. In the experiments carried out in this paper, the JavaRAP [2] system was used for anaphora resolution. It resolves third person pronouns, lexical anaphora, and identifies pleonastic pronouns.

Given that JavaRAP does not resolve anaphora of first-person pronouns, we added a process for the resolution of these cases. The process added is as follows:

1. Identify the author of the document, which is usually the first name in the document. For this purpose, the Stanford POS tagger [3] was used.
2. Each personal pronoun in the first person set PRP={"I", "me", "my", "myself"} generally refers to the author.
3. Replace each term of the document that is in the PRP set, by the document author name identified in step 1.

Take the following text for showing how this procedure works:

Emily Oster *flips our thinking on AIDS in Africa. So* ***I*** *want to talk to you today about AIDS in sub-Saharan Africa.* ***I*** *imagine you all know something about AIDS*

Step 1: ***Emily_NNP Oster_NNP*** *flips_VBZ our_PRP$ thinking_NN on_IN AIDS_NNP in_IN AfricaSo_NNP.*

In this case, the 2 first terms that have the NNP label are selected to identify the author. **Author = Emily Oster**.

Step 2: ***I_PRP*** *want_VBP to_TO talk_VB to_TO you_PRP today_NN about_IN AIDS_NNP in_IN sub-Saharan_NNP Africa_NNP ._. **I_PRP** imagine_VBP you_PRP all_DT know_VBP something_NN about_IN AIDS_NNP ._..*
In these lines two words are identify within the defined PRP set.

Step 3: *So* ***Emily Oster*** *want to talk to you today about AIDS in sub-Saharan Africa.* ***Emily Oster*** *imagine you all know something about AIDS.*

Here the words within the PRP set are replaced by the name of the author of the document.

[2] `http://wing.comp.nus.edu.sg/ qiu/NLPTools/JavaRAP.html`
[3] `http://nlp.stanford.edu/software/tagger.shtml`

2.2 Information Extraction

The information extraction module consists of the following two submodules: Hypothesis Generation and Information Retrieval. Both submodules are described as follows.

Hypothesis Generation. This module receives as input the set of questions with their multiple choice answers, which were previously processed in the module *Questions Analysis*. In this work we define hypothesis as the concatenation of the question with each of the possible answers. This hypothesis is intended to become the input to the Information Retrieval (IR) module, i.e., the query. In order to generate the hypothesis, first the "question keyword" is identified and subsequently replaced by each of the five possible answers, thereby obtaining five hypotheses for each question. The process is illustrated in the following example:

Question: **Where** was Elizabeth Pisani's friend incarcerated?

Answer 1: in the Philippines
Answer 2: in the Taiwan Island
Answer 3: in the Islands of Malaysia
Answer 4: in the Greater Sunda Islands archipelago
Answer 5: in the Lesser Sunda Islands archipelago

From the previous question and their possible answers, we obtain the following hypotheses:

hypothesis 1: **in the Philippines** was Elizabeth Pisani's friend incarcerated?
hypothesis 2: **in the Taiwan Island** was Elizabeth Pisani's friend incarcerated?
hypothesis 3: **in the Islands of Malaysia** was Elizabeth Pisani's friend incarcerated?
hypothesis 4: **in the Greater Sunda Islands archipelago** was Elizabeth Pisani's friend incarcerated?
hypothesis 5: **in the Lesser Sunda Islands archipelago** was Elizabeth Pisani's friend incarcerated?

The benefit of using these hypotheses as queries for the IR module is to search passages containing words that are in both, the question and the multiple-choice answer, instead of search passages containing words from the question and the answer, independently.

Information Retrieval. The IR module was built using the Lucene[4] IR library. It is responsible for indexing the document collection, and for the further passage retrieval, given a query. Each hypothesis obtained in the *hypothesis generation* module is processed in order to identify the query keywords, removing *stop words* (using the stop word list of python NLTK[5]). Every processed hypothesis is sent to the IR module.

[4] `http://lucene.apache.org/core/`
[5] `http://nltk.org/`

The IR module returns a relevant passage for each hypothesis. This passage is used as a support text to decide whether or not the hypothesis can be the right answer. For each hypothesis the first passage returned is taken (only one), which is considered the most important one. This process generates a pair "Hypothesis + Passage (H-P)", along with a lexical similarity score calculated by lucene.

2.3 Answer Validation

The answer validation module aims to assign a score based on the textual entailment judgment to the pair H-P generated in the *Information Retrieval* module.

It has been proven that the textual entailment judgment may improve the performance of the hypothesis validation, given a support text, which in this case is the retrieved passage [9,10,11]. The aim of this module is to obtain the textual entailment judgment over all the $H - P$ pairs that it receives as input. In order to determine whether or not the passage P implies an hypothesis H, we implemented an approach based in an research work[12] presented in the Crosslingual Textual Entailment task of the SEMEVAL-2012[6]. In this work the set provided in that conference is used as a training data. The textual entailment judgment is performed over the hypotheses-passages set as test data.

For this particular problem all the previously developed models were tested, determining that the best performance is obtained when the following 10 features are used: the number of n-grams of words and characters ($n = 1, \cdots, 5$), which share each pair of sentences. In addition, the length of both sentences are included to the feature set, since it has been proven to help to obtain the textual entailment judgment. Since this problem can be seen as a classification one, after several experiments, it was decided to use a 4-layer neural network, using the WEKA[7] data mining tool.

2.4 Answer Selection

For this last phase of the system, the method shown in Algorithm 1 is developed based on the following rules:

1. Check the entailment judgment between the hypothesis and the recovered passage. If the judgment is "no_entailment", then this algorithm discards this answer, in other case, the lexical similarity score obtained by lucene and the prediction percentage given by the textual entailment judgement are added.
2. For each question, the answer obtaining the highest sum of scores is selected as the correct answer.

The reason for discarding the hypothesis with "no_entailment" judgment is that even thought the IR module returned a passage for the hypothesis, this one does

[6] `http://www.cs.york.ac.uk/semeval-2012/task8/`
[7] `http://www.cs.waikato.ac.nz/ml/weka/`

Algorithm 1. *AnswerSelection*

Input: Hypotesis
Input: Support Text
Input: *lucene_score* : Lexical similarity score given by lucene
Input: *te_prediction*: Textual entailment Prediction percentage given by weka
Input: *te_judgment* : Textual entailment judgment
Output: List of correct answer for each question

foreach *pair* ($a_k = Hypotesis_k + Support\_text_k$) $d_i\_q_j$
where $i = 1 \ldots 12, j = 1 \ldots 10, k = 1 \ldots 5$ **do**
  if *judgment* = "*no_entailment*" **then**
    discards that possible answer;
  else
    $score[d_i, q_j, a_k] = lucene\_score + te\_prediction$;
foreach i, j, k *in* *score* **do**
  if $mayor[i, j] < score[d_i, q_j, a_k]$ **then**
    $mayor[i, j] = score[d_i, q_j, a_k]$;
    $mayorId[i, j] = k$;
foreach i, j *in* $d_i\_q_j$ **do**
  return *i, j, mayorId[i,j]*

not share sufficient information to support the selection of that hypothesis as the correct answer to the question. The use of the lexical similarity score obtained by lucene allows the system to determine which answer is more similar with its support text. The textual entailment prediction value obtained through the Weka tool adds extra information when the correct answer is selected.

3 Experimental Results

This section describes the datasets used for evaluating the methodology proposed in this paper. Additionally, the results obtained in the experiments carried out are reported and discussed.

3.1 Corpus Description - QA4MRE Task

In order to determine the performance of the system proposed in this paper we used the corpora provided in the QA4MRE task of the CLEF 2011 and 2012. The features of the two test datasets are detailed in Table 1.

3.2 Obtained Results and Error Analysis

The main measure used in this evaluation campaign is c@1, which is defined as shown in equation 1. This measure is defined in the QA4MRE task at CLEF 2011 with the purpose of allowing the systems to decide whether or not to answer

Table 1. Features of the two test datasets (QA4MRE 2011 and 2012 tasks)

Features	**2011**	**2012**
1. Topics	3	4
2. Topic details	Climate Change, Music & Society, and AIDS	Climate Change, Music & Society, Alzheimer and AIDS
2. Reading tests (documents)	4	4
3. Questions per document	10	10
4. Multiple-choice answers per question	5	5
5. Total of questions	120	160
6. Total of answers	600	800

a given question. The aim of this procedure is to reduce the amount of incorrect answers, maintaining the number of correct ones.

$$c@1 = \frac{1}{n}(n_R + n_U \frac{n_R}{n}) \tag{1}$$

where:
n_R: number of correctly answered questions.
n_U: number of unanswered questions.
n: total number of questions.

Table 2 presents the obtained results. It can be observed that in both data sets, the 2011 data set and the 2012 data set, the average over all best runs and over all runs were exceeded.

Table 2. Comparison of the results obtained by our QARCT system and the Average Scores over all runs and over best runs

Description	**2011**	**2012**
QARCT	**0.33**	**0.37**
Avg. over all best runs	0.28	0.32
Avg. over all runs	0.21	0.26
Random baseline	0.20	0.20

For evaluation purposes of this task and given the evaluation measure, it is considered that is better do not give answer at all than provide an incorrect answer. In that sense, we analized the number of questions that have been incorrectly answered. According to this error analysis, it was considered important to experiment with other passages division models: using n sentences rather than a single one, defining a window of n words that should include the passage. Unfortunately, this experiment did not improve the precision of the system, therefore it was discarded.

After an extensive review of the passages retrieved by each hypothesis, it was noticed that the *IR module* is not consistent returning, in the first place, the passage containing the answer to the question, but instead the passages were returned sometimes in the second, third or even in the fifth place. This issue makes the validation step useless for these cases. It is therefore necessary that the *IR module* retrieves a 100% of the correct passages. As a possible solution it is planned to extend the number of the recovered passages from 1 to 5, for each hypothesis.

Another problem found in the *Answer Selection* module is that the lexical similarity score given by lucene is not enough to capture the similarity between the hypothesis and the support text, when they do not share the same words. To overcome this problem, two things can be done: 1) To include a query expansion module trying to add synonyms, hyperonyms, etc, in order to obtain a higher lexical similarity, and 2) To add a semantic similarity algorithm which can discover the degree of similarity between two sentences, even though they do not share the same words exactly. For example in the hypothesis: "she esteems him is Annie Lennox 's opinion about Nelson Mandela", the recovered passage is "Everyone one in the world respects Nelson Mandela, everyone reveres Nelson Mandela"; but the score assigned by lucene is too small and it does not select that answer as the correct one. The addition of semantic similarity score will help to raise the score of this two phrases and select the correct answer because it will probably find the relation between the words "esteems", "revers" and "respect".

4 Conclusion and Future Work

In this paper we have presented a complete methodology for tackling the problem of question answering for reading comprehension tests. Additional modules can be added to this methodology, or maybe a refinement of each step presented may be done. However, we consider that the proposal is complete in terms of such modules needed in order to solve the aforementioned problem.

The implementation of the first person anaphora resolution algorithm helped Lucene to find more precise information for retrieving more accurately those passages that contain the possible answer. This type of anaphora resolution was not implemented in the original software used in the experiments, therefore, we consider this contribution very important in this research work.

By adding the textual entailment module to the basic measures based on lexical similarity, it allowed to correctly answer a higher number of questions. Additionally, this module allowed to determine whether or not to answer the given question, which have a high impact in the final scores of the proposed system.

We have compared the performance of the system presented in this paper with those reported in the QA4MRE task of CLEF 2011 and 2012. We observed that the obtained results overcome the average of the runs submitted to that conference. There still is more research to do as future work.

It is planned to analyze the use of Machine Learning techniques for the answer validation module. For this purpose, it is necessary to determine the features that fulfills an answer when it is correct or incorrect. Based on these features, a classifier should be trained in order to obtain a model capable of identify whether or not an answer is correct or not. Additionally, we are considering to implement semantic similarity measures with the aim of improving the level of matching between the hypothesis and the possible passages of the target text, when these two sentences do not share the exactly same words, but those that are semantic similar.

References

1. Hirschman, L., Light, M., Breck, E., Burger, J.D.: Deep read: a reading comprehension system. In: Proceedings of the 37th meeting of the Association for Computational Linguistics, Stroudsburg, PA, USA (1999)
2. Charniak, E., Altun, Y., de Salvo Braz, R., Garrett, B., Kosmala, M., Moscovich, T., Pang, L., Pyo, C., Sun, Y., Wy, W., Yang, Z., Zeller, S., Zorn, L.: Reading comprehension programs in a statistical-language-processing class. In: ANLP/NAACL Workshop on (2000)
3. Riloff, E., Thelen, M.: A rule-based question answering system for reading comprehension tests. In: Proceedings of the 2000 ANLP/NAACL Workshop on Reading Comprehension Tests as Evaluation for Computer-Based Language Understanding Sytems. ACL, Stroudsburg (2000)
4. Ng, H.T., Teo, L.H., Lai, J., Kwan, J.L.P.: A machine learning approach to answering questions for reading comprehension tests. In: Proceedings of EMNLP/VLC (2000)
5. Hirschman, L., Gaizauskas, R.: Natural language question answering: the view from here. Nat. Lang. Eng. 7(4), 275–300 (2001)
6. Peñas, A., Hovy, E.H., Forner, P., Rodrigo, Á., Sutcliffe, R.F.E., Forascu, C., Sporleder, C.: Overview of qa4mre at clef 2011: Question answering for machine reading evaluation. In: CLEF (Notebook Papers/Labs/Workshop) (2011)
7. Peñas, A., Hovy, E.H., Forner, P., Rodrigo, Á., Sutcliffe, R.F.E., Sporleder, C., Forascu, C., Benajiba, Y., Osenova, P.: Overview of qa4mre at clef 2012: Question answering for machine reading evaluation. In: CLEF (Online Working Notes/Labs/Workshop) (2012)
8. Vicedo, J.L., Ferrandez, A.: Importance of pronominal anaphora resolution in question answering systems. In: Proceedings of the 38th Annual Meeting of the Association for Computational Linguistics, pp. 555–562. ACL (2000)
9. Pakray, P., Bhaskar, P., Banerjee, S., Pal, B.C., Bandyopadhyay, S., Gelbukh, A.F.: A hybrid question answering system based on information retrieval and answer validation. In: CLEF (Notebook Papers/Labs/Workshop) (2011)
10. Bhaskar, P., Pakray, P., Banerjee, S., Banerjee, S., Bandyopadhyay, S., Gelbukh, A.F.: Question answering system for qa4mre@clef 2012. In: CLEF (Online Working Notes/Labs/Workshop) (2012)
11. Clark, P., Harrison, P., Yao, X.: An entailment-based approach to the qa4mre challenge. In: CLEF (Online Working Notes/Labs/Workshop) (2012)
12. Vilariño, D., Pinto, D., Tovar, M., León, S., Castillo, E.: Buap: Lexical and semantic similarity for cross-lingual textual entailment. In: Proceedings of the 6th International Workshop on Semantic Evaluation. ACL, Montréal (2012)

Determining the Degree of Semantic Similarity Using Prototype Vectors*

Mireya Tovar[1,2], David Pinto[2], Azucena Montes[1,3], and Darnes Vilariño[2]

[1] Centro Nacional de Investigación y Desarrollo Tecnológico (CENIDET), Mexico
[2] Faculty Computer Science, Benemérita Universidad Autónoma de Puebla, Mexico
[3] Engineering Institute, Universidad Nacional Autónoma de Mexico
{mtovar,amontes}@cenidet.edu.mx,
{dpinto,darnes}@cs.buap.mx

Abstract. Measuring the degree of semantic similarity for word pairs is very challenging task that has been addressed by the computational linguistics community in the recent years. In this paper, we propose a method for evaluating input word pairs in order to measure the degree of semantic similarity. This unsupervised method uses a prototype vector calculated on the basis of word pair representative vectors which are contructed by using snippets automatically gathered from the world wide web.

The obtained results shown that the approach based on prototype vectors outperforms the results reported in the literature for a particular semantic similarity class.

Keywords: Semantic similarity, hierarchical relationships, prototype vectors.

1 Introduction

With the exponential growth of the information contained in the World Wide Web it arises the need for automating user processes such as searching, information retrieval, question answering, etc. One of the main problems of this automation is that much of the information remains unstructured, i.e., it is written in natural language and its ambiguity is difficult to be automatically processed by machines. The Semantic Web attempts to solve these problems by incorporating semantic to the web data, so that it can be processed directly or indirectly by machines in order to transform it into a data network [1]. For this purpose, it has been proposed to use some knowledge structures such as ontologies for giving semantic and some structure to unstructured data. Among other applications, an ontology is a lexical/semantic resource that allows to perform semantic annotation of web pages contents. Thus, we consider very important to investigate the manner of evaluate the quality of these kind of resources that are continuously been used in the framework of semantic web.

* This work is partialy supported by CONACYT and PROMEP under grants: CONACYT 54371, PROMEP/103.5/12/4962 BUAP-792 and project CONACYT 106625.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 364–373, 2013.

Gruber [2] defines an ontology as: "an explicit specification of a conceptualiation". An ontology includes classes, instances, attributes, relationships, constraints, rules, events and axioms. In many cases, ontologies are structured as hierarchies of concepts modeled either by means of part-whole or class-inclusion semantic relationships. In particular, the class-inclusion relationships are also named is-a, hyponymy or subsumption. There exist, others type of semantic relationships that are not hierarchical such as synonyms, antonyms, etc, however, in this paper, we focus the experiments in the evaluation of semantic hierarchical relationships.

There are two types of nodes in an ontology: concepts and instances [3], but in this paper, we are particularly interested in determining the degree of similarity between a given pair of instances of the ontology that share a semantic hierarchical relationship.

A number of diverse classification methods have been addressed for identifying relationships between concepts or instances [4], [5] and [6]. For instance, for identifying whether or not a given instance (a pair of words *flower*:*tulip*) belongs to a specific relationship (class-inclusion) [7].

Other approaches identify the degree of semantic similarity between a set of word pairs that is known that they belong to a certain class (semantic relationship) [8].

The latter case is the matter of this research work and clearly this problem goes beyond of identifying the membership of an instance in a given class, that is, to detect the variability of the instance with respect to the class.

The remaining of this paper is structured as follows. Section 2 describes with more detail the problem of measuring the degree of semantic similarity for hierarchical relationships. The state of the art is also discussed in Section 2. In Section 3, we present the model proposed for addressing the problem aforementioned. Section 4 show and discuss the results obtained by the presented approach. Finally, in Section 5 the findings and the future work are given.

2 Degree of Semantic Similarity

Even if there exist a number of widely used semantic relationships (see Table 1), in this paper we consider only two classes: Class-Inclusion and Part-Whole for determining the degree of semantic similarity between two instances of an ontology.

The degree of semantic similarity involves the process of determining a ranking for pairs that belong to the same semantic class. For instance, let us consider the following word pairs: $\{dog : bark\}$, $\{cat : meow\}$ and $\{floor : squeak\}$ that share the ENTITY:SOUND semantic relationship. The intuition is that the first pair is more similar to the second than with the third one. In this sense, it is very important to construct formulae that allows to rank word pairs sharing the same semantic similarity. In this paper, we analize techniques for obtaining such ranking by means of prototype vectors. For the experiments carried out, we have considered only two semantic relationships: class-inclusion and part-whole. A description of these two relationships follows.

Table 1. High-level classes and instances examples that belong to some subclass [9]

High-level class	Subclass	Instances examples
1 **CLASS-INCLUSION**	Taxonomic	flower:tulip
2 **PART-WHOLE**	Object:Component	car:engine
3 SIMILAR	Synonymity	car:auto
4 CONTRAST	Contradictory	alive:dead
5 ATTRIBUTE	Item:Attribute	beggar:poor
6 NON-ATTRIBUTE	Item:Nonattribute	harmony:discordant
7 CASE RELATIONS	Agent:Recipient	doctor:patient
8 CAUSE-PURPOSE	Cause:Effect	joke:laughter
9 SPACE-TIME	Location:Action/Activity	school:learn
10 REFERENCE	Sign:Significant	siren:danger

The Class-Inclusion relationship defines a parent-child relationship (taxonomy, e.g. flower: tulip), whereas part-whole relationship divides a concept as a whole in different parts (e.g engine:car). Both relationships, as mentioned before, are considered to be hierarchical [10], [11], [12]. The subclasses of the Class-Inclusion and the Part-whole classes are shown in Table 2 and Table 3, respectively. Syntactic patterns for each subclass, together with instance examples are also presented in these Tables.

Table 2. The **Class-Inclusion** class: subclasses, syntactical patterns and examples of instance pairs that belong to each subclass

	Subclass	**Syntactical patterns**	**Examples of instances** $(Y : X)$
1b	Functional	Y functions as an X	ornament:brooch, weapon:knife, vehicle:car
1c	Singular Collective	a Y is one item in a collection/group of X	cutlery:spoon, clothing:shirt, vermin:rat, medicine:asprin
1d	Plural Collective	Y are items in a collection/group of X	groceries:eggs, refreshments:sandwiches, drugs:amphetamines
1e	ClassIndividual	Y is a specific X	queen:Elizabeth, river:Nile, city:Berlin

One of the most important forums that have tackled the problem of identifying the degree of semantic similarity is Semeval[1]. There have been some teams that presented different approaches for this particular problem. Thus, the state of the art is scarce but it follows.

In [13] it was proposed two systems that tackled the problem. Their methodology employed lexical patterns generated from the contexts in which the word pairs occurs. They constructed patterns using the example word pairs for each subclass (see the third column of Table 2). They used a corpus with 8.4 million

[1] `http://www.cs.york.ac.uk/semeval-2012/`

Table 3. The **Part-Whole** class: subclasses, syntactical patterns and examples of instance pairs that belong to each subclass

Subclass	**Syntactical patterns**	**Examples of instances** $(X : Y)$
Object:Component	a Y is a part of an X	car:engine, face:nose, novel:epilogue, turtle:carapace
Collection:Member	X is made from a collection of Y	forest:tree, anthology:poem, fleet:ship, medley:melodies
Event:Feature	Y is typically found at an event such as X	rodeo:cowboy, banquet:food, wedding:bride
Activity:Stage	X is one step/action/part of the actions in Y	shopping:buying, planting:gardening, kick-off:football
Item:Topological Part	Y is one of the areas/locations of X	room:corner, mountain:foot, table:top
Object:Stuff	X is made of / is comprised of Y	glacier:ice, salt:sodium, lens:glass, parquet:wood
Item:Distinctive Nonpart	X is devoid of / cannot have Y	tundra:tree, horse:wings, perfection:fault, society:pariah
Item:Ex-part/Ex-possession	an X once had/owned/possessed Y but no longer	apostate:belief, wood:splinter, prisoner:freedom, metal:dross

documents from Gigaword and 4 million articles from Wikipedia. They ranked word pairs using a model predicting the probability that they belong to the input relationship. They proposed two approaches for ranking the word pairs of the subclasses: The **UTD-NB** approach used a probabilistic model, whereas **UTD-SVM** employed a SVM-rank model to rank the word pairs. Their performance was interesting, achieving good results reported in the Semeval conference [8].

In [14] it was proposed three unsupervised approaches that used the Gloss Vector measure found in the package WordNet::Similarity. The author expanded the vector of glosses by using the relationships associated to each word pair. Additionally, superglosses have been produced. The cosine measure was employed for ranking the results obtained for each pair of words. The corpus used was the complete collection of glosses and examples from WordNet 3.0, i. e., 118,000 glosses.

In [15] a supervised approach has been proposed. They used lexical, semantic, WordNet-based and contextual features. In order to rank the obtained results, the cosine measure was employed. Moreover, they used a very restricted corpus for testing their approach in comparison with the other teams of the competition.

The approach presented in this paper is described in the following section.

3 A Prototype-Based Model

In order to calculate a ranking for a set of word pairs that belong to the same semantic relationship, we have constructed a prototype vector for this relationship. This prototype is calculated as the average value among all the representative vectors for each word pair. We assume that the prototype vector (also known as the class centroid) has enough quality for representing such semantic relationship. Thus, the final ranking is calculated by means of the distance that exist between each word pair representative vector and the prototype vector.

The representative vectors for each word pair are calculated in the basis of information gathered from the Web by using a search engine[2]. Thus, we obtain snippets that contain the two words of the given pair. All the snippets for each word pair are analized for constructing a representative vector using the TF-IDF weighting schema. In Figure 1 we show the unsupervised approximation for determining the degree of instance prototypicality within a given subclass. As we mentioned before, we only considered two classes (class-inclusion and part-whole), and 12 ($n = 12$) subclasses, as shown in Tables 2 and 3.

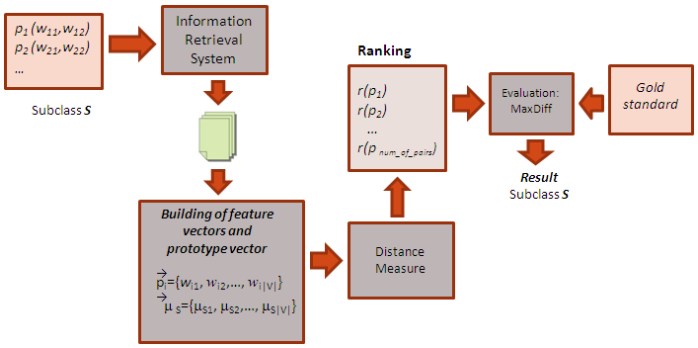


Fig. 1. The proposed architecture for calculating the degree of semantic similarity

The architecture proposed presents the following phases:

- *Gathering the corpus.* The corpus consists of short texts (snippets) containing the keywords of the instances (word pairs). We used the Google API for gathering such snippets from the world wide web.
- *Building the feature vectors.* Using the vocabulary of the corpus, we built the feature vectors for each subclass. Normally, each subclass contain between 27 to 31 instance pairs, and the feature vector contains frecuency-based lexical features of the terms associated the instance pairs. Let V the vocabulary of the whole corpus, and $\overrightarrow{p_i}$ the representative vector for the word pair p_i. Thus, $\overrightarrow{p_i} = \{w_{i,1}, w_{i,2}, \cdots, w_{i,|V|}\}$, where $w_{i,k}$ is the weight of the k-th element of the vocabulary in the $\overrightarrow{p_i}$ representative vector of the word pair p_i.

[2] In this case we used Google.com

This weight is calculated using two formulae: term frequencies (TF) and the combination of term frequency and inverse document frequency (TF-IDF).

– *Constructing the prototype vector.* The prototype vector is constructed by considering the mean value among all the feature vectors for all the possible word pairs in a given subclass. Thus, this prototype vector may be seen as the centroid of the subclass. Formally, for each $p_i \in S$ with $i = \{1, \cdots, num\_of\_pairs\}$, where S is the current subclass, the centroid μ_S of this subclass is calculated as shown in Eq.1.

$$\overrightarrow{\mu_S} = \{\mu_{S,1}, \mu_{S,2}, \cdots, \mu_{S,|V|}\}$$
$$\text{with } \mu_{S,k} = \frac{1}{num\_of\_pairs} \sum_{i=1}^{num\_of\_pairs} w_{i,k} \quad (1)$$

– *Calculating the ranking* We apply different distance measures between the feature vectors and their corresponding prototype vector for determining a score that indicates the degree of representativeness of the instance pairs with respect to the whole subclass. Thus, the closer a representive vector to the centroid, the higher the ranking of representativeness. In other words, we assume that those word pairs that are closer to the subclass prototype vector are more representative for this subclass.
In order to measure the distance among the representative vectors and their corresponding prototype vector, we employed two classical distance measures: Euclidean (Equation 2) and Manhattan (Equation 3).

$$d(\overrightarrow{p_i}, \overrightarrow{\mu_S}) = \frac{\|\overrightarrow{p_i} - \overrightarrow{\mu_S}\|}{|V|} \quad (2)$$

$$d(\overrightarrow{p_i}, \overrightarrow{\mu_S}) = \frac{\sum_{k=1}^{|V|} |p_{i,k} - \mu_{S,k}|}{|V|} \quad (3)$$

– *Evaluating the ranking.* In order to determine the performance of the presented approach, we have used the MaxDiff evaluation measure. MaxDiff is an analitycal techique that indicate the preference that a respondent has for a set of alternatives [16], [17]. This measure provides an average evaluation result, when evaluating the score of each pair of instances of the subclass with respect to a given gold standard. This reference of evaluation (gold standard) is built as follows. The respondents are asked to evaluate four or five pairs of words in an specified subclass and later they choose the best and worst pair of instance prototipical of the relationship [8].

4 Experimental Results

In this section we present the results obtained with the proposed approach. These results are compared with those reported in [8]. We have reported only the 12 subclasses that we have analized in this paper. We first describe the dataset used in the experiments, and later the obtained results are presented and discussed.

4.1 Dataset

Tables 4 and 5 present the number of word pairs to be evaluated for each subclass, the number of possible swaps (reversals), and examples of word pairs. The reversals were added to the subclass with the aim of determining how efficient are the approaches that consider these reversals pairs. In particular, with relationships that are completely directional such as Part-Whole, but also for other type of simetric classes, such as synonymy.

An example of a reversal word pair could be *dogs:animals* and *animals:dogs* of the Part-Whole class and the Plural Collective subclass. The ranking for each one should be different because the relationship is valid in only one direction. In this example, the correct word pair should be *animals:dogs*, because the representative pattern "Y are items in a collection/group of X" fulfills the relationship semantics when the pattern is used.

Table 4. The **Class-Inclution** class: subclasses, number of pairs and examples of instance pairs that belong to each subclass to evaluate

Subclass	Number of pairs	Reversals	Example
Functional	41	6	instrument:clarinet, fuel:gasoline, seat:stool, seat:chair, lubricant:oil, home:tree
Singular Collective	42	5	internet:website, book:novel, beverage:water,fruit:apple, art:sculpture
Plural Collective	43	5	birds:crows, colors:blue, silverware:spoon, dogs:animals animals dogs
ClassIndividua	33	2	horse:Palomino, snake:Cobra, Earth:planet, king:Arthur, university:Yale

4.2 Results

We gathered approximately 264,000 snippets from Internet in order to made up the reference corpus for the 12 subclasses to be evaluated. Table 6 and 7 show the average performance obtained by the four different runs executed in the experiments using the Class-Inclusion and the Part-Whole classes, respectively. The other results correspond to those reported in the literature. In particular, the run named **Euclidian-TF** uses **TF** as a weighting schema, employing the Euclidian distance measure for ranking the degree of semantic similarity. **Euclidian-TF-IDF** uses **TF-IDF** as a weighting schema in combination with the Euclidian distance measure. The runs named **Manhattan-TF** and **Manhattan-TF-IDF** are similar but they use the Manhattan distance measure instead of the Euclidian one.

From the obtained results we can observe that there is not a significative difference between the two distance measures employed. Actually, it can be seen that the Euclidean distance generally improves the Manhattan one. This means that the snippets that correspond to a given word pair that were gathered from

Table 5. The **Part-Whole** class: subclasses, number of pairs and examples of instance pairs that belong to each subclass to evaluate

Subclass	Number of pairs	Reversals	Example
Object:Component	44	5	house:room, recipe:ingredient, fin:fish, motor:boat, eye:lashes
Collection:Member	38	4	flock:sheep, album:songs, paragraph:word, herd:antelopes
Event:Feature	39	3	funeral:coffin, church:preacher, baptism:priest, competition:athlete
Activity:Stage	40	5	soaping:showering, stitching:sewing, purling:knitting, tennis:volleying
Item:Topological Part	43	4	tree:root, river:bed, coast:east, bush:roots
Object:Stuff	43	5	boots:leather, lawn:grass, box:cardboard, sock:thread
Item:Distinctive Nonpart	39	4	forest:sand, hearing:deaf, pride:embarrassment, venus:life
Item:Ex-part/Ex-possession	42	4	corpse:life, repair:break, notebook:paper, widow:husband

Table 6. Results for CLASS-INCLUSION Class

Approaches	MaxDiff
Euclidian-TF	**39.15**
Manhattan-TF	**38.85**
UTD-NB	37.60
UMD-V1	35.60
UMD-V2	33.13
UTD-SVM	31.58
Euclidian-TF-IDF	**31.55**
BUAP	31.43
Random	30.98
Manhattan-TF-IDF	**30.60**
UMD-V0	29.23

the Internet are quite similar because the representative vectors are close to the prototype vector (it is a dense group of representative vectors). This behaviour is valid for approximately 39% of the word pairs (for the Class-Inclusion class), whereas this percentage was approximately 30% for the Part-Whole class. The rest of word pairs need to be tuned in order to retrieve more representative snippets, or a new document collection which may be used for the same purpose.

It is remarkable that the **TF** representation schema obtained a very good performance with the Class-Inclusion class. We consider that this behavior is due to the unsupervised system proposed do not consider symmetric vs asymmetric relationships (reversal word pairs). The proposed system neither consider

Table 7. Results for the PART-WHOLE Class

Approaches	MaxDiff
UTD-NB	40.89
UTD-SVM	35.65
BUAP	35.05
Euclidian-TF-IDF	**33.30**
Manhattan-TF-IDF	**32.13**
Random	31.86
Manhattan-TF	**30.45**
Euclidian-TF	**29.46**
UMD-V0	29.40
UMD-V2	28.55
UMD-V1	26.51

the order of the words in the relationship, nor the direction of the semantic relationship. However, both Class-Inclusion and Part-Whole are asymmetric relationships and the approach presented is highly sensible to reversals, therefore, the performance decreases. This consideration is also avoided in the information retrieval system leading to obtain snippets that may not represent adequately the semantic relationship. Further investigation should integrate the concept of symmetric vs asymmetric relationships into the methodology proposed.

In summary, the results obtained outperformed all the results reported in the literature for the Class-Inclusion class, whereas in the class Part-Whole, we have slightly improved the random baseline. Further experiments will allow to analize the manner we may improve these results.

5 Conclusions

The schema proposed for measuring the degree of similarity based on prototype vectors performed well for the Class-Inclusion class. A simple weighting measure such as term frequencies allowed to capture the neccesary features for representing the word pairs that share a semantic relationship. On the other hand, we did not succeed representing adequately for the Part-Whole class, an issued that must be further investigated.

From the results obtained, we were able to observe that the snippets retrieved have an acceptable quality, but they may also be improved by adding other lexical resources or document collections. As future work, we would like to analize more into detail the outliers in order to remove them for generating a much better reference corpus.

References

1. Solís, S.: La Web Semántica. Lulu Enterprises Incorporated (2007)
2. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. Technical Report KSL-93-04, Knowledge Systems Laboratory, USA (1993)

3. Ruiz-Casado, M., Alfonseca, E., Castells, P.: Automatic extraction of semantic relationships for wordnet by means of pattern learning from wikipedia. In: Montoyo, A., Muńoz, R., Métais, E. (eds.) NLDB 2005. LNCS, vol. 3513, pp. 67–79. Springer, Heidelberg (2005)
4. Celli, F.: Unitn: Part-of-speech counting in relation extraction. In: Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval 2010, pp. 198–201. Association for Computational Linguistics, Stroudsburg (2010)
5. Chen, Y., Lan, M., Su, J., Zhou, Z.M., Xu, Y.: Ecnu: Effective semantic relations classification without complicated features or multiple external corpora. In: Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval 2010, pp. 226–229. Association for Computational Linguistics, Stroudsburg (2010)
6. Negri, M., Kouylekov, M.: Fbk_nk: A wordnetbased system for multiway classification of semantic relations. In: Proceedings of the 5th International Workshop on Semantic Evaluation, SemEval 2010, pp. 202–205. Association for Computational Linguistics, Stroudsburg (2010)
7. Turney, P.D.: Similarity of semantic relations. Computational Linguistics 32, 379–416 (2006)
8. Jurgens, D.A., Mohammad, S.M., Turney, P.D., Holyoak, K.J.: Semeval-2012 task 2: Measuring degrees of relational similarity. In: Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval 2012, Montreal, Canada (2012)
9. Bejar, I., Chaffin, R., Embretson, S.: Cognitive and psychometric analysis of analogical problem solving. Recent Research in Psychology. Springer (1991)
10. Lim, E.: Intelligent Systems Reference Library: Knowledge Seeker: Ontology Modelling for Information Search and Management - A Compendium. Intelligent Systems Reference Library. Springer (2011)
11. Harpring, P.: Introduction to Controlled Vocabularies: Terminology for Art, Architecture, and Other Cultural Works. Introduction To Series. Getty Research Institute (2010)
12. Organization, N. (U.S.), N.I.S.O.: Guidelines for the Construction, Format, and Management of Monolingual Controlled Vocabularies. National information standards series. NISO Press (2005)
13. Rink, B., Harabagiu, S.: Utd: Determining relational similarity using lexical patterns. In: Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval 2012, June 7-8, pp. 413–418. Association for Computational Linguistics, Montréal (2012)
14. Pedersen, T.: Duluth: Measuring degrees of relational similarity with the gloss vector measure of semantic relatedness. In: Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval 2012, June 7-8, pp. 497–501. Association for Computational Linguistics, Montréal (2012)
15. Tovar, M., Reyes, J.A., Montes, A., Vilariño, D., Pinto, D., León, S.: Buap: A first approximation to relational similarity measuring. In: Proceedings of the 6th International Workshop on Semantic Evaluation, SemEval 2012, June 7-8, pp. 502–505. Association for Computational Linguistics, Montréal (2012)
16. Karash, K., Board, N.R.C.U.T.R., Program, T.C.R., Administration, U.S.F.T., Corporation, T.D.: Understanding How Individuals Make Travel and Location Decisions: Implications for Public Transportation. Report (Transit Cooperative Research Program). Transportation Research Board (2008)
17. Orme, B.: Maxdiff analysis: Simple counting, individual-level logit, and hb. Technical report, Sawtooth Software. Research Paper Series (2009)

Single Extractive Text Summarization Based on a Genetic Algorithm

René Arnulfo García-Hernández and Yulia Ledeneva

Autonomous University of the State of Mexico,
Santiago Tianguistenco, México
renearnulfo@hotmail.com, yledeneva@yahoo.com

Abstract. Extractive text summarization consists in selecting the most important units (normally sentences) from the original text, but it must be done as closer as humans do. Several interesting automatic approaches are proposed for this task, but some of them are focused on getting a better result rather than giving some assumptions about what humans use when producing a summary. In this research, not only the competitive results are obtained but also some assumptions are given about what humans tried to represent in a summary. To reach this objective a genetic algorithm is proposed with special emphasis on the fitness function which permits to contribute with some conclusions.

1 Introduction

According to Lee [1], the amount of information in Internet continues growing, but much of this information is redundant. Therefore, we need new technologies to efficiently process information. The automatic generation of document summaries is a key technology to overcome this obstacle. Given this, it is essential to develop automated methods that extract the most relevant information from a text, researched by Automatic Text Summarization (ATS) area [2], [3], [4], [5]. ATS is an active research area that deals with single- and multi-document summarization tasks. In single-document summarization, the summary of only one document is built, while in multi-document summarization the summary of a whole collection of documents (such as all today's news or all search results for a query) is built. While we believe that our ideas apply to both cases, in this work we have experimented only with single-document summaries.

Summarization methods can be classified into abstractive and extractive summarization. An abstractive summary is an arbitrary text that describes the contexts of the source document. Abstractive summarization process consists of "understanding" the original text and "re-telling" it in fewer words. Namely, an abstractive summarization method uses linguistic methods to examine and interpret the text, and then to find new concepts and expressions to best describe it by generating a new shorter text that conveys the most important information from the original document. While this may seem the best way to construct a summary (and this is how human beings do it), in real-life setting immaturity of the corresponding linguistic technology for text analysis and generation currently renders such methods practically infeasible.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2013, LNCS 7914, pp. 374–383, 2013.

An extractive summary, in contrast, is composed with a selection of sentences (or phrases, paragraphs, etc.) from the original text, usually presented to the user in the same order—*i.e.*, a copy of the source text with most sentences omitted. An extractive summarization method only decides, for each sentence, whether or not it will be included in the summary. The resulting summary reads rather awkward; however, simplicity of the underlying statistical techniques makes extractive summarization an attractive, robust, language-independent alternative to more "intelligent" abstractive methods. In this paper, we consider single extractive summarization.

The main problem for generating an extractive automatic text summary is to detect the most relevant information in the source document. Although, some approaches claim being domain and language independent, they use some degree of language knowledge like lexical information [5], key-phrases or a golden sample for supervised learning approaches [6] [7]. Furthermore, training on a specific domain tends to customize the extraction process to that domain, so the resulting classifier is not necessarily portable. For that reason, these works present a high dominion and language dependence degree.

A typical extractive summarization method [8] [9] consists in 5 steps: preprocessing, term selection, term weighting, sentence weighting and sentence selection; at each of them different options can be chosen. We will assume that the units of selection are sentences (these could be, say, phrases or paragraphs). Thus, final goal of the extractive summarization process is sentence selection.

Usually, in the preprocessing step the document is analyzed for removing words without meaning (stop words) and for getting a canonical representation of each word by applying a stemming algorithm in order to find relations between significant words. Moreover, some methods use more complex resources such as *Part-of-speech tagging*, lemmatization (instead of stemming), key words, key phrases, etc.

Most of the language-independent methods employing the n-gram as the unit in term selection step which is composed by all the sequences of n words of the document. Recently, the Maximal Frequent Sequence (MFS) model has been proposed as text model [8] [10] [11] [12] which tried to select only the important terms according to the frequency without the need of determine n. A MFS text model can be defined in terms of grams as all the frequent grams (of any size) that are not subsequence of other frequent grams. For considering that a gram is frequent it must be repeated at least a threshold times in the text, when the threshold it not specified it is assumed that is taken the lowest possible, *i.e.* two.

In third step is given an importance to the selected terms, for example the presence or absence of a term can be used as Boolean weighting, but in this weighting it is not possible to know which term is more important. An alternative is to use the frequency of the term as TF weighting, but a very frequent term is not always important since could be a stop word or a term that it is repeated in most of the sentences; therefore it is important for the entire document and not for a single sentence. This problem can be solving if the inverse document frequency is used as IDF weighting, in this case the frequency of a term is divided by the number of sentences where the term is presented; it means a frequent term is more important if it appears in a single sentence instead of all the sentences.

Normally, for composing the summary the sentences are selected according to its relevance in the sentence selection step. This way of sentence selection tends to produce redundant summaries. In this research, special attention is given to sentence selection step since this process must consider all the relevant information getting in previous steps. In this paper, a genetic algorithm is proposed for optimizing the sentences selection step based on the frequency of the words (1-grams).

2 Related Works

In different ways, several approaches have employed a genetic algorithm for the ATS task based on attribute selection [13] [14]. However, these kinds of approaches have in common that represent each sentence as a set of attributes extracted from the original text. The following features are gotten using only static and structural information from the original text, without linguistic knowledge.

Similarity to title [13] [14] [15] is a measure that arise sentences that have common words with the title. This is determined by counting the number of matches between the words in a sentence and the words in the title [13] or it is calculated as the cosine similarity [14].

Similarity to keywords [14] is an analogous measure to similarity to title.

Sentence length feature [13] [14] [15] gives more preference for longer sentences, under the idea that short sentences could bring, for example: datelines, numbers or author names. This measure is normalized to the longest sentence in the document.

Term weight feature [13] is based on the frequency of the terms presented in the sentence. The score of a sentence can be calculated as the sum of the weights of the terms in the sentence. A term will be more important if it appears frequently into the document but simultaneously it does not appear in others sentences.

Sentences position feature [13, 14, 16] relies in the baseline heuristic [17] that establishes the first sentences of a text can be considered as a good summary. Document collections created specifically for ATS systems has proved that it is a hard line to overcome. Normally, this feature assigns the inverse order number as the importance for the sentence, for example, if there are 10 sentences in the document, the first sentence has a score of 10/10, and the second one has a score of 9/10 and so on.

Sentence similarity feature [13] measures the similarity that has a sentence against the rest of sentences in order to avoid getting untypical sentences. Therefore, a sentence with high score is more probable to appear in the summary. One option to get the similarity between two sentences is to use the cosine similarity measure.

Numerical feature [13] is based on the idea that in the sentences where numerical data appears are more relevant. For measure this feature is calculated as the ratio of the number of the numerical data in the sentences over the sentence length.

It is possible to extract other dependent-linguistic features based on *Proper Noun*, [13] [14] *Thematic Word* [13], *Anaphors* [14], *Discourse Markers* [14].

Some approaches that using a genetic algorithms for the ATS task [13] [14] are based on attribute sentence selection in a supervised classification scheme [13], thus, for these approaches is needed to account with a previously set of golden summaries for training. Other approaches [15] use the GA in an unsupervised classification scheme, where the fitness function is formulated with some of the above features for evaluating the summarized sentences.

3 Proposed Genetic Algorithm

Genetic Algorithm (GA) is the most traditional evolutionary technique that has proved to be an alternative solution for an optimization problem. In the first step, the GA proposes a population of random solutions (*initial population step*) that are evaluated according to the objective function to optimize (*fitness function step*). In this sense, a solution for one problem is not absolute, it means, there is set of possible solutions where some are better than others. Considering mostly the best solutions (*parents selection step*), the GA proposes a new population mixing (*crossover step*) some parts from a canonical codification (*Chromosome encoding step*) of these good solutions in order to get better solutions (*evolution principle*). Eventually, the way of mixing some parts from the canonical codification could produce repeated solutions. Therefore, the GA applies a small variation (*mutation step*) to the canonical codification in the new population in order to explore new solutions. The new population is evaluated again and the process is repeated until a satisfactory solution is reached or until some arbitrary stop-criteria is reached (*stop condition*).

3.1 Proposed Genetic Operators

Preprocessing. Before the original text could be used for the GA, it is needed to adapt the entry of the original text to the format of the GA. In this step, the original text is separated in sentences. Also, the text is preprocessed with the well-known Porter Stemmer [18] in order to find related words. Since the proposed method is based on the frequency of the words as a measure of its relevance (section 4.3), this does not take into account the frequency of stop words because it is higher than meaningful words.

Chromosome Encoding. GA must encode each solution (chromosome) using a canonical way. One of the most used encodes for a chromosome is the binary representation. For the ATS problem we propose to represent the genes of a chromosome (C) with a vector of length n of binary values (C_n), where the C_i gene corresponds with the i-th sentence in the original text. If C_i gene has a value of 1 ($C_i = 1$) means that the i-th sentence is included in the summary, otherwise not.

Initial Population. After the chromosome encoding is setup, it is possible to create the first generation considering some parameters. Each gene can take a binary random value ($C_{i=1\ldots n} = Random[0,1]$). However, if a sentence is selected to appear in the summary ($C_i = 1$) then the number of words of the i-th sentence are summed to the number of words in the summary. The number of words in the summary must contain at least the number of words specified by the user (m). To guarantee that each sentence could be selected for the summary, there are created n number of chromosomes in the initial population and in each one a different gene is arbitrary set to 1.

$$Population = \left\{C_i^j \middle| j = 1 \ldots n, i = 1 \ldots n, C_{i=j}^j = 1, C_{i\neq j}^j = Random[0,1]\right\}$$

Fitness Function. One of the key steps of a genetic algorithm is the Fitness Function which in this case it is based on the idea of f-measure that it is a harmonic balance of recall and precision measures. Usually in information retrieval, precision is defined as

the number of correctly recovered units divided by the number of recovered units; and recall is defined as the number of correctly recovered units divided by the number of correctly units. In this way, precision measures the fraction of retrieved units that are relevant, while recall measures the fraction of relevant instances that are retrieved. However, for generating a summary (S), the maximum-words threshold (m) of a summary is considered. Consequently, the number of recovery units always is limited by the maximum-word threshold. Therefore the golden summary must have, for one side, the most relevant words of the original text (T) and, for the other side, must have expressivity, it means, it must not be redundant.

The relevance of a word w is represented by the appearing frequency of the word in the original text ($frequency(w,T)$), and the expressivity is represented if only are considered the different words that the summary can have ($\{\text{word} \in S\}$). In this sense, the best summary would contain the most frequent words with respect to the original text and each word must be different. In order to have a normalized measure the sum of the frequencies of the different words in the summary is divided by the sum of the frequencies of the most frequent words with respect to the original text:

$$\beta = \frac{\sum_{\text{p}=\{\text{word} \in S\}}^{m} frequency(p,T)}{\sum_{\text{q}=\{\text{word} \in T\}}^{m} frequency(q,T)}$$

Sentence position feature is a heuristic that has proved that the first sentences from the original text are good candidates of being part for the summary. Normally, the inverse position order of the sentence it is used as a measure of its relevance. The problem of measuring this feature in this way is that, for example, with a 30-sentence text, the first sentence will be 30 times more important that the last one. It makes almost impossible that the last sentence could appear in the summary. In contrast to [13] [14], we propose to make this difference softer using the linear equation with slope t, if t is -1 we will measure the sentence position as in [13] [14], and if t is 0, it will give the same relevance to each sentence. For a text with n sentences, if the sentence i was selected for the summary (it is, the chromosome$|C_i| = 1$) then its relevance is defined as: $t(i - x) + x$, where $x = 1 + (n-1)/2$ and t is the slope for discovering. In order to normalize the sentence position measure (δ), it is calculated the relevance of the first k sentences, where k is the number of selected sentences.

$$\delta = \frac{\sum_{|C_i|=1}^{n} t(i-x)+x}{\sum_{j=1}^{k} t(j-x)+x}, \quad x = 1 + \frac{(n-1)}{2}$$

Therefore the fitness function is: $fitness = \beta \times \delta$

Parent Selection. In this point, each chromosome must have associated a fitness value that will let to mostly select the best chromosomes. The evolution principle establishes that normally if two good solutions are crossing it could produce better solutions; nevertheless, in some cases the solution could be worse. In this step, we employ the classical roulette selection that gives more probability of being selected as

a parent, to the chromosomes that have a greater fitness value. In this way, the worst chromosome has the possibility of being selected, although it was slight probable.

Crossover. Classical crossover operators as *n-point crossover* does not work properly because the new child chromosome could represent a summary with more or less words than the user specified. Therefore, to create the new chromosome we propose to choose from both parents the genes randomly, but consider only those with value 1. In this way, if the C_i gene has a value of 1 in both parents, it has more probability of being selected for the child chromosome. Each time a gene in the child chromosome is selected the minimum number of words for the summary is reviewed.

Mutation. According to the evolution scheme, the mutation slightly happen in the nature with a low probability of 0.1 percentages, however is one of the fundamental mechanisms to preserve the evolution. The classical operator *inverse mutation operator* inverts the binary value of a randomly selected gene. In our proposed scheme, this operator will produce summaries with more or less words than the user specified. In this step we propose to apply the invert operator twice to the child chromosome, but the first time only the genes with value 1 are considering for invert the value; in the second time only the genes with value 0 are considering for invert the value. After that, the number of words in the summary is review it, if the numbers of words do not have the number of words specified by the user, another gene with value 0 is inverted, this process continues until the number of words specified by the user is satisfied.

4 Experimentation

We used the standard DUC 2002 document collection provided [19]. In particular, we used the data set of 567 news articles of different length and with different topics. Each document in the DUC collection is supplied with a set of human-generated summaries provided by two different experts[1]. While each expert was asked to generate summaries of different length, we used only the 100-word variants.

Evaluation Procedure. We used the ROUGE evaluation toolkit [20] which was found to highly correlate with human judgments [21]. It compares the summaries generated by the program with the human-generated (gold standard) summaries. For comparison, it uses n-gram statistics. Our evaluation was done using n-gram (1, 1) setting of ROUGE, which was found to have the highest correlation with human judgments, namely, at a confidence level of 95%. ROUGE lets to know the f-measure that is a balance (not an average) of recall and precision results.

Table 1 shows the ROUGE evaluation from our approach with the whole DUC-2002 collection; varying the slope from -0.25 to -0.75. In figure 1, it is possible to observe that our approach has the best f-measure when the slope is -0.625.

[1] While the experts were supposed to provide extractive summaries, we observed that the summaries provided in the collection were not strictly extractive: the experts considerably changed the sentences as compared with the original text.

Table 1. Results of our proposed approach varying slope from -0.25 to -0.75

Slope	Recall	Precision	F-measure
-0.25	0.4737	0.4686	0.4710
-0.375	0.4736	0.4736	0.4760
-0.5	0.480	0.4747	0.4772
-0.55	0.4829	0.4775	0.4801
-0.625	**0.4855**	**0.4802**	**0.4827**
-0.7	0.4843	0.4789	0.4815
-0.75	0.4790	0.4735	0.4761

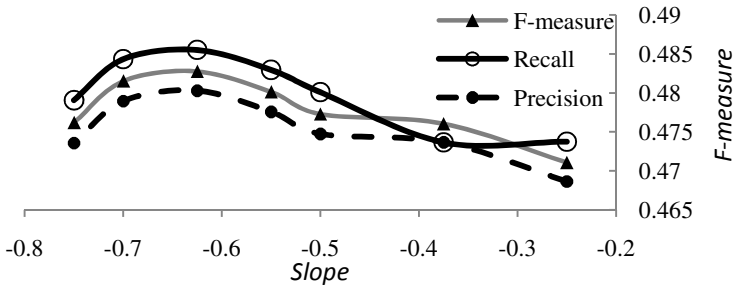


Fig. 1. Behavior of the performance when the slope is varying

5 Comparison with Related Works

In table 2, our proposed approach is compared to others approaches that have used the same DUC-2002 document collection for text summarization.

- **Baseline** (random) [3]: This is a heuristic in which the summaries are built from a set of sentences selected in random way. This simple strategy has the purpose of determine how significant the results can be achieved.
- **TextRank** [17]**:** The approach is a ranking algorithm based on graphs. A graph is built to represent the text, so that the nodes are words (or other text entities) interconnected by vertices with meaningful relationships. For the task of extracting sentences, the goal is to qualify whole sentences and sort highest to lowest rating. Therefore, a vertex is added to the graph for each sentence in the text. To establish connections (cycles) between sentences, define a relationship of similarity, where the relationship between two sentences can be seen as a process of "recommendation": a sentence that points to some concept in the text gives the reader a "recommendation" to refer to other sentences in the text that point to the same concepts and therefore a link can be established between any two sentences that share a common content.
- **Maximal Frequent Sequences (MFSs)** [3] [8] [9]. Ledeneva *et al.* [3] [8] [9] experimentally shows that the words which are parts of bigrams (2-word sequences) which are repeated more than once in the text are good terms to describe the content of that text, so also called the maximal frequent sequences

(sequences of words that are repeated a number of times and also are not contained in other frequent sequences). This work also shows that the frequency of the term as ranking of terms gives good results (while only count the occurrences of a term in repeated bigrams).

- **Baseline** (first): This heuristic selects the first sentences of the document until the desired size of the summary is reached [9]. Besides of being a simple heuristic, only four DUC-2002 systems (S1,S2,S3,S4) could outperform the baseline results (showed in table 2).
- **K-means:** The k-means algorithm creates clusters of similar objects. In [3] the k-means is used for creating clusters of sentences from the original text that allow identifying the main ideas; after that, from each cluster the most representative sentence is selected for the summary.
- **Topline [6].** In this work, a GA was used to calculate the best summaries that it is possible to find according with the ROUGE evaluation.

The comparison of the best F-measure results of our proposed approach with the above state-of-the-art approaches is presented in table 2. Since, any method can be worse than choosing random sentences (baseline: random) the significance of f-measure is recalculated as 0%. In opposite way, since any method can outperform the Topline is considered as 100%. Using baseline and topline is possible to recalculate the f-measure results in order to see how significant the results are (see table 2).

Table 2. Results of f-measure with other methods

System	F-measure	Significance
Baseline: random	0.3881	0%
TextRank:	0.4432	26.50%
MFS's (k-best)	0.4529	31.16%
Baseline: first	0.4599	34.53%
GA	0.4662	37.56%
S1	0.4683	38.57%
S2	0.4703	39.53%
TextRank	0.4708	39.77%
S3	0.4715	40.11%
MFS's (1best+first)	0.4739	41.26%
K-means	0.4757	42.13%
MFS's-EM-5	0.4774	42.95%
S4	0.4814	44.87%
Proposed GA	**0.4827**	**45.50%**
Topline [6]	0.596	100%

6 Conclusions

We have proposed a genetic algorithm for automatic single extractive text summarization task. Specifically, we proposed the preprocessing, chromosome

encoding, initial population, fitness function, parent selection, crossover and mutation step. Our genetic algorithm allow to consider the number of words that a user desire. All the parameters that the GA could need are calculated automatically considering the structure of the original text (in fact, it was applied to 567 documents of the DUC collection). In this sense, from the original text was possible determine the number of chromosomes in the population and the number of maximum iterations.

In contrast to the state-of-the-art works related to GA, the proposed GA is not based in a database that was built from features whose were extracted from sentences. Instead, our GA evaluates how good the summary is with respect to the original text, without the necessity of having a collection for training a classifier. In these sense, fitness function tell us more what a summary must contain instead of what process humans follows for building a summary.

Furthermore, we found that if there were a linear relevance with respect to sentence position in the original text, it is of 0.625 considering two consecutive sentences. This parameter was calculated for the DUC-2002 collection. As a future work, other collections will be tested with this parameter.

There are different terms that can be chosen as words, n-grams or MFS; we use words that are easier for extracting for the original text. There are other features that can be extracted from the sentences as similarity to title, sentence length, etc.; the proposed approach uses only the frequency of words and the sentence position. Also, it is important to note that our purposed approach works independently from linguistic resources. We think that this research is relevant since employing basic language-independent information from the original text, it was possible to outperform the others approaches that use the same collection.

References

[1] Lee, J.-H., Park, S., Ahn, C.-M., Kim, D.: Automatic Generic Document Summarization Based on Non-negative Matrix Factorization. Information Processing and Management 45, 20–34 (2009)

[2] Luhn, H.P.: The automatic creation of Literature abstracts. IBM Journal of Research and Development (1958)

[3] Garcia-Hernandez, R.A., Montiel, R., Ledeneva, Y., Rendon, E., Gelbukh, A., Cruz, R.: Text Summarization by Sentence Extraction Using Unsupervised Learning. In: Orejas, F., Ehrig, H., Jantke, K.P., Reichel, H. (eds.) Abstract Data Types 1990. LNCS (LNAI), vol. 534, pp. 133–143. Springer, Heidelberg (1991)

[4] Edmondson, H.P.: New Methods in Automatic Extraction. Journal of the Association for Computing Machinery (1969)

[5] Kupiec, J., Pedersen, J., Chen, F.: A trainable document summarizer. In: SIGIR 1995 (1995)

[6] Villatoro-Tello, E., Villaseñor-Pineda, L., Montes-y-Gómez, M.: Using Word Sequences for Text Summarization. In: Sojka, P., Kopeček, I., Pala, K. (eds.) TSD 2006. LNCS (LNAI), vol. 4188, pp. 293–300. Springer, Heidelberg (2006)

[7] Chuang, T., Yang, J.: Text Summarization by Sentence Segment Extraction Using Machine Learning Algorithms. In: Proc. of the ACL 2004 Workshop, Barcelona, España (2004)

[8] Ledeneva, Y.: PhD. Thesis: Automatic Language-Independent Detection of Multiword Descriptions for Text Summarization. National Polytechnic Institute, Mexico (2009)
[9] Ledeneva, Y.N., Gelbukh, A., García-Hernández, R.A.: Terms Derived from Frequent Sequences for Extractive Text Summarization. In: Gelbukh, A. (ed.) CICLing 2008. LNCS, vol. 4919, pp. 593–604. Springer, Heidelberg (2008)
[10] Garcia-Hernandez, R.A., Martinez-Trinidad, J.F., Carrasco, A.: Finding maximal sequential patterns in text document collections and single documents. Informatica. International Journal of Computing and Informatics (34), 93–101 (2010)
[11] Ledeneva, Y., Garcia-Hernandez, R., Gelbukh, A.: Multi-document summarization using Maximal Frequent Sequences. Research in Computer Science 47, 15–24 (2010)
[12] Garcia-Hernandez, R., Ledeneva, Y., Gelbukh, A., Citlalih, G.: An Assessment of Word Sequence Models for Extractive Text Summarization. Research in Computing Science (38), 253–262 (2008)
[13] Suanmali, L., Salim, N., Salem Binwahlan, M.: Genetic Algorithm based Sentence Extraction for Text Summarization. International Journal of Innovative Computing 1(1) (2011)
[14] Silla, C.N., Pappa, G.L., Freitas, A.A., Kaestner, C.A.A.: Automatic text summarization with genetic algorithm-based attribute selection. In: Lemaître, C., Reyes, C.A., González, J.A. (eds.) IBERAMIA 2004. LNCS (LNAI), vol. 3315, pp. 305–314. Springer, Heidelberg (2004)
[15] Qazvinian, V., Sharif, L., Halavati, R.: Summarising text with a genetic algorithm-based sentence extraction. Int. J. Knowledge Management Studies 2(4), 426–444 (2008)
[16] Cruz, C.M., Urrea, A.M.: Extractive Summarization Based on Word Information and Sentence Position. In: Gelbukh, A. (ed.) CICLing 2005. LNCS, vol. 3406, pp. 653–656. Springer, Heidelberg (2005)
[17] Rada, M., Tarau, P.: TextRank: Bringing Order into Texts. In: Proceedings of the Conference on Empirical Methods in Natural Language Processing, EMNLP 2004 (2004)
[18] van Rijsbergen, C.J., Robertson, S.E., Porter, M.F.: New models in probabilistic information retrieval. En línea (1980)
http://tartarus.org/~martin/PorterStemmer/index.html
(Último acceso: Enero 28, 2013)
[19] Document Understanding Conferences. En línea (Julio 16, 2002),
http://www-nlpir.nist.gov/projects/duc/index.html2
[20] Lin, C.Y.: ROUGE: A Package for Automatic Evaluation of Summaries. In: Proceedings of Workshop on Text Summarization of ACL (2004)
[21] Lin, C., Hovy, E.: Automatic Evaluation of Summaries Using N-gram Co-Occurrence. In: Proceedings of HLT-NAACL, Canada, (2003)
[22] Ledeneva, Y., Hernández, R.G., Soto, R.M., Reyes, R.C., Gelbukh, A.: EM Clustering Algorithm for Automatic Text Summarization. In: Batyrshin, I., Sidorov, G. (eds.) MICAI 2011, Part I. LNCS, vol. 7094, pp. 305–315. Springer, Heidelberg (2011)

Author Index