

Advances in Machine Learning and Image Processing

Research in Computing Science

Series Editorial Board

Editors-in-Chief:

Grigori Sidorov (Mexico)
Gerhard Ritter (USA)
Jean Serra (France)
Ulises Cortés (Spain)

Associate Editors:

Jesús Angulo (France)
Jihad El-Sana (Israel)
Alexander Gelbukh (Mexico)
Ioannis Kakadiaris (USA)
Petros Maragos (Greece)
Julian Padget (UK)
Mateo Valero (Spain)

Editorial Coordination:

María Fernanda Ríos Zacarías

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 102**, octubre 2015. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No.: 04-2005-121611550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de Licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 102**, October 2015. Printing 500. The authors are responsible for the contents of their articles. All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording or otherwise, without prior permission of Centre for Computing Research. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Advances in Machine Learning and Image Processing

Oscar Herrera Alcántara (ed.)



Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2015

ISSN: 1870-4069

Copyright © Instituto Politécnico Nacional 2015

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional “Adolfo López Mateos”, Zacatenco
07738, México D.F., México

<http://www.rcs.cic.ipn.mx>

<http://www.ipn.mx>

<http://www.cic.ipn.mx>

The editors and the publisher of this journal have made their best effort in preparing this special issue, but make no warranty of any kind, expressed or implied, with regard to the information contained in this volume.

All rights reserved. No part of this publication may be reproduced, stored on a retrieval system or transmitted, in any form or by any means, including electronic, mechanical, photocopying, recording, or otherwise, without prior permission of the Instituto Politécnico Nacional, except for personal or classroom use provided that copies bear the full citation notice provided on the first page of each paper.

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

Editorial

This volume of the journal “Research in Computing Science” contains selected papers related to image processing and machine learning. The papers were carefully chosen by the editorial board on the basis of the at least two reviews by the members of the reviewing committee or additional reviewers. The reviewers took into account the originality, scientific contribution to the field, soundness and technical quality of the papers. It is worth noting that various papers for this special issue were rejected.

As far as image processing is concerned, the papers of this volume describe 3D visualization of the research results, influence of luminance in color segmentation, chromatic improvement of images, face recognition based only on eyes’ information, feature descriptors for augmented reality registration, binary segmentation of multiband images, and improvement of image resolution with fuzzy logic. As far as pure machine learning is concerned, the paper of this volume discuss clustering ensemble selection considering quality and diversity and feature selection aimed to improve the performance of an electric arc furnace.

I would like to thank Mexican Society for Artificial Intelligence (Sociedad Mexicana de Inteligencia Artificial) and MICA I 2015. Also, I am grateful to Polytechnic University of Morelos (Upemor), Tecnológico de Monterrey Campus Cuernavaca, Electrical Research Institute (IIE) and the National Center for Research and Technology Development (CENIDET) for their support during preparation of this volume.

The entire submission, reviewing, and selection process, as well as preparation of the proceedings, were supported for free by the EasyChair system (www.easychair.org).

Oscar Herrera Alcántara
October 2015

Table of Contents

| | Page |
|---|------|
| High Performance Computing with a Big Data: 3D Visualization of the Research Results..... | 9 |
| <i>Eva Pajorová and Ladislav Hluchý</i> | |
| Influence of Luminance L^* in the $L^*a^*b^*$ Color Space during Color Segmentation in Highly Saturated Color Images | 21 |
| <i>Rodolfo Alvarado-Cervantes, Edgardo M. Felipe-Riveron, Vladislav Khartchenko, and Oleksiy Pogrebnyak</i> | |
| Chromatic Improvement of Backgrounds Images Captured with Environmental Pollution Using Retinex Model | 33 |
| <i>Mario Dehesa, Alberto J. Rosales, Francisco J. Gallegos, Samuel Souverville, and Isabel V. Hernández</i> | |
| Face Recognition based Only on Eyes' Information and Local Binary Pattern | 41 |
| <i>Francisco Rosario-Verde, Joel Perez-Siles, Luis Aviles-Brito, Jesus Olivares-Mercado, Karina Toscano-Medina, and Hector Perez-Meana</i> | |
| Scene Dedicated Feature Descriptor with Random Forest Training for Better Augmented Reality Registration | 51 |
| <i>Andras Takacs, Edgar A. Rivas-Araiza, and Jesus Carlos Pedraza-Ortega</i> | |
| Binary Segmentation of Multiband Images | 63 |
| <i>Claudia Sánchez and Mariano Rivera</i> | |
| Fuzzy Logic Applied to Improvement of Image Resolution using Gaussian Membership Functions | 77 |
| <i>Samuel Souverville, Jorge A. Rosales, Francisco J. Gallegos, Mario Dehesa, Isabel V. Hernández, and Lucero V. Lozano</i> | |
| Clustering Ensemble Selection Considering Quality and Diversity | 89 |
| <i>Roham Ranjbar, Hamid Parvin, and Farhad Rad</i> | |
| Feature Selection for Improvement the Performance of an Electric Arc Furnace | 101 |
| <i>Amado Sánchez Sánchez, José Crispén Hernández Hernández, Haydee Patricia Martínez Hernández, David Ibarra Guzmán, Arturo Contreras Juárez, Arturo Aguila Flores, and Perfecto Malaqías Quintero Flores</i> | |

High Performance Computing with a Big Data: 3D Visualization of the Research Results

Eva Pajorová and Ladislav Hluchý

Institute of Informatics, Slovak Academy of Sciences, Bratislava,
Slovakia

utrrepaj@savba.sk

Abstract. Our research in institute is oriented on high performance computing like GRID and Cloud computing. A lot of international projects oriented on main research or in energy execution and in natural disasters damages execution utilize the HPC (High Performance Computing) as is grid or cloud computing. Within such solution raises requirement of visualization service for presentation of the intermediate or final research results. Our basic aim of our research resolved in projects is the creation visual service for the modeling and 3D rendering of research simulations results. We have developed some 3D visualization tools for present simulation results for applications on the field of astronomical main research and research in natural disasters or disasters in public buildings, before fire, floods, blackout and bombs. But also we have developed visualization in the field of energy execution results with a big data. Paper presents some of 3D visualization tools, whose have been developed in our institute.

Keywords: Big data, high performance computing, grid, 3D visualization tool

1 Introduction

Astronomical simulations, natural disasters, energy and crisis simulation is a very complicated, challenging problem sensitive to the input big data required. Therefore, intense research and development of sophisticated software systems and tools is extremely important for such simulations [6]. For our - Slovak forests, original methodology for forest vegetation classification and new fuel models have been developed and proper forest fire simulations related to the locality Krompla (National Park Slovak Paradise), where the large destructive fire appeared and its reconstruction have been analyzed. These efforts induced the end of better auxiliary tools for 3D visualization of obtained simulation results and for animation of the forest fire spread [3].

Advance in sciences and engineering has put high demand on tools for high-performance large-scale visual data exploration and analysis. For example, *astronomical scientists* can now study evolution of all solar systems on lot of astronomical simulations. These simulations can generate large amount of data, possibly with high resolution (in three dimensional space) and long time series. Single-

system visualization software running on commodity machines cannot scale up to the large amount of data generated by these simulations. To address this problem, there have been developed a lot of different Grid-based visualization frameworks for time-critical, interactively controlled file-set transfer for visual browsing of spatially and temporally large datasets in a Grid environment. Lot of frameworks for grid and cloud based visualization are solved such problem. We can travel during evolution of sophisticated grid based visualization frameworks with a new one actualized functionalities. For example „Reality Grid“, „Uni - Grid“, „Terra Grid [6] “.The Uni – Grids [1] project developed a Grid Service infrastructure compliant with the Open Grid Service Architecture (OGSA). It is based on the UNICORE Grid software initially developed in the German UNICORE and UNICORE Plus projects [2]. The paper presents a design and utility of 3D visualization tools which are solving in our Institute.

2 Astronomical simulations

The design is tested on the astronomical simulations in the scope of collaboration between Astronomical Institute of SAS, Catania Observatory and Adam Mickiewicz University in Poznan. The simulation was ported to EGEE by Institute of Informatics SAS [4].

For a long time computing of Intensive Parametric Studies one of a main problem is to control if executions converge to the correct way. Client naturally wants to see the intermediate results. Such problem calls for correct way of submission representing a reduction of the number of jobs as a visual control of the results in time while the application is running. The described tool is able to visualize the partial results of the application. The user can completely control the job during execution, and can change the input parameters while execution is still running. Both tools - tool for submission, designed before and continued sequence visualization tool provided complete solution of the specific main problem in Grid environment.

Visualization is designed as a plug in module. Client asking for visualization is as a Visualization client. Outputs data on the storage element are as an inputs data for visualization jobs. Configuration file will be included by Visualization tool (VT) for the application. Therewith can be activated control script. Workers to modify data to the visualize formats, but not only, workers also to prepare the typical visualization scenes. Client can to render such scenes on the browser and he can make the visual control and to modify executions. Multiple visualizations generated from a common model will improve the requirements creation, reviewing and understanding process. Visual representations, when effective, provide cognitive support by highlighting the most relevant interactions and aspects of a specification for a particular use. The goal of scientific visualization is to help scientists view and better understand their data. This data can come from experiments or numerical simulations. Often the size and complexity of the data makes it difficult to understand by direct inspection. Also, the data may be generated at several times during an experiment or simulation and understanding how the data varies with time may be difficult. Scientific visualization can help with these difficulties by representing the data so that it may be viewed in its

entirety. In the case of time varying data, animations can be created that show this variation in a natural way. Using virtual reality techniques, the data can be viewed and manipulated naturally in a true three dimensional environment (e.g. depth is explicitly perceived and not just implied). All these techniques can allow scientists to better understand their data. Viewing the data in this way can quickly draw the scientist's attention to interesting and/or anomalous portions of the data. Because of this, we encourage scientists to use scientific visualization from the beginning of their experiments and simulations and not just when they think they have everything operating correctly. This also allows the scientists to develop a set of visualization tools and techniques that will help them understand their data as their research matures. In depend of our astronomical example; in order to understand immediately the evolution of the investigated proto-planetary disc we have developed a Visualization Tool (VT). Examples as output from VT you can see on figures 1 and 2.

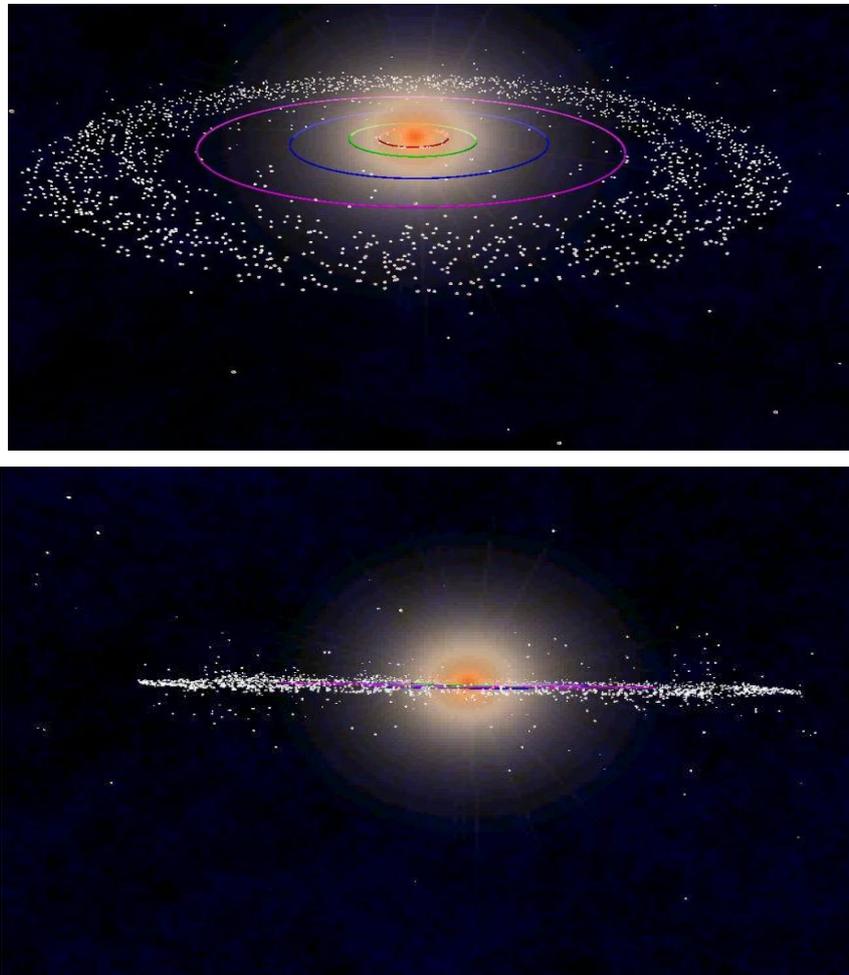


Fig. 1. Evolution of Protoplanetary disk during first giga year.

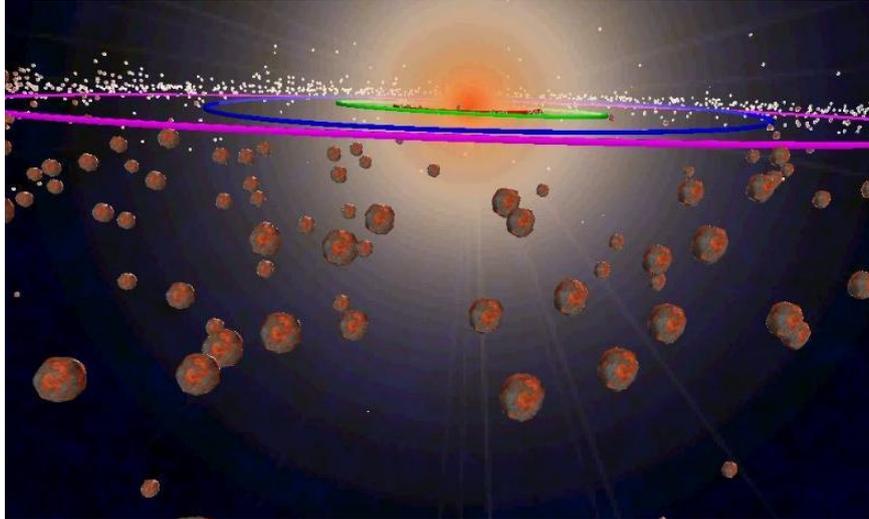


Fig. 2. The final results of evolution the Protoplanetary disk

3 Simulations of Natural Disasters

Natural disaster simulation and execution with a huge data usually spend long execution time. Good solution for execution is represented by grid and actually on cloud computing. In both infrastructures visualization has the main position as a way to control the execution process. Visual control has in all infrastructure very useful position. The simulation was realized as a sequence of parameter studies, where each sub-simulation was submitted to the grid as a separate parameter study. The job management was rather time consuming due to the analysis of failed jobs and to their re-submission. Visualization is included as a visual control process. For example, understand the fire spread we have developed a Visualization Tool (VT). The VT is composed of several modules, which are responsible for creating scenes and converting data to, the “visualize” format. The components generating rendering scenes are easy to exchange, according to the requirements of the given application. In case of our gratified application the output data of the simulation located on the SE can be used directly as the input for the VT. The final product of the VT includes a set of files containing data in the VRML (Virtual Reality Modeling Language) format. These output files can be rendered by many available VRML web-browsers. The whole visualization process is maintained through a visualization script, whose basic function is invoking the individual VT components in successive steps, transferring data, and handling error events. The script is written using the Bourne shell scripts and all VT modules are implemented in the C++ language. The VT can be embedded into the framework described above, or can be used separately as a stand-alone program. By using the VT the client can stop the execution process, change the input parameters and restart the execution process again. In grid environment, such architecture can be used for all applications from different science spheres which have the character of a

parametric study. Actually, the research community needs not only “traditional” batch computations of huge bunches of data but also the ability to perform complex data processing; this requires capabilities like on-line access to databases, interactivity, fine real-time job control, sophisticated visualization and data management tools (also in real time), remote control and monitoring. The user can completely control the job during execution and change the input parameters, while the execution is still running. Both tools, the tool for submission designed before and continued sequential visualization tool, provide complete solution of the specific main problem in Grid environment. Example of big fire is shown in figure 3.

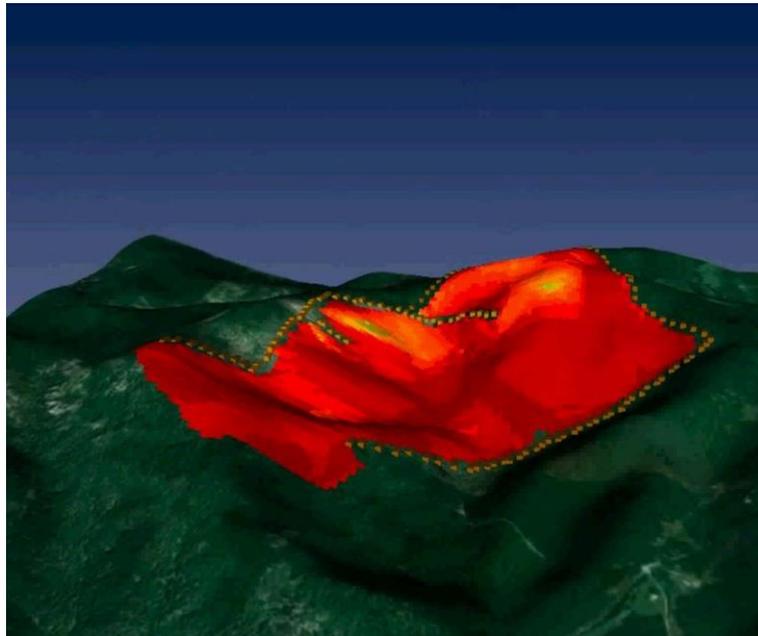


Fig.3. Visualization from big fire on Krompla hill.

4 Crisis Simulations

In crisis time like are fires, bombs, floods ...many deaf people are using lip reading as a main communication form. A viseme is a representational unit used to classify speech sounds in the visual domain and describes the particular facial and oral positions and movements that occur alongside the voicing of phonemes. A design tool for creating correct speech visemes is designed. It's composed of 5 modules; one module for creating phonemes, one module for creating 3D speech visemes, one module for facial expression and module for synchronization between phonemes and visemes and lastly one module to generate speech triphemes. We are testing the correctness of generated visemes on Slovak speech domains. Our developed tool is one of the actually developed software tools which ensure smooth link between regular distance learning and training of hearing impaired. For this, additional Sing Language (SL) information will be

inserted in the main data stream of the video warning message. The corresponding SL information will be represented by video sequences with two SL interpreters; one will be contour image and second will be virtual speaking head. Both will be visualized scaled down in one of the lower corners of the main image. In order to significantly reduce this additional information, the contour images will be used instead of the full video. They are obtained after processing of the consecutive TV frames of the SL interpretation. The contour images represent very well the movements of the interpreter's hands and give very good vision of his/her face expression, which is of high importance for the sign comprehensibility. This approach permits hearing impaired people orientate himself during crisis situations like are fire, flood, bombs in big halls, in big store, in railway station, cinemas and exhibition grounds. In Crisis time the monitors included in public areas have changed mode. They change normal mode to crisis mode and they are starting to generate the warning messages [14]. A tool is able to complete warning messages with text windows included on bottom of the monitor; see figure 4. Text window is included on bottom of the monitor. Slovak speech visemes you can see on figure 5.



Fig. 4. Virtual head with text windows included on bottom of the monitor.

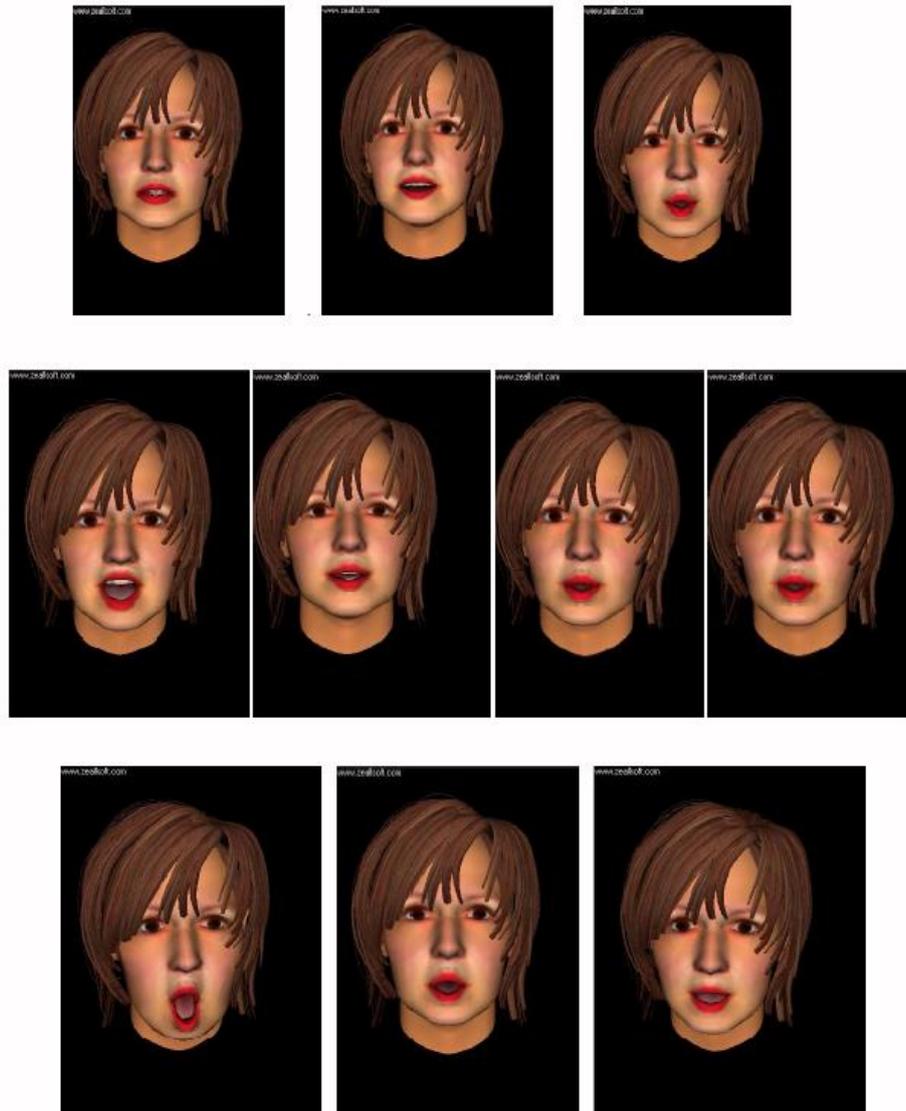


Fig. 5. The main of Slovak speech visemas.

5 Energy-water Management Simulations

Environment, that provides sufficient information for 3D calculations and simulations in water management required to design a 3D virtual terrain of Bratislava and its surroundings and the water supply pipes, water towers and 3D view nodes. According

to the third coordinate of an annotation type (.dgn) files to transfer, we received the shape-files, thus obtained 3D coordinates. We used them to create a digital 3D model - TIN of the Bratislava and surroundings terrain. See figure 6.

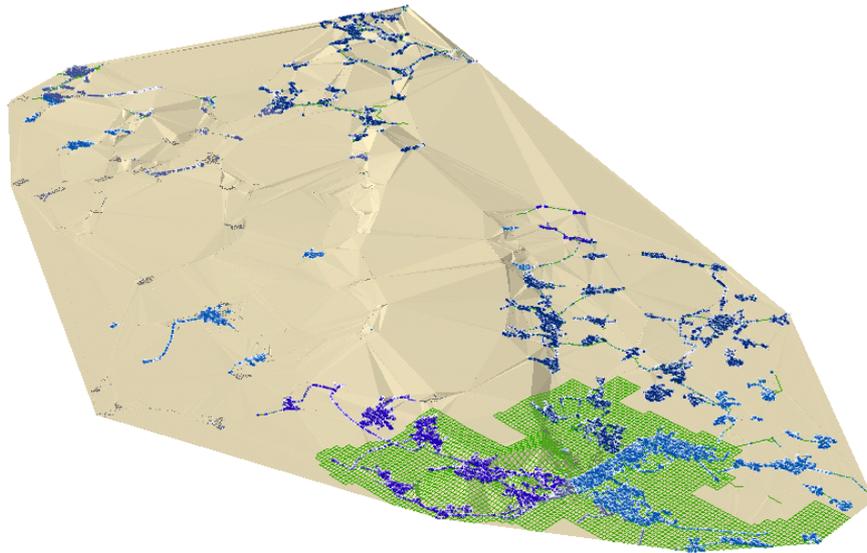


Fig. 6. 3D model -"TIN" of Bratislava and surroundings

In such an environment we've created from the 3D model to which we show ortho-photomap of Bratislava and surroundings. The advantage of 3D visualization is that pipes for the purpose of displaying the results of the simulations we can tap to display not just beneath the surface, where it is located, but also on the surface of the ground in fact [1], where you can watch the results of the simulation figure 7.



Fig. 7. The 3D model of Bratislava and surroundings with water pipes.

An important component for the display of water pressure and the flow rate of water loss, are high-rise buildings. In order to make it easier to differentiate the amount of buildings, so we knew on a virtual model of the building so that buildings appear with one floor are displayed in a different color than the building with two floors and three floors, as well as other buildings, etc. So we have achieved a virtual model of the environment and the surroundings of Bratislava with water pipes, water - towers and nodes and also with the buildings, the amount of which is colorfully differentiated. 3D virtual model are designed to provide an adequate environment for displaying the results of calculations already in the water economy [2]. See figure. 8.



Fig.8 Virtual model of Bratislava and surroundings, with the water pipes and buildings

In this environment the proposed 3D model displaying calculation results from the simulation calculated with the simulation program EPANET [8] on the computer clusters. To view a simulation of the water pressure within 24 hours after we have chosen 3D display using transparent 24 - TINs - models, created from HGL01 up to HGL24 output data. From these results we have created 24 transparent 3D TIN - models. Each TIN is the level of water pressure due to the high-rise buildings. If any of the extra tall buildings overlap the transparent TIN, it means that its level is over the normal water pressure. See figure 9.

Water flow rates during 24 hours we display from Flow01 to Flow24 outputs data. To view the flow of water during the 24 hours we have chosen one color scale. In the framework of this color range are graded from minus maximum value till plus the maximum value; a total of 8 fields. See in the figure 11.

Outputs, which we have test till yet, have been the executed during the 24 hours, where they were too small differences in the data which is also reflected on the display. In the future we want to display multiple simulation outputs from EPANET cluster executions, before for more time range. Also we will try to find the best form of 3D display and animation outputs.



Fig. 9. A sample of 3D visualization of the simulation results computed by EPANET-water pressure during 24 hours.

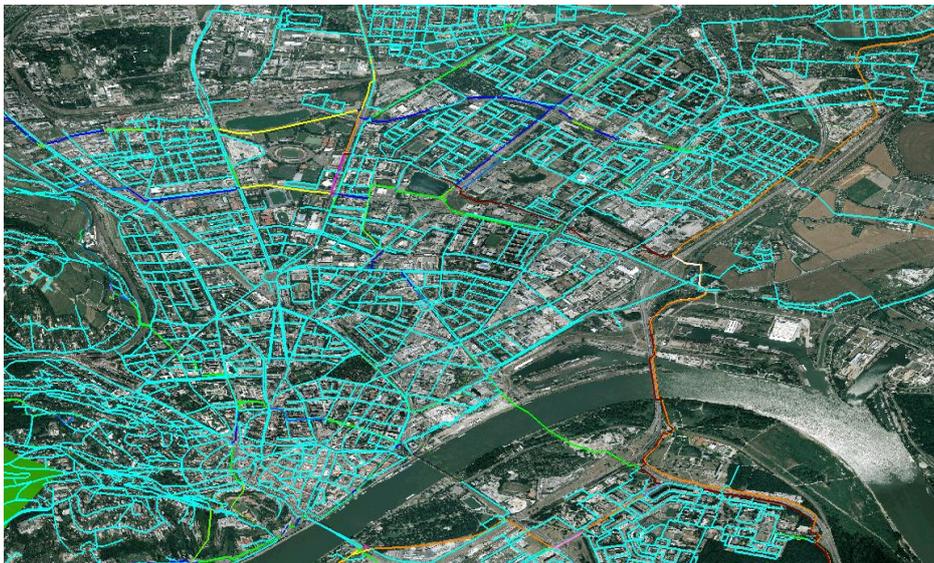


Fig.10. Examples showing the results of the simulation of water flow during the 24 hours, calculated by the simulation program EPANET and computing on clusters.

6 Conclusions

This paper proposed a developed 3D Visualization tools for the visualization output research results from simulations in different research fields. In a future we plan to innovate and expand our new one visualization tools, which meets in maximum the requirements of clients.

Acknowledgements. The article is supported by the project: Centre of water supply risk of a large city, Nr. 26240220082 and VEGA 2/0054/12.

References

1. The Reality Grid Project on the web: <http://www.realitygrid.org/> Uni Grid on web: <http://www.unigrids.org/>
2. Glasa, Ján et al.: Analysis of forest fire behavior by advanced computer fire simulators. *Communications Scientific Letters of the University of Žilina*, vol. 2, p. 26–31 (2011)
3. Astaloš, Ján et al.: Slovak Participation in the World LHC Computing Grid. In: 6th International Workshop On Grid Computing For Complex Problems, Bratislava: GCCP, L. Hluchý, P. Kurdel, J. Sebestyénová (eds.), Bratislava : Institute Of Informatics SAS, pp. 21–27 (2010)
4. Wilkins-Diehr, N., Gannon, D., Klimeck, G., Oster, S., Pamidighantam, S.: TeraGrid Science Gateways and Their Impact on Science. *IEEE Computer* 41(11):32–41 (2008)
5. Cruz, M.G., M.E. Alexander, R.H. Wakimoto: 3D Nature, LLC. Using VNS (Manual).
6. Arvada, C.O.: 3D Nature, LLC. Assessing canopyfuel stratum Emerging 16 Technology characteristics in crown fire from fuel types of North America. *International Journal of Wildland Fire* 12:39–50 (2003)
7. Orland, B.: SmartForest: a 3-D interactive forest visualization and analysis system. In: *Proceedings, Decision Support 2001 - Resource Technology 94*, pp. 181–190, Bethesda, MD: American Society for Photogrammetry and Remote Sensing (1994)
8. Software: EPANET. | Drinking Water Research | US EPA , EPANET is software that models water distribution piping systems. EPANET is public domain software that may be freely copied and distributed. <http://www.epa.gov/nrmrl/wswrd/dw/epanet.html>

Influence of Luminance L^* in the $L^*a^*b^*$ Color Space during Color Segmentation in Highly Saturated Color Images

Rodolfo Alvarado-Cervantes¹, Edgardo M. Felipe-Riveron^{2*}, Vladislav Khartchenko¹ and Oleksiy Pogrebnyak²

¹ Centro de Investigaciones Teóricas, Facultad de Estudios Superiores Cuautitlán, Universidad Nacional Autónoma de México, Cuautitlán Izcalli, Mexico

² Instituto Politécnico Nacional, Centro de Investigación en Computación, Mexico

rodolfo.alvarado.cervantes@gmail.com; edgardo@cic.ipn.mx; vlad@unam.mx; olek@cic.ipn.mx

Abstract. In this paper a study of the influence of luminance L^* at the CIE $L^*a^*b^*$ color space during color segmentation in highly saturated color images is presented. A comparative study is made between the behavior of segmentation in color images using (1) the Euclidean metric of the RGB channels (2) the Euclidean metric of a^* and b^* in CIE $L^*a^*b^*$ color space and (3) an adaptive color similarity function defined as a product of Gaussian functions in a modified HSI color space. For the evaluation, synthetic images were particularly designed to accurately assess the performance of the color segmentation. The testing system can be used either to explore the behavior of a similarity function (or metric) in different color spaces or to explore different metrics (or similarity functions) in the same color space. From the results it was obtained that the color parameters a^* and b^* are not independent of the luminance parameter L^* as one might initially assume. In the majority of cases the CIE $L^*a^*b^*$ color space was more influenced by the faded shadow than the RGB color space. The segmentation using the Euclidean metric in $L^*a^*b^*$ color space suffered errors in all cases. It manifested in different degrees and at different levels of faded shadow (less than 10% to 80%).

Keywords: Color image segmentation; CIE $L^*a^*b^*$ color space; color metrics; color segmentation evaluation; synthetic color image generation

* Corresponding author.

1 Introduction

Image segmentation consists of partitioning an entire image into different regions, which are similar in some predefined manner. It is an important and difficult task in image analysis and processing. All subsequent steps, such as object recognition depend on the quality of segmentation [1].

For some time the development of segmentation algorithms attracted remarkable consideration compared with the relatively fewer efforts on their evaluation and characterization [2, 3, 4, 5]. Since none of the proposed automatic segmentation algorithms published is generally applicable to all types of images and different algorithms are not equally suitable for particular applications, the performance evaluation of segmentation algorithms and its characterization are very important subjects in the study of segmentation [3, 5].

Perceptual uniform color spaces such as CIE $L^*a^*b^*$, with the Euclidean metric to quantify color distances are commonly used in color image segmentation of natural scenes using histogram based or clustering techniques among others [1].

To evaluate the segmentation performance of the Euclidean metric in the $L^*a^*b^*$ color space, we designed a system that generated synthetic color images, with its associated ground truth (GT), and evaluated the results with Receiver operating characteristics (ROC) curves [7]. A short study of evaluation methods is presented in section 2. We present the evaluation system in section 3, where the synthetic images, designed to evaluate the efficiency of achieved color information from given segmentation algorithms are explained in detail. A comparative study between the behavior of segmentation in color images using (1) the Euclidean metric of the RGB channels (2) the Euclidean metric of a^* and b^* in the $L^*a^*b^*$ color space and (3) an adaptive color similarity function (defined as a product of Gaussian functions in a modified HSI color space [6]) is presented in section 4. Conclusions are given in section 5.

2 Previous Works

In recent years considerable effort has been devoted to the problem of color segmentation in digital images given its importance and potential. Until few years ago, the majority of published approaches for the segmentation of color were based on monochromatic techniques applied to each color component of the image in different color spaces (RGB or other) and in different ways to produce a color composite. These approaches have an inherent problem of significant loss of color information during the process [6].

The first comprehensive survey on evaluation methods of image segmentation is presented in Zhang (1996). It brings a coherent classification of existing methods at that time. Progress made in the subject during the five years after the first survey is presented in Zhang (2001). Another actualization is presented five years later¹⁵ embracing together the principal methods of segmentation evaluation available up until 2007.

Zhang and others (2006) present a comprehensive survey on unsupervised methods of segmentation evaluation. The authors propose a hierarchy of published methods at that time by summarizing the main branches and locating the group of unsupervised methods on it. They mention their advantages, such as no requirement for GT to obtain quantitative results. They also propose the main lines of future research for this kind of methods.

Zhang and Gerbrands (1992) present a way to design synthetic images and a corresponding GT for evaluating segmentation algorithms. They introduce a general framework and general design considerations. They also present a system for generating synthetic images in shades of gray taking into account their design considerations. The behavior of a segmentation method in gray images using thresholding is studied and some remarks are obtained.

3 Design of Synthetic Images for Benchmark Testing

In [4] the authors present three important design considerations for creating synthetic images: 1. Synthetic images should be appropriate for a quantitative study and should allow objective evaluations of their properties; 2. The synthetic images should reflect the main features of real images, i.e. corruption factors, such as noise and blurring, variation of parameters such as size, shape, etc.; 3. The system should allow the generation of images with progressive variations of each parameter. In this way the study of the influence of each individual parameter is possible.

Comparative tests between: (1) the Euclidean metric using only a^* and b^* parameters in the $L^*a^*b^*$ color space [8], (2) the Euclidean metric of the RGB channels, and (3) the adaptive color similarity function presented in [6] were performed. The manner in which the tests were implemented is as follows:

In the case of the $L^*a^*b^*$ color space, the RGB image was previously transformed to $L^*a^*b^*$ color space discarding in all cases the luminance L^* in order to calculate the Euclidean distance on the planes a^*b^* (color information) independently of the illumination. Then the centroid (average of the values a^* and b^*) representing the colors of the figure and the background in the color space $L^*a^*b^*$ was calculated. Details are shown in [8].

In the second case the Euclidean distance of the R, G and B color channels is calculated. The centroids are obtained taking the average of the values R, G, and B.

For the case (3) of the adaptive similarity function [6] the following steps were performed:

1. Samples of both background and figure were taken, from which centroid and standard color dispersion was calculated. Details can be consulted in [6]
2. The 24-bit RGB image (true color) was transformed to a modified HSI color space.
3. For each pixel, the similarity function to the centroids of figure and background was calculated creating two color similarity images (CSI) [6].
4. Each pixel of the RGB image was classified by calculating the maximum value for each pixel position between the CSI images of the figure and that of the background.

The base shape of the synthetic test image was created with the following features:

1. Concave and convex sections in order to make it more representative of real images, such as natural flowers. 2. Extreme omnidirectional curvature in the entire image to hinder obtaining the edges applying mask edge detectors. 3. The object was centered in the image.

The resulting flower-shaped object in the image is considered as the object of interest and as the ground truth GT in all subsequent tests (Figure 1 left).

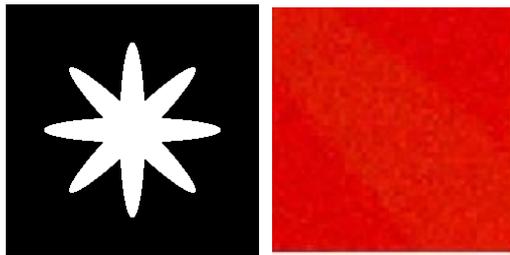


Fig. 1. Flower-shaped ground truth (left) and an image zoomed showing the introduced Gaussian noise (right)

In addition to this object of interest, several features were imposed in order to hinder its color-based segmentation:

1. Low contrast. The contrast between the object and the background in all images was very low for an observer, including some in which at a first glance the user cannot see the difference (e.g. Flower 5 in Figure 2). The difference between the color characteristics of the object of interest and the background we call “Delta” and it occurs at different directions of the HSI space. The tests were performed in color quadrants 0, 60, 120, 180, 240 and 300 degrees. 2. Blurred edges with an average filter. A sliding mean filter of size 3 x 3 pixels was applied to the whole image in order to blur the corners and to make object detection more difficult; this was done before the introduction of Gaussian noise. 3. Introduction of Gaussian noise with SNR value = 1 (Figure 1 right). The noise was applied to each of the RGB channels individually, and later we assembled the channels to create the RGB color image with noise. Figure 1 right shows an example.

The basic colors selected for both object and background were based on maintaining constant intensity to 0.9 and saturation to 0.9 and only varying the hue. Hue was selected as the parameter because its change integrates the three RGB color channels together, making it more difficult to be processed by extending grayscale techniques to each color channel, thus forcing the segmentation algorithms in evaluation to use the color information holistically.

Samples of pixels corresponding to the figure were obtained by two squares of 2 x 2 pixels starting at the pixel (84, 84) and (150, 150). Samples for background pixels were obtained by two squares of 2 x 2 pixels starting at pixel (15, 15) and (150, 180).

The images were generated in the sectors 0, 60, 120, 180, 240 and 300 degrees corresponding to the images flower_0, flower_1 ... flower_5 (Figure 2). To these test images we later applied to each one a faded shadow in increments of 10% in each step.



Fig. 2. Testing with High Saturation with Delta in HUE

A shadow fading was applied to all noisy blurred images with the light center in the fixed coordinates (150,150) in images of 256 x 256 pixels. It was applied gradually with 10% increments in each step. Figure 3 shows this for Flower 0.



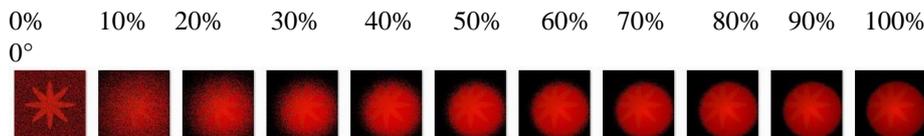
Fig. 3. Example in color quadrants with a faded shadow applied at 0 degrees

4 Results and Discussion

In this section we show the results in terms of TP (true positives) and FP (false positives) plotted against the level of shadow fading, representing each 10% step of increment. The first position means no shadow and position 11 means 100% shadow fading. All the images had the same post-processing: elimination of areas smaller than 30 pixels and a morphological closing with a circular structuring element of radius equal to two pixels.

The results of the application with the solution given by [6] of the color image segmentation with a different level of shadow fading (shown in every bottom row of each color) compared with those obtained with the Euclidean metric of the a^* and b^* parameters in the $L^*a^*b^*$ color space (shown in every top row of every color) and the results obtained with the Euclidean metric of the RGB color channels (shown in every middle row of every color) are included in Figure 4 for each color quadrant (0° , 60° , 120° , 180° , 240° and 300°) and at 10% increments of the shadow fading.

As it is shown in the graphs in Figure 5 (plotting TP and FP of each level of faded shadow) and in coincidence with the visual analysis of the corresponding flower (see Figure 4), segmentation failures in the $L^*a^*b^*$ space start at different levels of faded shadow, whereas the color similarity function [6] is practically immune to the faded shadow (see Figure 6).



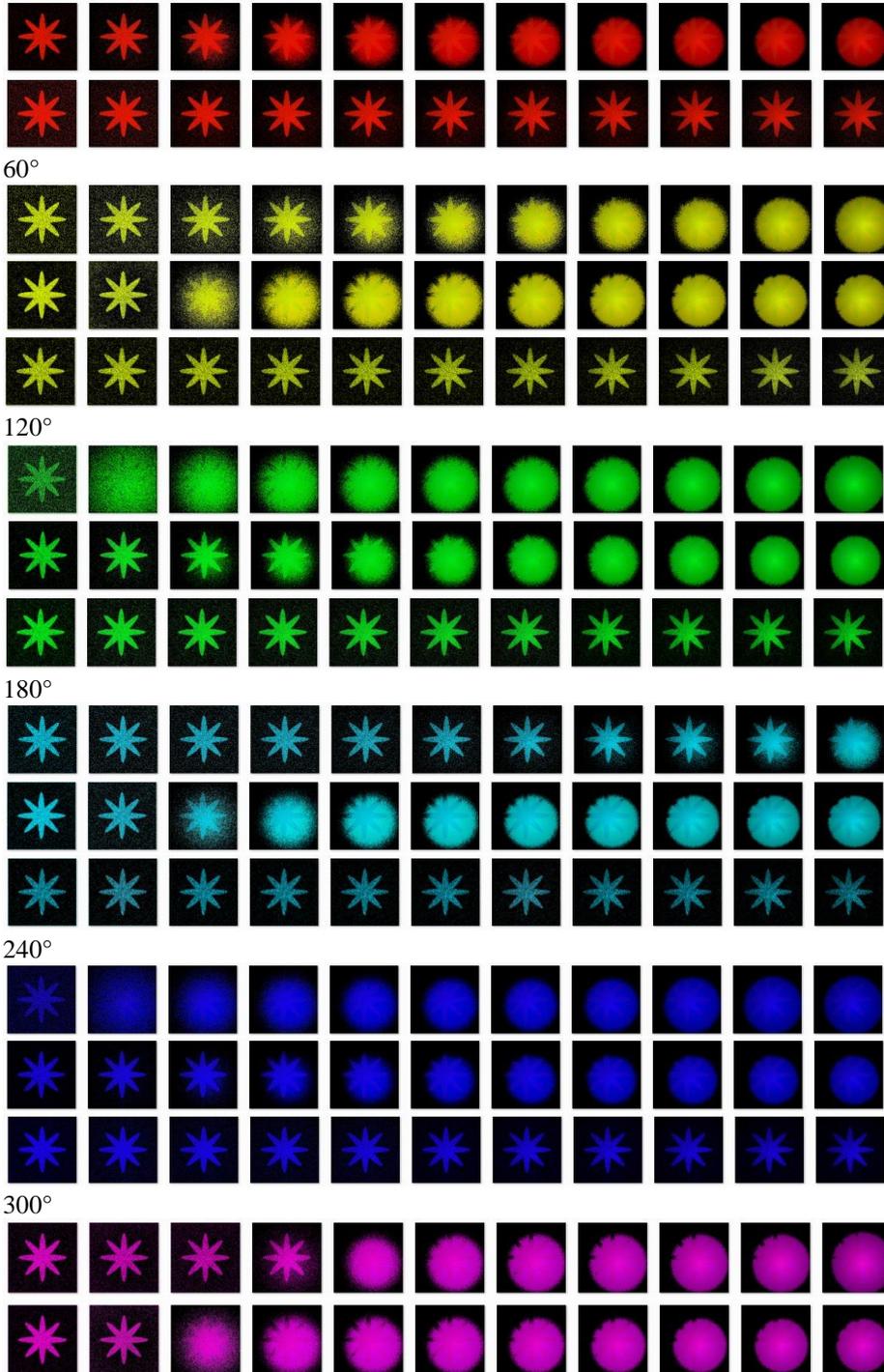




Fig. 4. Results of the color segmentation achieved between the Euclidean metric of the a^* and b^* parameters in the $L^*a^*b^*$ color space (top rows of each color), the Euclidean metric of the RGB color channels (middle rows of each color) and the adaptive color similarity function [6] (bottom rows of each color), for each color quadrant (0° , 60° , 120° , 180° , 240° and 300°) and at 10% increments of shadow fading in each step.

Three types of trends can be noticed in sectors with 120 degrees of difference: 1. Rise of the curve abruptly (Flowers 0, 2 and 4 which corresponds to the R G B color channels) with high sensibility to the faded shadow (higher than RGB); 2. Slow Rise (Flowers 1 and 5) lower than RGB, and 3 Insensitive increase (Flower 3) at near 90%. Table 1 summarizes the observations concerning the behavior of the plot curves comparing the Euclidean metric in $L^*a^*b^*$ color space, Euclidean metric in RGB, and the adaptive color similarity function [6].

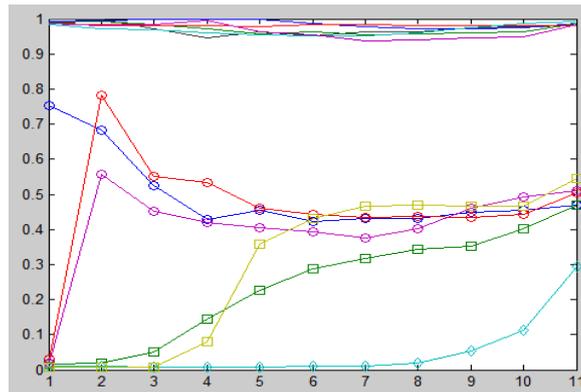


Fig. 5. Plot of TP and FP using Euclidean metric of a^* and b^* parameters in $L^*a^*b^*$ color space

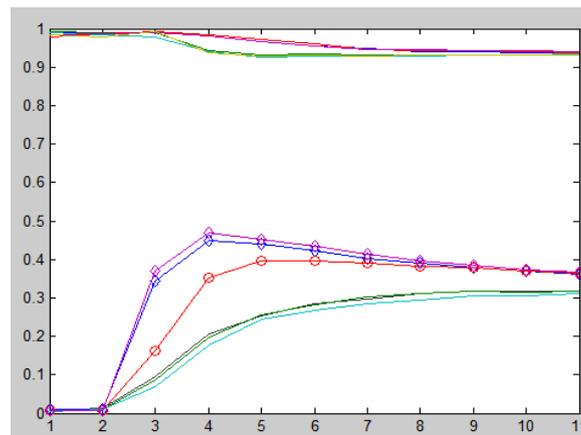


Fig. 6. Plot of TP and FP using Euclidean metric of RGB

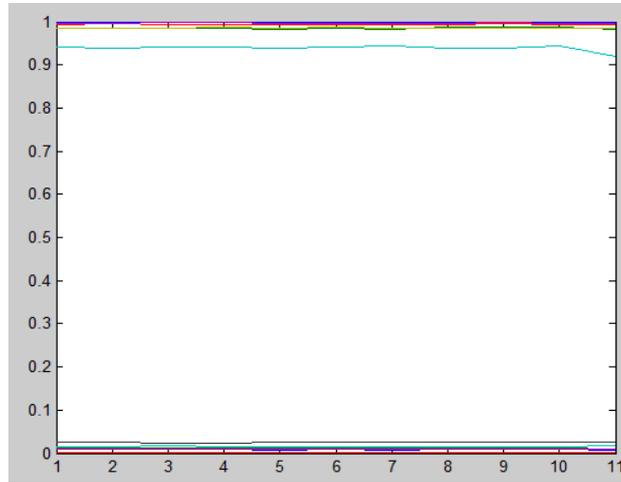


Fig. 7. Plot of TP and FP using the adaptive color similarity function [6]

Table 1. Observations concerning the behavior of the plot curves comparing L*a*b*, RGB, and an adaptive color similarity function

| Flower | Line Color | Euclidean metric in L*a*b* rejecting L* | Euclidean metric in RGB | Color similarity function [6] |
|--------|------------|--|--|-------------------------------|
| 0 | Blue | 0% (position 1) Worst case | 20% (position 3) Sharply increases | Immune |
| 1 | Green | 30% (position 4) Increases slowly and progressively | 30% (position 4) Increases slowly and progressively | Immune |
| 2 | Red | 10% (position 2) Sharply increases | 30% (position 4) Sharply increases | Immune |
| 3 | Cyan | 80% (position 9) Increases at 45° | 30% (position 4) Increases slowly and progressively | Immune |
| 4 | Purple | 10% (position 2) Sharply increases | 20% (position 3) Sharply increases | Immune |
| 5 | Yellow | 30% (position 4) Sharply increases | 30% (position 4) Increases slowly and progressively | Immune |

To obtain representative ROC curves illustrating the behavior of the Euclidean metric in L*a*b* rejecting L*, the Euclidean metric of RGB and the adaptive color similarity function [6] in all color sectors under study, we calculated the average TP and FP for all color flowers, obtaining the results shown in Figures 8, 9 and 10.

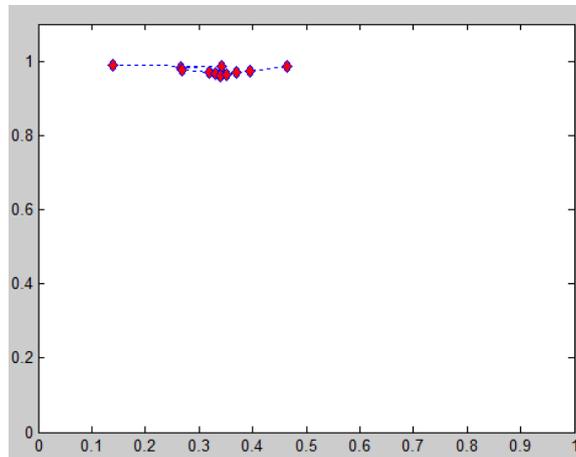


Fig. 8. ROC curve of Euclidean metric of a^* and b^* parameters in $L^*a^*b^*$ color space.

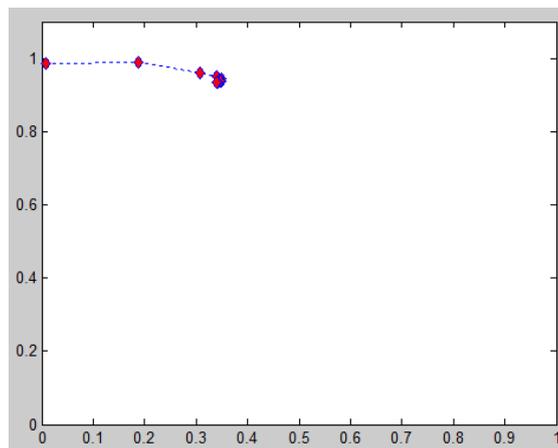


Fig. 9. ROC curve of the Euclidean metric of RGB Channels

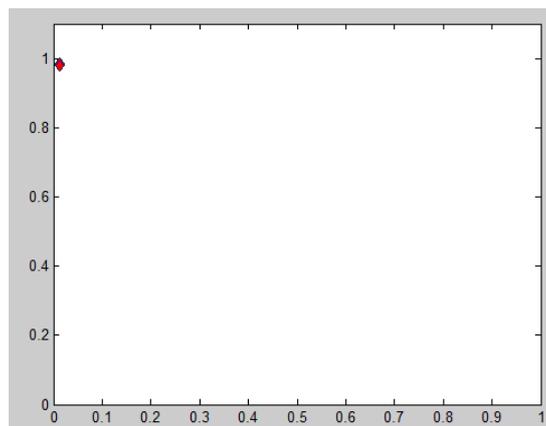


Fig. 10. ROC curve of the adaptive color similarity function [6]

In the ROC curve corresponding to the average of TP and FP of all flowers, it can be seen that the CIE $L^*a^*b^*$ results were poor from the beginning (worse than the RGB results) and continually moves to the upper right area of the ROC curve that can be thought of as the 'liberal' side (coordinate 1, 1) as they make positive classifications, and, although there is weak evidence that almost all positives were classified properly, they have a high rate of false positives.

The RGB results keep stable only the initial two steps and quickly moves to a point in the upper right area of the ROC curve that can be thought of as the 'liberal' side (coordinate 1, 1) as they make positive classifications, and, although there is weak evidence that almost all positives were classified properly, they have a high rate of false positives.

From the shown corresponding ROC curve it follows that the adaptive color similarity function is maintained in the high efficiency area (coordinate 0, 1) while the color segmentation in CIE $L^*a^*b^*$ space and the Euclidean metric of the RGB color channels progressively move away from the high efficiency area with a relatively small change in luminance L^* .

5 Conclusions

Regarding the evaluation of the color segmentation method with really difficult conditions, we can notice that the segmentation algorithm using the CIE $L^*a^*b^*$ color space and discarding L^* in calculating the Euclidean distance, suffered errors in all cases. It manifested in different degrees and at different levels of faded shadow (less than 10% to 80%). Three types of trends can be noticed in sectors with 120 degrees of difference: 1. Rise of the curve abruptly (Flowers 0, 2 and 4 which corresponds to the R G B color channels) with high sensibility to the faded shadow (higher than RGB); 2. Slow Rise (Flowers 1 and 5) lower than RGB, and 3 Insensitive increase (Flower 3) at near 90%.

The segmentation algorithm implementing the Euclidean metric of the RGB color channels maintained a similar and regular behavior with two types of trends: (1) sharp increase of errors manifested in this case in 20% to 30% of the fading shadow and (2) increases slowly and progressively in 3 cases.

The adaptive color similarity function performed well in all tests and remained close to the high efficiency zone of the ROC curves (coordinates 0, 1) without noticeable changes when the level of faded shadow increases as shown in the corresponding PLOT curves.

As it can be seen from the results of both direct segmentation, and from PLOT & ROC curves, the adaptive color similarity function in all cases exceeded: (1) The Euclidean distance in color space $L^*a^*b^*$ but discarding L^* and the Euclidean metric of the R, G and B color channels and (2) the use of Euclidean distance of the RGB channels. The adaptive color similarity function performed well in all cases with rates higher than 95% of true positives (TP) and false positive (FP) rate less than 3% on average.

According to the experimental results we believe that keeping high values of TP increased only from the FP is due to the position of the center of the shadow fading in

(150, 150). If this position is moved away from the object of interest, we can reduce the quantity of TP.

In future work we wish to evaluate different color zones like with different saturations, gray images, and with delta saturation among others. Our testing system can be used either to explore the behavior of a similarity function (or metric) in different color spaces or to explore different metrics (or similarity functions) in the same color space. Instead of exchanging color spaces in the experiments, it would only be necessary to exchange the metric or the similarity function.

It can be noticed that the non-consideration of the luminance parameter L^* in calculating Euclidean distance in the CIE $L^*a^*b^*$ color space (in each pixel of the object or of the background) did not made it immune to changes in lighting; so simple shadow can alter the quality of the results, concluding from them that the parameters a^*b^* from the CIE $L^*a^*b^*$ color space are not independent of the L^* parameter as one might suppose.

Acknowledgements. The authors of this paper wish to thank the Centro de Investigaciones Teóricas, Facultad de Estudios Superiores Cuautitlan (FES-C); Universidad Nacional Autónoma de México (UNAM), México; PAPIIT IN112913 and PIAPIVC06, UNAM; Centro de Investigación en Computación (CIC); Secretaría de Investigación y Posgrado (SIP); Instituto Politécnico Nacional (IPN), México, and CONACyT, México, for their economic support to this work.

References

1. Plataniotis, K.N., Venetsanopoulos, A.N.: Color Image Processing and Applications. First Edition, Springer, Berlin Heidelberg Germany; 354 p. (2000)
2. Zhang Y.J.: A Survey on Evaluation Methods for Image Segmentation. Pattern Recognition 29(8), 1335–1346 (1996)
3. Zhang, Y.J.: A review of recent evaluation methods for image segmentation. In :Proceedings of the 6th International Symposium on Signal Processing and Its Applications, pp. 148–151 (2001)
4. Zhang Y.J, Gerbrands J.J.: On the Design of Test Images for Segmentation Evaluation. In: Proceedings EUSIPCO 1, pp. 551–554 (1992)
5. Zhang Y.J.: A Summary of Recent Progresses for Segmentation Evaluation. In: Zhang Y.J. Advances in Image and Video Segmentation. IGI Global Research Collection, Idea Group Inc (IGI), pp. 423–439 (2006)
6. Alvarado-Cervantes R., Felipe-Riveron E.M., Sanchez-Fernandez L.P.: Color Image Segmentation by means of a Similarity Function. In: 15th Iberoamerican Conference on Pattern Recognition, CIARP 2010, Rodolfo Alvarado-Cervantes, Edgardo M. Felipe-Riveron y Luis P. Sánchez-Fernandez, I. Bloch, R.M. Cesar, Jr. (Eds.): Sao Paulo, Brazil, November 8-11, 2010, LNCS 6419, pp. 319–328, Springer, Heidelberg (2010)
7. Fawcett T.: An introduction to ROC analysis. Pattern recognition Letters 27: 861–874 (2006)
8. <http://www.mathworks.com/help/images/examples/color-based-segmentation-using-the-l-a-b-color-space.html> (Revised on April 27, 2015)

Chromatic Improvement of Backgrounds Images Captured with Environmental Pollution Using Retinex Model

Mario Dehesa, Alberto J. Rosales, Francisco J. Gallegos, Samuel Souverville,
and Isabel V. Hernández

Instituto Politécnico Nacional, ESIME Zacatenco, Ciudad de México,
Mexico

m_dehesa@hotmail.com, arosaless@ipn.mx

Abstract. A commonly problem of digital image processing systems that use video cameras for control navigation, as those used in cars or planes control, is that these systems depend on image contrast and the environmental pollution as fog, smog or rain. These environment characteristics, filters wavelengths of the light, which causes that the captured images, were modified by the video camera, decreasing its efficiency. It is proposed to improve the chromatic content of captured images, where environmental pollution is present, using the Retinex model. This algorithm implementation uses different characteristics such as lightness changes and color contrast; these characteristics produce different results for every Retinex model proposed showing differences in color and luminance modification of the captured image. In this paper are proposed and compared three different Retinex models; these models are the Simple Retinex, the Multi-scale Retinex and the Multi-scale Retinex with Color Correction.

Keywords: Retinex, contrast, lightness, color

1 Introduction

To be able to recognize license plates, face recognition, navigation systems, etc., it is important to have an acceptable visibility Rank, so the different components of an image could be observed. However, in real conditions, the clearness of the captured image is variable, where the conditions of the atmospheric air were pure, unfortunately the quality of the air changes from one place to another, and from one time moment to another, even if it is taken in the same place. Captured images in extreme conditions, where no visibility exist, because different conditions like dense fog, huge dust content caused by combustion materials, remains suspended in low layers of atmosphere, and does not allow maximum visibility.

The Retinex model, proposed by Dr. E. Land, has the characteristic to emulate the Human Vision System (HVS) behavior, and reproduce the color constancy phenome-

non also called chromatic adaptation. Because of this phenomenon, is possible to identify that the actual color of an object does not affect the lightning color source [1], for example, if a space is illuminated for a tungsten filament bulb, the light that illuminates the object is warm light, however the objects inside the illuminated space retain their color balance. The similar behavior happen if the space is illuminated using a cold light source like green, blue, etc. This phenomenon is described by the Dr. Edwin H. Land. His research investigation explains how HVS understand colors and still today is kept under scientific research with many applications in fields like medical radiography, underwater photography, forensic photography , etc. [2].

In this research work, are used the methods of the Simple Retinex, the Multi-scale Retinex and the Multi-scale Retinex with Color Correction algorithms to prove and compare them as an improvement chromaticity method of the captured images.

2 Retinex Models

Were extracted the images used to compare different implementations of the Retinex algorithms from different digitizing media, like photographic cameras, cellphones or video cameras, all of this using weather environmental control.

Retinex models use mathematic operators resembling the color constancy phenomenon of the HVS. Color constancy is a desirable phenomenon in digitizing images, in order to identify the actual object color no matters lightning source characteristics. For this, there are different ways that improve the Retinex model [3].

The Retinex models receive as input, the magnitude values of the Red, Green and Blue channels to make an estimation of reflectance for each one, this has the purpose of being able to identify the lighting source, which allows knowing the intensity present in each pixel of the digitized image.

2.1 Color Constancy

Color constancy is a mechanism that allows HVS to identify the actual color of an object no matters the lightning source color, for example, from photometric point of view, a red object under a green light source, should produce the same spectral distribution as a green object under a red light source. Even in this condition, the object color remains stable regardless light source wavelength changes [4].

This phenomenon separates the light reflectance from light source, which can define the spectral illumination characteristics and makes an approximation of the reflectance value. The color constancy seen from HVS perception use space and chromatic resolution of each pixel to define scene appearance [5]. The following explains each Retinex applied suggested models.

2.2 Simple Retinex

Simple Retinex algorithm improves digitized captured images in low lightness conditions, using the equation (1). This algorithm imitates the chromaticity and local adaptability lightness for each pixel to get an approach of actual colors in a scene.

$$L = 1/N \sum_{i=1}^N (\log I(x_p) - \log(I(x_i))), \quad (1)$$

where L represents the lightness of a pixel (x_p), influenced by N pixels (x_i), where $i=1,2,3\dots,N$, which will be selected in a random path. This process applies for every color channel of the RGB color space [6].

2.3 Multi-scale Retinex (MSR)

This model allows color identification in a more dynamic way because of the algorithm reduces abrupt illumination changes. The MSR model proposed by D. Jobson [4] is defined by the equation (2),

$$R_{MSR_i} = \sum_{n=1}^N \omega_n R_{n_i} = \sum_{n=1}^M \omega_n [\log I_i(x, y) - \log(F_n(x, y) * I_i(x, y))], \quad (2)$$

where $I = R, G,$ and B channels. M is the scales number, ω_n is a scale weight associated to an involving function $F_n(x, y)$ (Eq. (3)), $*$ express convolution, (x, y) are the coordinates of the pixels, I_i is the actual image and is computed using the equation (4), R_{MSR_i} is the output of the process,

$$F_n(x, y) = C_n \exp[-(x^2 + y^2)/2\sigma_k^2], \quad (3)$$

$$I_i(x, y) = S_i(x, y)r_i(x, y), \quad (4)$$

where σ_k is the typical standard deviation of the Gaussian envelope and its magnitude controls the envelope extension and the whole function is normalized as C_n such that $\int F(x, y) dx dy = 1$, S_i represents the illumination and r_i is the scene reflectance.

2.4 Multi-scale with Color Correction (MSRCR)

The color theory investigation has generated different opinions about the nature of the color image components, one of this is known as –grey world-, this theory assumes that the average value of color variations that comprises the image should average a gray tone in common [7]. In images where is found a dominant color, the MSRCR method can give a result of gray image due to low saturation color. To correct the dominant color delivered for the MSRCR method, modifies the output of the MSR model, multiplying the dominant color for an image chromaticity restoration function [4]. The first step is to calculate chromaticity as shown in equation (5):

$$I'_i(x, y) = \frac{I_i(x, y)}{\sum_{j=1}^S I_j(x, y)}, \quad (5)$$

where i represents the respective channel, S is the channel's number used by the image, generally $S=3$ in the space color RGB. The equation (6) describes Multi-scale with color correction algorithm:

$$R_{MSRCR_i}(x, y) = C_i(x, y)R_{MSR_i}(x, y), \quad (6)$$

where $C_i(x, y)$ is defined in equation (7):

$$C_i(x, y) = f[I'_i(x, y)]. \quad (7)$$

The MSRCR provides the necessary color restoration, eliminating the color distortions and gray zones in the MSR output.

3 CIELab Color Space Evaluation Method

The evaluation methods used to measure image quality are closely related, in comparison to the original version or under some ideal image criteria proposed in the literature; the image distortion quantification is required in many image-processing fields. Color quantification incorporates psycho-physic elements related to human perception becoming more complex in a computing evaluation. However, the evaluation of the human vision system criteria has better results compared with image distortion evaluations [8].

Is used the CIELab color space to evaluate Retinex models, mainly because of its perception uniformity. CIELab Color space sets that for observe colors, they must be observed under a background going from white to grey with an standard D_{65} type illumination source which has similar features with the mid-day light, and with a temperature of 6,504 °K [8]. Every source with these features is called D_{65} . CIELab is the chromatic model used for color description in the human eye perception. This model is represented by three parameters, color lightness (L), position between red and green color (a), and the position between yellow and blue color (b) [9]. Vectors a and b are calculated as shown in equation (8) [8]:

$$\begin{aligned} a &= [C_1 - C_2/11] = [R_a - 12G_a/11 + B_a/11], \\ b &= (1/2)[C_2 - C_1 + C_1 - C_3]/(4.5) = (1/9)[R_a + C_2 + 2B_a], \end{aligned} \quad (8)$$

The CIELab chromaticity vector is represented by the magnitude of the distance between vectors a and b . This indicates that for a high chromaticity value, the image color is more intense or saturated. Therefore, the average chromaticity present in the image processed will be better [10]. The average value of every one of the pixels where chromaticity were computed, represents a vector computed in equation (9):

$$C_{MN} = \sum \sqrt{a_{xy}^2 + b_{xy}^2} / MN. \quad (9)$$

4 Results

In the subjective criteria results obtained in the original image as environmental pollution produces gray areas and image processing with Simple Retinex and Multi-scale algorithms, the light source is identified and can be corrected only lighting, while the algorithm MSRCR, corrected lighting and color correction and thus can have better visibility through air pollution.

To do an image evaluation, is required a transformation in the RGB to CIELab color space, and the average chromaticity of the image is calculated. The best result is indicated by the highest value obtained.

In the Fig. 1 can be saw the objective and subjective results for each image with environmental pollution, being processed with the Single Retinex, MSR, and MSRCR models. The chromaticity computing evaluation for each image is performed using

equation (8), the first step is to transform the RGB color space into the CIELAB color space, and use the a and b CIELab components, for every pixel, to obtain the average values agree to the number of values in a $M \times N$ image dimensions (M, N), where M is the number of rows and N is the number of columns.

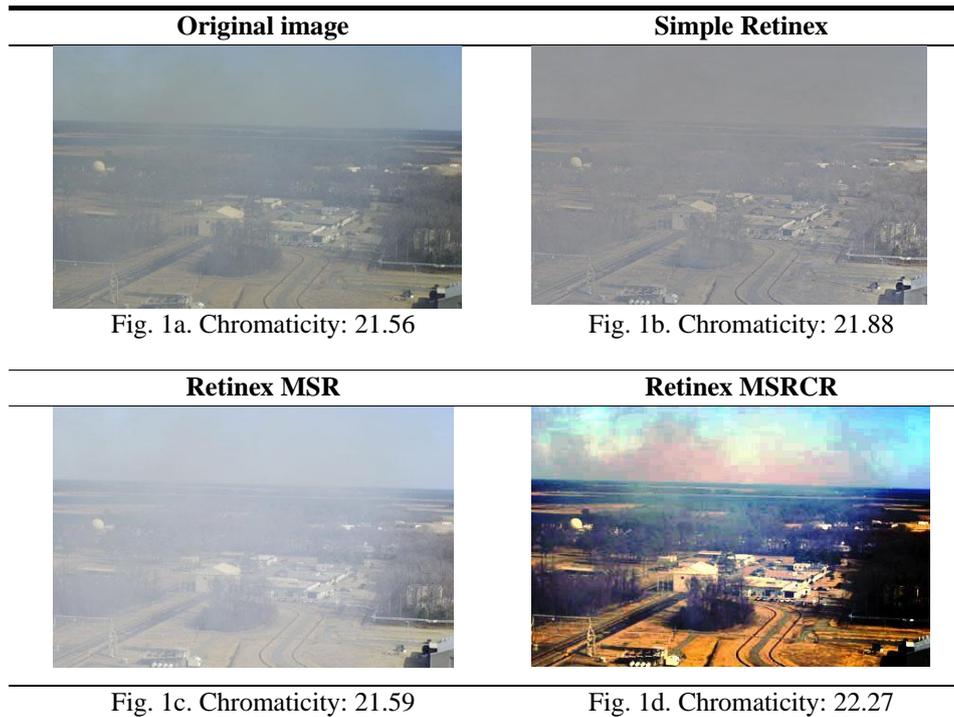
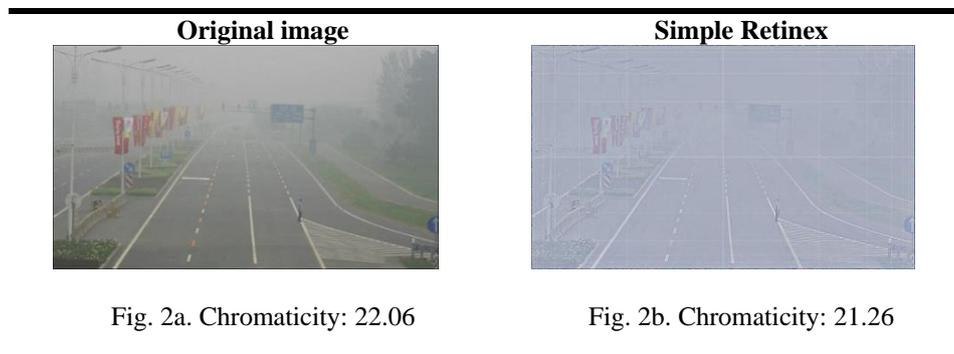


Fig. 1. Fig. 1a was taken with pollution, the other ones increases in chromaticity intensities as shown in Figs. 1b and 1c, where the best result is given by the analyzed MSRCR algorithm shown in Figure 1d.



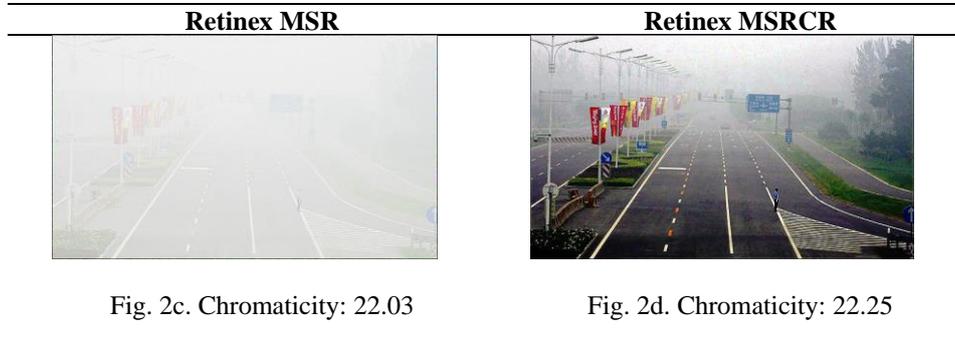


Fig. 2. Fig. 2a was taken with foggy, and it was processed by the Simple and multiscale Retinex algorithms (2b and 2c), it reduces the chromaticity levels of the image, contrary happens with the Retinex MSRCR where chromaticity values increases.

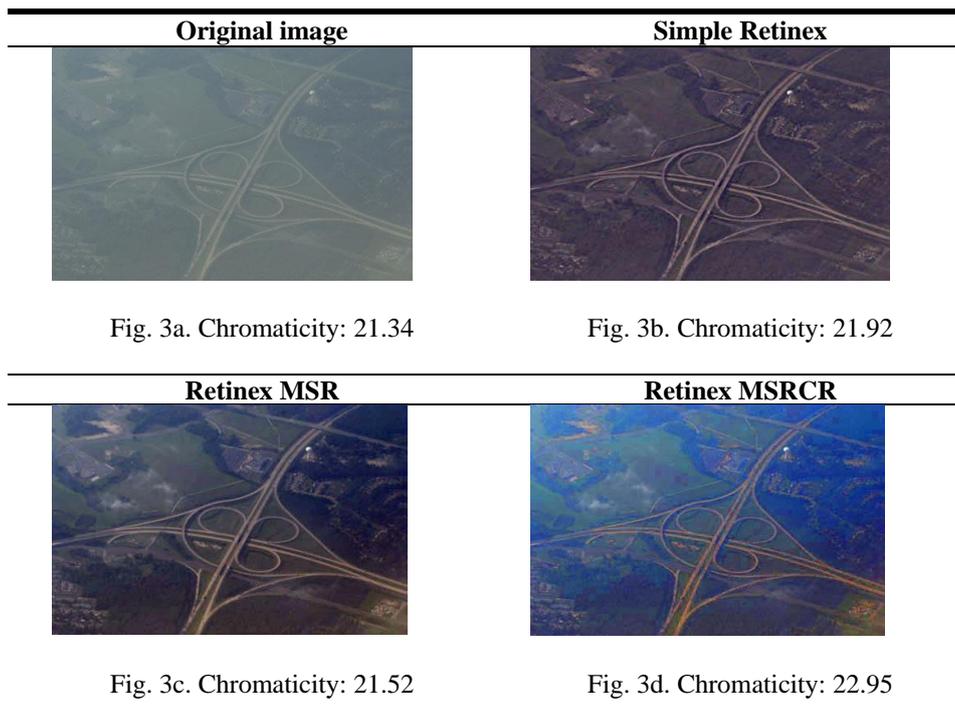


Fig. 3. Fig. 3a is an aerial photograph taken with pollution, and the image was processed by the Simple and multiscale Retinex algorithms, highest image chroma is processed by the MSRCR algorithm.

Original image



Figure 4a. Chromaticity: 22.34

Simple Retinex



Figure 4b. Chromaticity: 20.99

Retinex MSR



Figure 4c. Chromaticity: 21.95

Retinex MSRCR



Figure 4d. Chromaticity: 22.59

Fig. 4. Fig. 4a was taken with pollution, processing results with Simple and MSR Retinex algorithms not improve the content of chromaticity, in Fig. 4d is observed the improved chromaticity content.

It is observed that the values obtained from the color evaluation methods, the Retinex model is efficient in chromaticity correction having better results in contrasting colors, these images corrupted with foggy and pollution are characterized as images affected in their pitch illuminant and wavelength content to capture air pollution.

5 Conclusions

To improve chromaticity content of the images captured in environments with air pollution, the Retinex model can be a tool that will serve to make corrections in the contrast of the images. In the results we can see that the efficiency of the Retinex model is directly related to the content of wavelengths that could be captured in digital imaging and because these vary the results that may be misinterpreted to be processed with a particular Retinex method.

Retinex can be applied to images with pollution as a tool that makes the correction of contrast colors, the performance of Retinex depend on the density of pollution, causing the captured colors do not match the colors obtained under a type lighting D_{65} .

Acknowledgements. The authors would thank to the Instituto Politécnico Nacional de México and the CONACyT for their support in the realization of this research work.

References

1. E. H. Land, J. J. McCann: Lightness and Retinex Theory. *Journal of the Optical Society of America* (1971)
2. G. Hines, U. Rahman, G. Woodell: Single-Scale Retinex Using Digital Signal Processor. *Global Signal Processing Expo (GSPx)* (2004)
3. J. M. Morel, A. B. Petro, C. Sbert: Fast implementation of color constancy algorithms. *SPIE*, vol. 7241 (2009)
4. D. J. Jobson, Z. Rahman: A multiscale Retinex for Bridging the gap between color images and the human observation of scenes. *IEEE Transactions on image processing*, n° 7 (1997)
5. A. Rizzi, C. Gatta: From Retinex to Automatic Color Equalization: issues in developing a new algorithm for unsupervised color equalization. *Journal of Electronic Imaging* (2004)
6. J. McCann, B. Funt: Retinex in MATLAB. *Journal of Electronic Imaging*, n° 13, (2004)
7. G. Buchsbaum: A spatial processor model for object colour perception. *Journal of the Franklin Institute* (1980)
8. X. Zhang, B. A. Wandell: Colour image fidelity metrics evaluated using image distortion maps. *Imaging Science and Technology Program, Department of Psychology* (1998)
9. J. Schanda: *Colorimetry: Understanding the CIE System*. Wiley-Interscience (2007)
10. M. D. Fairchild: *Color Appearance Models*. Second Edition, John Wiley & Sons, Ltd (2005)
11. V. Tsagaris, G. Ghirstoulas, V. Anastassopoulos: A Measure for Evaluation of the Information Content in color images. *IEEE Inter. Conf. Imag. Process.* (2005)

Face Recognition based Only on Eyes' Information and Local Binary Pattern

Francisco Rosario-Verde, Joel Perez-Siles, Luis Aviles-Brito, Jesus Olivares-Mercado, Karina Toscano-Medina, and Hector Perez-Meana

Instituto Politécnico Nacional, Mexico City,
Mexico

jolivares@ipn.mx

Abstract. In this paper the implementation of the Local Binary Pattern algorithm for face recognition is presented using the partial information of the face, the main contribution of this work is that segmenting the parts of face (forehead, eyes, mouth) can make the recognition a person using only their eyes and getting a percentage of up to 69%, which considering the limited information provided a good success rate is obtained. In the test phase AR facedatabase it was used and using the method of Viola Jones face is located and segmented to obtain templates for each person and each part of his face and Euclidean distance was used for classification task. Because in a real application do not always have all the face of the person to identify the proposed system shows that you can get good results with partial information about it, in addition the results show that in the ranking 6 always provided the right person, which is also useful in real applications.

Keywords: Face recognition, partial information, local binary pattern, eyes information

1 Introduction

Nowadays insecurity is observed in restricted areas such as banks, shopping centers, airports, etc. Therefore it is necessary have a strict control of the persons that enter to this kind of places, and thus know if the people belong to it or not. This requires recognizing people without invading your privacy; this is done by a biometric analysis.

Biometrics is a responsible discipline of automatic recognition of persons through their physical features (face, retina, iris, voice, fingerprint, etc.) or behavioral traits (gait, writing, etc.). For this, the system uses a facial recognition, but if it is a system implemented in an uncontrolled environment take place to problems of occlusion of the face like lighting conditions at the time of image capture, plus, the face to recognize may contain facial hair, makeup and accessories such as sunglasses, caps, hats or scarves. The recognition of facial images allows determining the identity of a person, when you compare a picture

of this face with reference images stored in a database that contains the regions of interest taken in a controlled environment.

This work proposes a solution to the problem of face recognition using information from the eyes, through the algorithms: Viola-Jones, segmentation of the face for the extraction of features (LBP), system training through the overlap of images LBP and finally using Euclidean distance for the comparison with the database in order to carry out the identification of the person.

2 Methodology

The block diagram of proposed system is based on a series of processes as shown in Fig. 1, where each block is described below.

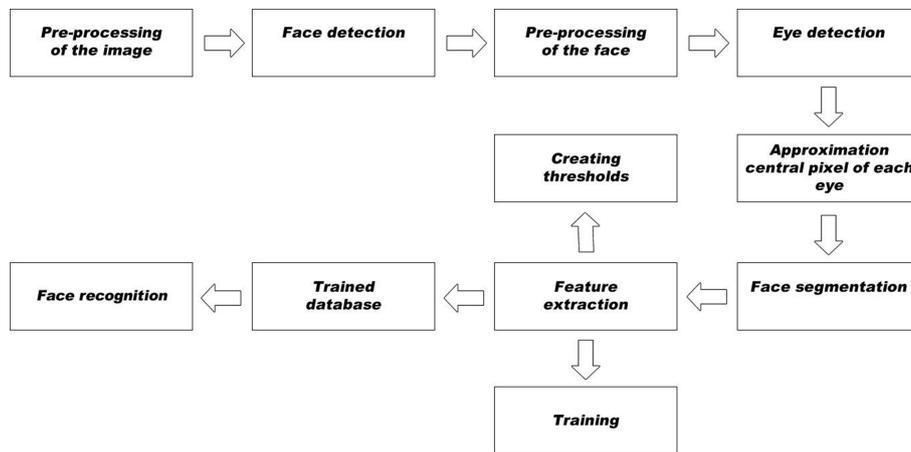


Fig. 1. Block diagram of the system

2.1 Face Detection

To implement the face detection was used the Viola-Jones algorithm which uses a classifier in cascade through simple descriptors called "Haar", which can be calculated efficiently using a representation of the intermediate image call it integral image [6]. If a face is not found in the original image, it is carried out a pre-processing of the image, in order to have a higher rate of detection of face. Fig. 2 shows the stage of pre-processing and possible processing of the input image to face detection. In the first instance is the original image (a), (b) is the result of the pre-processing of the input image using the adaptive histogram equalization to contrast limited (CLAHE) [4] and in subsection (c) the results of the possible processing if and only if, in the stage of pre-processing it will not detect any face in the input image.



Fig. 2. Image Processing.

(a) Original Image. (b) Pre-processing image. (C) Further processing in the event of failing to detect a face in the first iteration.

2.2 Eyes Detection

The eyes are detected through the toolbox Face Parts Detection [5], which through an algorithm in cascade finds the eyes and also in a variable stores the coordinates where start the eyes and the pixels that have long and wide for each one of them.

The eyes detection of this toolbox is very effective, however, when the faces have eyeglasses, problems of lighting, contrast and even if the face is slightly inclined, it can submit detection problems that affect the facial segmentation algorithm, which seeks a eye detection as more closely aligned as possible. To improve eyes detection, a processing was done of the face in the same way as was done in the face detection described above.

2.3 Approximation of the Center Pixel of the Eyes

The facial segmentation algorithm is based on the distance that exists between the center of the eyes, therefore, was proposed a method to obtain the approximation of the center pixel of each eye. Once detected the eyes in the face as shown in Fig. 3, Face Parts Detection returns 4 values for each eye detected, the starting coordinates (C_i) and the dimensions of the area of detection (DAD).

C_i and DAD are denoted by:

$$C_i = (X_1, Y_1), \quad (1)$$

$$DAD = (width, height), \quad (2)$$

with the previous data is make the approximation of the coordinates of the center pixel ($Pc = (X_{pc}, Y_{pc})$) of each eye using the following equations:

$$X_{pc} = X_1 + (width/2), \quad (3)$$

$$Y_{pc} = Y_1 + (height/2). \quad (4)$$

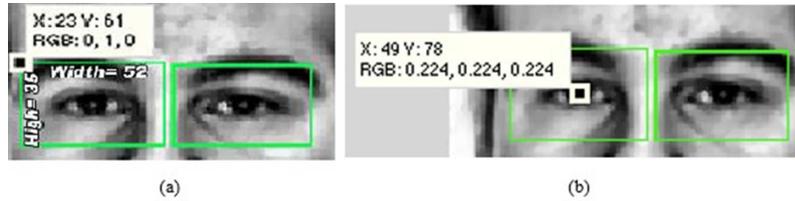


Fig. 3. Results of approximation of the center pixel. (a) data generated by Face Parts Detection. (b) center Pixel obtained.

In case that not finding the eyes in the previous stage, the system gives the user the choice to get manually the location of the center of each eye, subsequently it take the coordinates of the pixels and based on them segment the face in their parties more significant.

2.4 Face Segmentation

The implementation of the segmentation algorithm is based on the proportions of the face that are calculated from the distance of the center between the eyes, which are given by the points (X_1, Y_1) and (X_2, Y_2) , respectively. Coordinates X_1, Y_1, X_2 , and Y_2 are shown graphically in Figure 4.

The distance between the eyes, DO , is given by:

$$DO = X_2 - X_1. \quad (5)$$

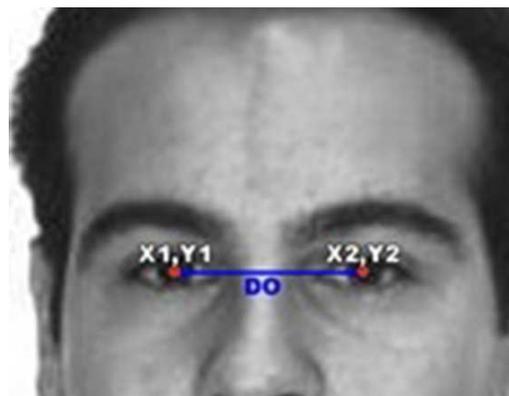


Fig. 4. Eyes coordinates.

The segmentation algorithm that was used gets 8 regions of interest (mouth, forehead, eyes, nose, left eye, right eye, left half and right half of the face), which are obtained with different proportions, in addition that they are all related to the distance from the center of the eyes (equation 5). The segmentation results obtained are similar to those proposed by Zisheng et al. [1], where the face is segmented into 4 regions. Taking as reference the previous procedure Table 1 shows the proportions for each region of interest; in this work only took the regions of the eyes.

Table 1. Proportions of the face

| Region | Coordinate in x | Coordinate in y | High | Width |
|-----------|--------------------|--------------------|------------|----------|
| Eyes | $X_1 - 0.5 * DO$ | $Y_1 - (0.5 * DO)$ | $0.8 * ED$ | $2 * DO$ |
| Left Eye | $X_1 - (0.5 * DO)$ | $Y_1 - (0.5 * DO)$ | $0.8 * DO$ | DO |
| Right Eye | $X_1 + (0.5 * DO)$ | $Y_1 - (0.5 * DO)$ | $0.8 * DO$ | DO |

2.5 Feature Extraction

The LBP algorithm introduced by Ojala et al. [3] is one of the most efficient methods for describing texture. The original LBP method, that is, the hLBPH, uses masks of 3 x 3 pixels, called the "texture spectrum", to represent a neighborhood around a central pixel, as shown in Figure 5(a), where the values of the neighboring pixels are compared with the central pixel, taking that pixel value as the threshold. Pixels are labeled as 0 if values are smaller than the threshold; otherwise, they are labeled as 1, as shown in Figure 5(b). Next, the pixel labels are multiplied by 2^p , where $0 \leq p \leq 7$ is the position of each pixel in the neighborhood, as shown in Figure 5(c). Finally, the resulting values are added to obtain the label of the central pixel in that neighborhood, yielding Figure 5(d). This method produces 128 possible values for the central pixel label. This process is repeated for the entire image, producing a LBP labeling matrix (with the same size as the input images), which is used to estimate the vector for the face image features.

2.6 Training System

The training was done with only three faces already processed with the LBP algorithm; in order not to lose characteristics between each overlay face. The three images will be known as the original image (IO), image to join (UI) and the resulting image (IR). To generate the template we have to reduce to half the values of the arrays of each face in order to obtain an average that will be the resulting image (IR), that is to say:

$$IO = IO * \frac{1}{2}, \quad (6)$$

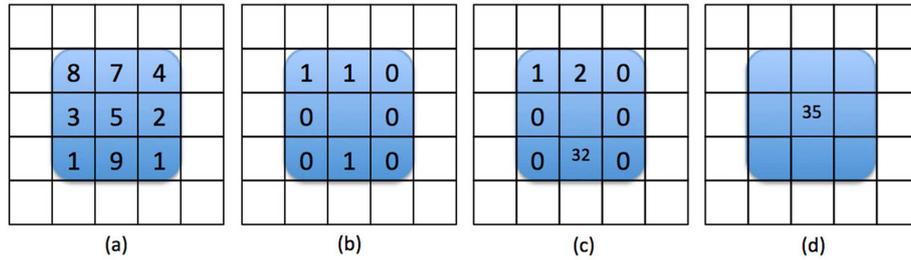


Fig. 5. LBP neighborhood example.

$$IU = IU * \frac{1}{2}, \quad (7)$$

$$IR = IO + IA. \quad (8)$$

For the next image is performed the same procedure, only that now we multiply to go by $\frac{1}{2}$ as well as the next image to join (IU_n), in order to obtain the image template (IP) of that face.

$$IP = (IR * \frac{1}{2}) + (IU * \frac{1}{2}). \quad (9)$$

Giving as a result the image template (IP) shown in Fig. 6.



Fig. 6. Final Template.

3 Results

To obtain the results is calculated the euclidean distance to determine which person is the winner. Two different tests were performed, the first of these

is the facial verification which consists of two stages, the first consists in the face detection, to perform the segmentation and getting features of the regions of interest with LBP, in the second stage we take a sample of 5 persons, of which it was obtained the segmentation and characteristics of each one to carry out a comparison between these regions and the templates produced with their corresponding thresholds, where, if any region of the face is in the range of the threshold will accept that image and give by the fact that the individual has similarity with the person to recognize.

Table 2 shows the results obtained when performing the verification using the left eye and right eye as well as both eyes, in the temporary results tells us what percentage of the images entered in the thresholds, that is, at least some of the regions of interest used are accepted in the verification. In the column of final results will have to perform a confirmation, which is that, at least two of the regions of interest come in the thresholds.

Table 2. Verification Result.

| | Temporary Results | Final Results | False positives |
|----------|-------------------|---------------|-----------------|
| Person 1 | 80% | 45% | 36% |
| Person 2 | 73.68% | 52.63% | 18.18% |
| Person 3 | 73.68% | 47.36% | 0% |
| Person 4 | 84.21% | 68.42% | 0% |
| Person 5 | 65% | 50% | 0% |
| Average | 75.31% | 52.58% | 10.96% |

The other test was the facial recognition classified in a ranking 1, 3 and 6, this test was conducted with a total of 20 people, and each person has around 20 images to 25 images. The ranking is the probability that the person linked to the image under analysis is within a group of N people which have the lower values in the Euclidean distance, not importing the image to be associated with the wrong person, it is important that the correct person is within this group N [2].

For the tests as well as in verification, we used the regions of the eyes as a whole and separately, in Fig. 7, shows the graph of the percentage of the number of images that were within the respective ranking for the left eye. Fig. 8 shows the graph of the percentage of the number of images that were within the respective ranking for the right eye.

Fig. 9 shows the graph of the percentage of the number of images that were within the respective ranking for both eyes, in which it was obtained the highest percentage in ranking 6 since it reached the 94%.

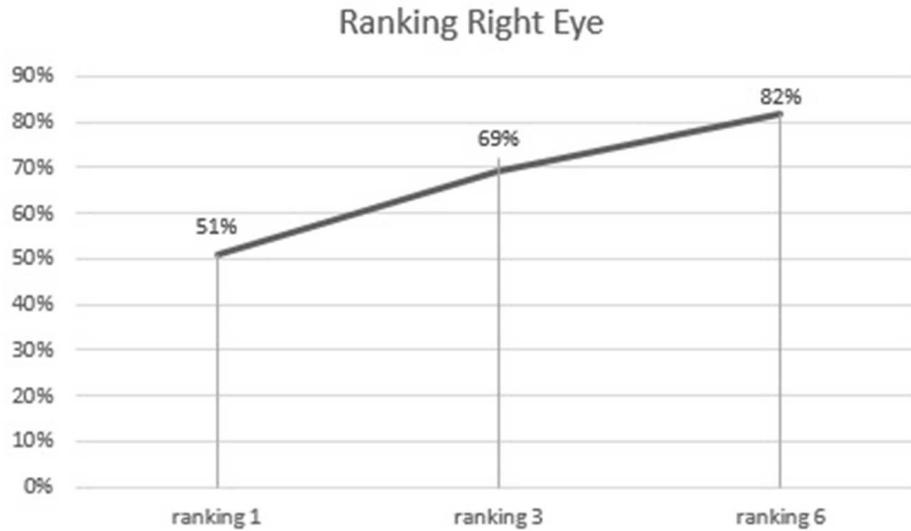


Fig. 7. Ranking Right Eye.

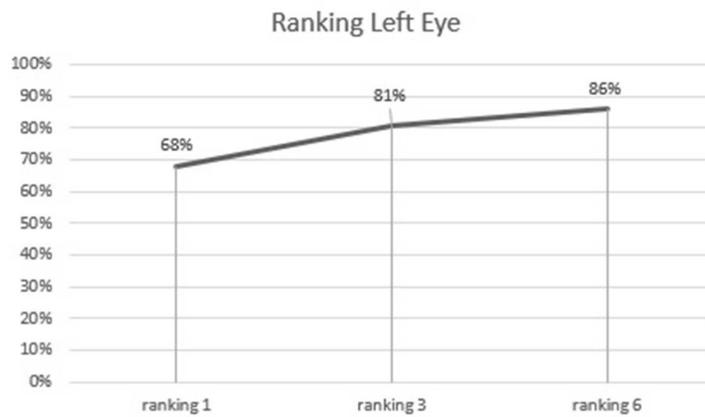


Fig. 8. Ranking Left Eye.

4 Conclusions

This investigation culminated successfully, the goals are met to carry out the recognition of face without being invasive, using only one eye either left, right or both of the person to recognize versus the database of templates, coupled with the effectiveness of the proposed algorithms for processing, detect, segment,

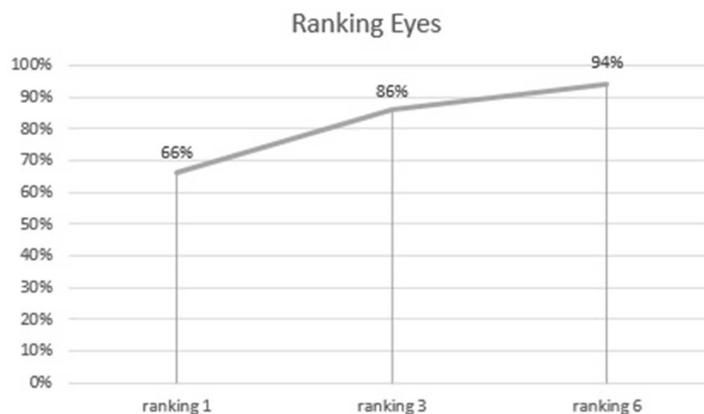


Fig. 9. Ranking Eyes.

extract features, among others, achieving good results in the different tests of ranking: 1 with 66%, 3 with 86% and highlighting ranking 6 where it was obtained up to 94% of recognition of the person of interest using both eyes, that is, the face to recognize it is part of the group of 6 people with faces that share similar characteristics, being more easy to detect who is the person of interest. For the tests as well as in the verification, we used the regions of the eyes as a whole and separately, in the Fig 6,7,8 show the graphs of the percentage of the number of images that were within the respective ranking for the left eye, right eye, and both eyes, respectively.

Acknowledgement. We thank the National Science and Technology Council of Mexico and to the National Polytechnic Institute for the financial support during the realization of this work.

References

1. Li, Z., Imai, J., Kaneko, M.: Facial-component-based bag of words and phog descriptor for facial expression recognition. In: Systems, Man and Cybernetics, 2009. SMC 2009. IEEE International Conference on. pp. 1353–1358 (Oct 2009)
2. Mathworks: Computer Vision System Toolbox. <http://www.mathworks.com/help/vision/index.html>
3. Ojala, T., Pietikainen, M., Harwood, D.: Performance evaluation of texture measures with classification based on kullback discrimination of distributions. In: Pattern Recognition, 1994. Vol. 1 - Conference A: Computer Vision and Image Processing., Proceedings of the 12th IAPR International Conference on. vol. 1, pp. 582–585 vol.1 (1994)

Francisco Rosario-Verde, Joel Perez-Siles, Luis Aviles-Brito, Jesus Olivares-Mercado, et al.

4. Reza, A.M.: Realization of the contrast limited adaptive histogram equalization (clahe) for real-time image enhancement. *Journal of VLSI signal processing systems for signal, image and video technology* 38(1), 35–44 (2004)
5. Tanaka, M.: Face Parts Detection. <http://like.silk.to/matlab/detectFaceParts.html> (2014)
6. Viola, P., Jones, M.: Robust real-time object detection. *International Journal of Computer Vision* 4, 51–52 (2001)

Scene Dedicated Feature Descriptor with Random Forest Training for Better Augmented Reality Registration

Andras Takacs, Edgar A. Rivas-Araiza, and Jesus Carlos Pedraza-Ortega

Universidad Autónoma de Querétaro, Querétaro,
Mexico

Abstract. The most important part of an Augmented Reality system is the tracking system to support an accurate and robust registration. In outdoor environments, the continuously changing environmental characteristics and elements make hard to achieve this tracking process. The main point of this operation is that the descriptor has to work with great accuracy in all kind of situations. The most used descriptors have this distinctive capacity, but computers and mobile devices process them in a long time frame. This paper investigates a new trained, lighter, scene dedicated descriptor, which takes into account the scene characteristics. The descriptor is loaded with elements that can be computed faster and have distinctive information about the selected area. The complete descriptor is used for semantical feature extraction with the aid of a trained Random Forest classifier. For validation purposes, the descriptor was tested against the most used descriptors and in some cases it proved to be faster and equally reliable.

Keywords: Augmented reality, descriptors, random forest classifier

1 Introduction

In recent years, the use of Augmented Reality (AR) has been steadily growing. The stability of the AR applications is continuously improving, but in outdoor environments has lots of flaws due to the rapidly changing environmental factors and the mobile devices still limited storage capacity and processing power. The main challenge is to create a light and robust application for outdoor environments. The use of facade recognition and segmentation with a trained, environment dedicated descriptor is a possible way to stabilize these applications in built-up areas.

1.1 Related Work

In order to support digital data with real scenery it needs to be solved against other two crucial problems; Tracking and Registration. Tracking is the method how AR system specifies its position in the 3D environment and it is crucial

for the stable Registration. For this in outdoor AR different techniques were developed along the years, to deal with the changing light conditions, markerless environment and sparse areas among others. At the beginning of the last decade mainly magnetic sensors were used, such as the gyroscope, GPS, accelerometer or compass [2]. The development in the field of computer and mobile processing units facilitated to exploit more the video see-through AR applications with image processing functions. By the end of the decade even though the magnetic sensors were exploited, by using the gravitational force to get better tracking and registration [15, 16] or tracking the GPS position and cloud server for more stable outdoor tracking [22], the tracking is mainly done through the camera tracking environment features, or artificial markers. Computing frame by frame the spatial positions of this features specifies the systems status. In recent years in order to improve the video see-through AR tracking, the use of geo-tagged panoramic images increases the performance of AR systems [1]. However, those approaches are using a cloud-based processing unit to help the tracking system with a dependency to an internet connection. Also, there is a proposal [13] of a method that uses Random Forest to get a better feature tracking for PTAM [14]. However, they state that their system manages only about 650 scenes and both learning and recognition processes are implemented in online fashion.

According to [12] the automatic facade techniques were a response to the growing need of mass 3D reconstruction and modelling in city planning, geo-applications like "Google Earth" or "Microsoft Virtual Earth" and in 3D GPS navigation systems to reduce the reconstruction time and the storage size of the data. There are various techniques which were developed during the years, and there are differences not only in the technique used to extract features but the source data used also differs. Some researchers use input from terrestrial laser scanner like [18] where the features are detected from the point density, others use a mixed source, they gather the information from the laser scanner and images simultaneously for the reconstruction [4]. More techniques exist for reconstruction from images, by using different approaches to obtain necessary information. Also, there is the "bag of key point" method, which uses a general image categorization technique [10]. This method uses low-level SIFT image features as descriptors assigned to high-level image clusters called "vocabularies" for training a multi-class classifier. In [5] it is presented a technique for image parsing of architectural scenes. This is achieved via segmenting the images into visually recognizable regions (sky, foliage, building and street). Moreover, [11] developed further the technique of [10] using Opponent SIFT as descriptor and Randomized Decision Forest as classifier achieving a faster classifier than its predecessor.

1.2 Contribution

This paper presents the results a comprehensive performance evaluation of a specialized feature descriptor in terms of both computational efficiency and retrieval performance. The main purpose is to show that a specialized feature descriptor can produce better results in terms of performance and can be as efficient as the

state of the art feature descriptors. The basic concept is to create lightweight descriptors with elements that specialized to the corresponding environment (buildings, green areas, and sparse areas) after training and empower them with machine learning techniques to be stable in the mentioned areas. The further goal is to create a new modular descriptor where the system can automatically detect the scenery and decide the composition of the descriptor.

1.3 Organization of this paper

Firstly the Random Forest classifier will be presented which will be followed by the state of the art descriptors which were used for the evaluation of dedicated descriptor. The second part of the paper will begin with an overview of the experimental setup, in Section V the results will be presented will finish with the Conclusion in Section VI.

2 Random Forest

The Random Forest [7] is a high-performance discriminative classifier, handling a large set of features without having difficulties due to the curse of dimensionality [11]. It is a supervised learning method that construct an ensemble of recursively created random binary decision trees (Fig.1) during the training period and learn more than one class at a time. In the classification process it returns the most voted class given a feature vector v_i by averaging the final probabilities p_τ of each tree.

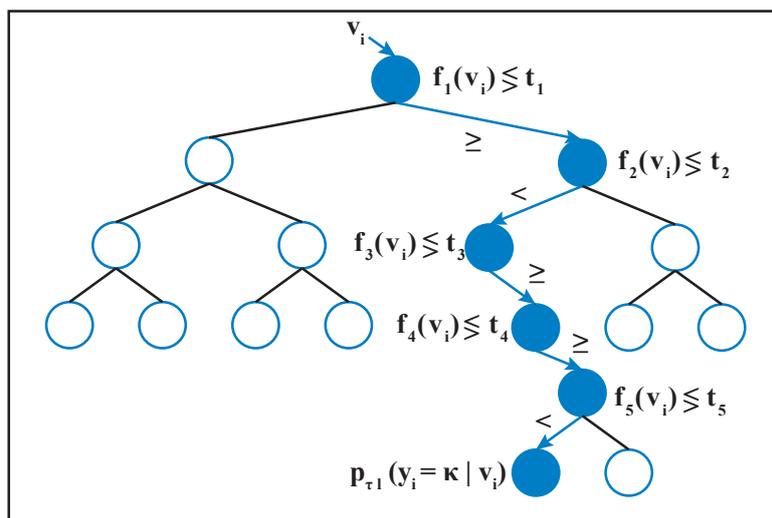


Fig. 1: Binary decision tree.

$$p(y_i = \kappa|v_i) = \frac{1}{T} \sum_{\tau=1}^T p_{\tau}(y_i = \kappa|v_i). \quad (1)$$

The strength of this process, compared to the random decision trees which may suffer from overfitting, that it has been aggregated randomly at two stages during the building of the forest in the training session. First at the Bootstrap Aggregation [7] where random subsets of data are created and from which the trees are learned, and second during the creation of the decision trees at the split functions using only a random fraction of all features.

3 The Descriptors

During the development, the two most used descriptor and their colour versions were tested for their characteristics in order to investigate the speed and accuracy performance of the dedicated descriptor.

3.1 State of the Art Descriptors

Scale Invariant Feature Transform (SIFT) Has been for the past 10 years the most used and referenced descriptor with 128 elements, which consist a set of orientation histograms on 4×4 pixel neighbours over a 16×16 region around the key point. The magnitudes are weighted by a Gaussian function afterward [17].

Speeded-Up Robust Features (SURF) It was built by [3] based on SIFT but, according to the authors it has a better performance. It is smaller in size a 64-dimension vector calculated from a squared region centred on the key point. The region is split into 4×4 subregions. They calculate a Gaussian weighted horizontal and vertical Haar wavelet, which are summed over the sub-regions, and also they calculate the absolute values of the same responses.

Opponent SIFT According to [19] this is the best performing SIFT descriptor on coloured images. It is calculated in the same way as the classical SIFT descriptor but for all the opponent color channel, where the color space contains one intensity and two chromatic channels. That adds up to a 384-dimension vector. These highly decorrelated channels were calculated in the following way.

$$\begin{pmatrix} O_1 \\ O_2 \\ O_3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{3}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix}. \quad (2)$$

Opponent SURF To retrieve the color information [9] created this descriptor by calculating the original SURF descriptors on the 3 opponent color spaces, which gives a 192-dimensional vector.

3.2 Environment Dedicated Descriptor

The newly proposed descriptor is an 113-dimension vector computed from a 9×9 patch selected around each key point. The size was chosen to be big enough to pick up edges, low-level changes on the image, and still reduce the saving time and size of data to the forest. The elements were chosen with the following characteristics:

Position - *2 values* - 2D image coordinates of the patch centres to separate points which are on the top (sky), on the bottom (street) and in the middle (facade).

Patch Mean - *6 values* - The mean of the Red, Green, and Blue (from the RGB channels) and Saturation values (from the HSV channels) over the patch are calculated to exploit the color changes on the images. Sine and Cosine of the mean of the Hue values over the patch are also estimated. Because the Hue is angular, it has a discontinuity. The red value at 0° is almost the same as the red at 360° . With the Sine and Cosine pair, we can make them equal.

The Third Order Central Moments - *24 values* - The third order central moments were generated to get distinctive shape description of the patch. The $\mu_{03}, \mu_{30}, \mu_{21}, \mu_{12}$ of the RGB and HSV channels over the patch measure the skew and the symmetry of the point spread around the mean of the patch. Firstly the M_{00} raw image moment calculated by

$$M_{ij} = \sum_x \sum_y x^i y^j I(x, y), \quad (3)$$

then the two components of the centroid:

$$\bar{x} = \frac{M_{10}}{M_{00}}, \quad \bar{y} = \frac{M_{01}}{M_{00}}. \quad (4)$$

Then the third order Central moments then defined as

$$\mu_{pq} = \sum_x \sum_y (x - \bar{x})^p (y - \bar{y})^q f(x, y). \quad (5)$$

The higher order moments describe more fine variations in the shape, but they are more sensitive to noise and left out for that reason.

Distance Transform - *81 values* - Distance transform measures the distance between the pixel and the nearest detected canny edge point. The values of the distance transfer are growing as the point is further away from the edge, in this way the values at the flat areas are at their maximum which is a good distinctive component in the descriptor to help the forest to separate the patches in flat areas from the patches from areas where lots of transition are located.

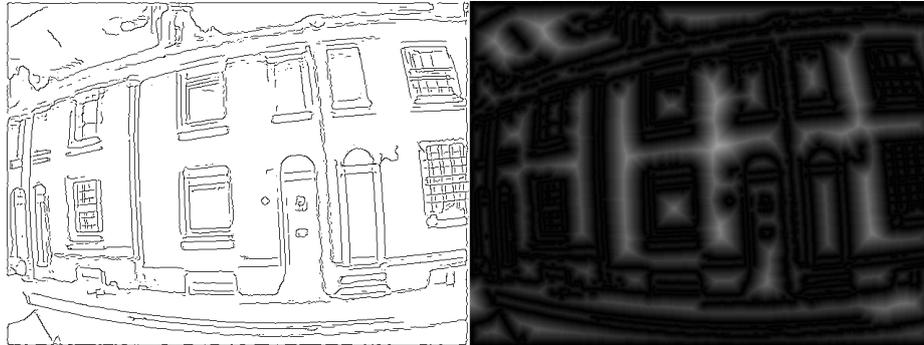


Fig. 2: Distance transform results.

4 Experiments

The steps of the experiment were identical for all the researched descriptors. First the interest points were specified then the descriptor values were calculated. The computed descriptor vectors were loaded to the Random Forest training method. In the final step, the database containing the trained decision trees was used in the classification method where the object features were segmented.

Descriptor Extraction To retrieve all possible information on the image, evenly spaced feature centre points were specified with the same distance to each other. The whole image has been blurred to remove the unnecessary edges and noise, and the Canny edges [8] were calculated for the distance transform image. The resulted images (blurred color and distance transform) was used for to extract the necessary information. The descriptor vectors we created out of the data extracted from the 9×9 patch area around the centre points.

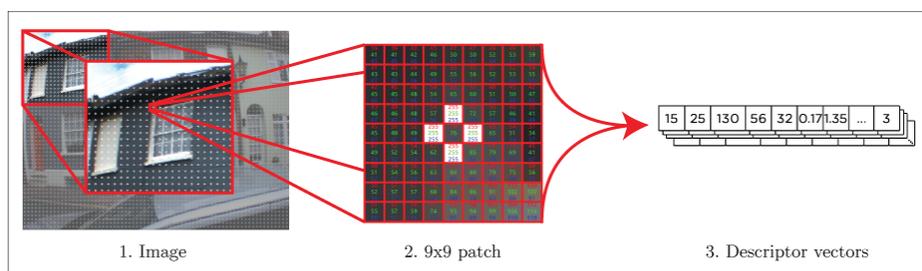


Fig. 3: Descriptor extraction.

Random Forest training The creation of Random forest follows the standard framework developed by [6]. The 90% of the images were used to create for each descriptor a database with 100 trees. At every non-leaf node in each tree, a binary test was assigned which chooses 4 variables in order to find the best split. The tree growing stops when it reached its maximum depth (15) or the maximum number of training images were reached. The forest was saved for the later classification and test.

Classification The final function is a feature segmentation algorithm. The saved trained Random Forest was used for a pixel-by-pixel classification on the test image to detect finer details on the facade. The results of this classification can be observed on Fig. 4. where each color represents a class, the yellow circles belong the wall class, the red the window or door class, the blue circles marking the roof class and the green circles showing the other class. We can corroborate that the descriptor is strong in the door and window areas and produce good results on the roof areas. The result images were turned into a binary image in order to eliminate the fine noise with morphological operations. This step reduces the irregularities around features which could cause a problem in the rectangle fitting (for example it creates a connection with other window areas). After detecting the contours of the segmented area, the algorithm utilizes topological analysis method [20] which counts all the non-zero components and extracts the boundaries on the binary image. To fit the rectangle around the area the Ramer-Douglas-Peucker algorithm (which recursively divides the line between the given first and last point) was used to approximate the polygon enclosed by the previously detected contour using with another polygon with less vertices. After the bounding box is detected, the results are re-projected to the original image.



Fig. 4: Project descriptor classification results.

5 Results

The training and test images were saved from a video recording. It was recorded in two different occasions with different lighting conditions in Queens Gardens street in Brighton, United Kingdom. These videos were stored frame by frame and from this large set of images a dataset was selected. To test and evaluate the performance of the different descriptors, the results of the predictions from each image were compared to the ground truth of the same image. A confusion matrix was set from that information for each descriptor, which was the base for performance evaluation. On Table 1. we can see throughout the testing the Opponent SIFT and SIFT descriptors provide the most reliable performance. Above 70% were the correct detection rate. Interestingly the Opponent SIFT descriptor was designed for a coloured environment, but in the tests it did not give better performance as the simple SIFT descriptor. On the contrary, the descriptor which was designed for greyscale imagery, had a better performance throughout. On the other hand, the performance of the proposed descriptor designed for this project is giving the third best performance throughout the tests, and most importantly in the test where the window bounding rectangles were extracted, the speed of the proposed descriptor called Project is for long the fastest.

Table 1: Total true positives.(%)

| Descriptor | Number of training images | | | | |
|---------------|---------------------------|-------|-------|-------|-------|
| | 9 | 20 | 30 | 40 | 52 |
| SIFT | 74.71 | 77.52 | 73.58 | 71.46 | 72.02 |
| Opponent SIFT | 72.52 | 75.47 | 70.61 | 69.15 | 70.85 |
| SURF | 57.85 | 57.19 | 55.17 | 55.35 | 54.72 |
| Opponent SURF | 57.27 | 56.26 | 55.89 | 55.49 | 55.97 |
| Project | 67.42 | 63.89 | 58.17 | 59.77 | 61.92 |

The Fig 5. shows the details of the precision of each tested descriptor in each segmentation category. The values of true positive points in each category show the level of accuracy of each descriptor. In each class, the Project descriptor was operating with high exactness even in sparse areas like the other areas where the efficiency of the SIFT and Opponent SIFT descriptors gave a poorer performance. In Fig 4. the distribution of classified points are displayed where we can observe the Project descriptor's results.

In Fig 6. it can be observed the time being spent by the computer to reading, calculating the necessary data for the descriptor and based on the outcome of the classification, segmenting the window and door areas. The result data shows that while the most effective SIFT descriptor needed 223 seconds, the Opponent SIFT descriptor for the same work needed three times as much effort in time as in this case the computer has to do the same calculation on the three color channels.

The Opponent SURF descriptor occupied 298 seconds for the work which is three times slower than the Project descriptor and its performance in detection were inferior to this descriptor. These results correlate with the finding of [21] where the SIFT descriptor was more accurate in feature matching but in a considerable longer time frame. Fig 4. shows the final outcome of the segmentation algorithm where it can be noticed the strength of window detection and the weakness of the descriptor vector in terms of distortion and rotation.

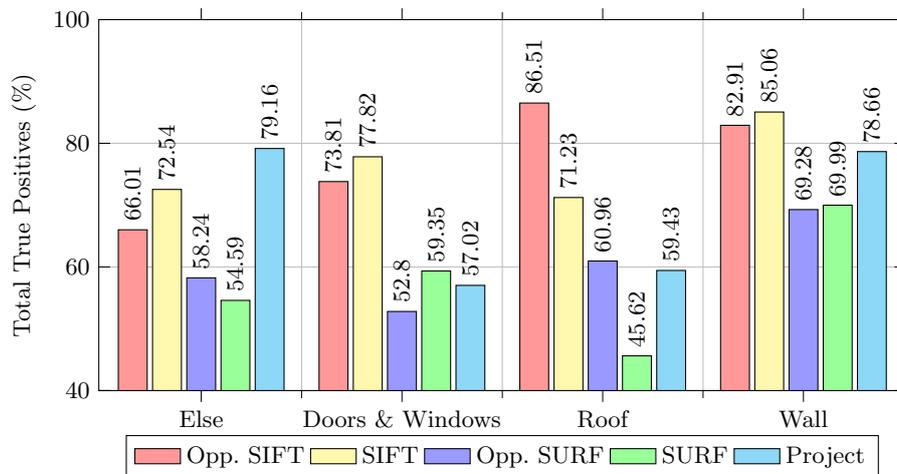


Fig. 5: Correctly qualified point ratio in category groups after training descriptors with 52 images

6 Conclusions

In this paper feature extraction with a new scene dedicated descriptor were studied based on speed and accuracy. The results show that although the mainstream descriptors have reliable performance in detect image features, a descriptor which is created specifically for a certain environment can have similar accuracy but in a shorter time period. This projects a new path to investigate a trained dynamic descriptor which can adjust characteristics of the retrieved information according to the environment. Based on the results it is also planned to stabilize the Project descriptor for rotation, light change and distortion, and to create another version for different environmental characteristics.

Acknowledgement. We would like to thank the Consejo Nacional de Ciencia y Tecnologia through the project number 340519 without whom this paper could not have been completed. Also, I would like to thank the Universidad Autnoma de Quertaro for its facilities and support.

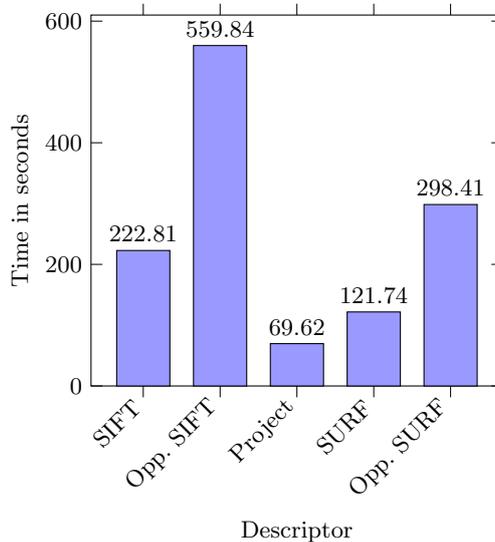


Fig. 6: Descriptor performance in time

References

1. Arth, C., Klopschitz, M., Reitmayr, G., Schmalstieg, D.: Real-time self-localization from panoramic images on mobile devices. In: Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on. pp. 37–46 (2011)
2. Azuma, R., Bailiot, Y., Behringer, R., Feiner, S., Julier, S., MacIntyre, B.: Recent advances in augmented reality. *IEEE Comput. Graph. Appl.* 21(6), 34–47 (2001)
3. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (surf). *Comput. Vis. Image Underst.* 110(3), 346–359 (jun 2008)
4. Becker, S., Haala, N.: Refinement of building facades by integrated processing of lidar and image data. *International Archives of Photogrammetry, Remote Sensing and Spatial Information Science* 36, 7–12 (2007)
5. Berg, A., Grabler, F., Malik, J.: Parsing images of architectural scenes. In: *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*. pp. 1–8 (2007)
6. Breiman, L.: Bagging predictors. *Mach. Learn.* 24(2), 123–140 (aug 1996)
7. Breiman, L.: Random forests. *Mach. Learn.* 45(1), 5–32 (oct 2001)
8. Canny, J.: A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on PAMI-8(6)*, 679–698 (1986)
9. Chu, D.M., Smeulders, A.W.M.: Color invariant surf in discriminative object tracking. In: *ECCV Workshop on Color and Reflectance in Imaging and Computer Vision* (2010)
10. Csurka, G., Dance, C.R., Fan, L., Willamowski, J., Bray, C.: Visual categorization with bags of keypoints. In: *In Workshop on Statistical Learning in Computer Vision, ECCV*. pp. 1–22 (2004)
11. Frohlich, B., Rodner, E., Denzler, J.: A fast approach for pixelwise labeling of facade images. In: *Pattern Recognition (ICPR), 2010 20th International Conference on*. pp. 3029–3032 (2010)

12. Gool, L.V., Zeng, G., den Borre, F.V., Müller, P.: Towards mass-produced building models. In: Stilla, U., Mayer, H., Rottensteiner, F., Heipke, C., Hinz, S. (eds.) *Photogrammetric Image Analysis*. pp. 209–220. Institute of Photogrammetry and Cartography, Technische Universitaet Muenchen (sep 2007)
13. Guan, T., Wang, C.: Registration based on scene recognition and natural features tracking techniques for wide-area augmented reality systems. *Multimedia, IEEE Transactions on* 11(8), 1393–1406 (2009)
14. Klein, G., Murray, D.: Parallel tracking and mapping for small ar workspaces. In: *Mixed and Augmented Reality, 2007. ISMAR 2007. 6th IEEE and ACM International Symposium on*. pp. 225–234 (2007)
15. Kurz, D., Benhimane, S.: Gravity-aware handheld augmented reality. In: *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*. pp. 111–120 (Oct 2011)
16. Kurz, D., Benhimane, S.: Augmented reality: Handheld augmented reality involving gravity measurements. *Computers & Graphics* 36(7), 866–883 (2012)
17. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. *Int. J. Comput. Vision* 60(2), 91–110 (nov 2004)
18. Pu, S.: Automatic building modeling from terrestrial laser scanning. In: Oosterom, P., Zlatanova, S., Penninga, F., Fendel, E.M. (eds.) *Advances in 3D Geoinformation Systems*, pp. 147–160. *Lecture Notes in Geoinformation and Cartography*, Springer Berlin Heidelberg (2008)
19. Van de Sande, K.E.A., Gevers, T., Snoek, C.G.M.: Evaluating color descriptors for object and scene recognition. vol. 32, pp. 1582–1596 (2010)
20. Suzuki, S., Be, K.: Topological structural analysis of digitized binary images by border following. *Computer Vision, Graphics, and Image Processing* 30(1), 32–46 (1985)
21. Valgren, C., Lilienthal, A.J.: Sift, SURF & seasons: Appearance-based long-term localization in outdoor environments. *Robotics and Autonomous Systems* 58(2), 149–156 (2010)
22. Wu, Y., Choubassi, M.E., Kozintsev, I.: Augmenting 3d urban environment using mobile devices. In: *Mixed and Augmented Reality (ISMAR), 2011 10th IEEE International Symposium on*. pp. 241–242 (2011)

Binary Segmentation of Multiband Images

Claudia Sánchez¹ and Mariano Rivera²

¹ Universidad Panamericana,
Facultad de Ingeniería, Aguascalientes, Ags.,
Mexico

² Centro de Investigación en Matemáticas,
Departamento de Ciencias Computacionales, Guanajuato, Gto.,
Mexico

cnsanchez@up.edu.mx, mrivera@cimat.mx

Abstract. We present a method for binary segmentation of multiband images based on a combination of dimensionality reduction techniques (Weighted PCA and Quadratic Programming Feature Selection), classification methods (Gaussian Mixtures Models and Random Forest) and segmentation method (Quadratic Markov Measure Field Models). In this work, four pixels descriptors are presented: Color, Discrete Cosine Transform, Gradient Fields and Adjacency Matrix. Our method combines the outcome of several classifiers using an optimization criterion. That results in a robust method for image segmentation based on color, textures and orientation. We evaluate our method capabilities with different image types for example: color images in RGB format and satellite images. Experimental results demonstrated our method performance.

Keywords: Image segmentation, maximum likelihood estimation, classification, multiband images, dimensionality reduction, weighted PCA, feature selection

1 Introduction

Image segmentation is an important issue in the image processing and computer vision areas, consists in divide an image in regions with similar features like color, texture, orientation, etc. It is the process to label each pixel in the image with the objective that pixels with the same label have similar features. More specific, image binary segmentation divide an image in two regions, it has practice applications as:

1. Separate the main object and the background,
2. Localization of tumors and other pathologies,
3. Classification of ground cover with the analysis of satellite images,
4. Face recognition.

Some algorithms and techniques of general purpose have been developed for the image segmentation, for example: [2], [5], [10], [13] and [14]. Unfortunately these methods can be used only in images with a small number of bands as RGB images.

The multiband images more common are the color images. In these images, the pixel color is obtained as the combination of three colors: red, blue and green, so these images have three bands. Another example of multiband images are the satellite images, which are the result of the information of the land cover captured by sensors of the artificial satellites. This kind of images are a tool very useful for the study of weather, land cover, etc. The application of the binary segmentation on these images could be used for land cover classification, detection of zones with particular features like agriculture areas, cities, etc. Based in [4], data of the different year seasons are used for the binary segmentation in satellite images, they use Feature Selection techniques for the reduction of the dimensionality and the Minimum Distance to the Center of the Classes or Maximum Likelihood for the classification. In [17] detect the vegetation in satellite images with a supervised learning technique for the classification, using the NDVI coefficient (Normalized Difference Vegetation Index) that is a transformation of the infrared band, near infrared band and the first main component.

Formally an image with n bands is an array of n bidimensionality arrays where each array has the information of the corresponding band, so a pixel is

$$x_{ij} = [x_{ij1}, x_{ij2}, \dots, x_{ijk}, \dots, x_{ijn}]. \quad (1)$$

So x_{ijk} has the value of the pixel (i, j) in the k band.

The proposed method, in contrast with the others methods mentioned in this section, is able to make a binary segmentation in different kinds of images as RGB images, infrared images, and satellite images. In addition, this method learns the feature that better difference between classes.

2 Brief Review of the Methods Used in this Work

2.1 Descriptors

A pixel descriptor is a set of data that represents the information of the color, texture and orientation of a specific pixel and its neighbors. The descriptors used in this work are:

Color: this descriptor has the pixel values in all the bands, as show in the equation 2.

$$D_{ij}^c = [x_{ij1}, x_{ij2}, \dots, x_{ijk}, \dots, x_{ijn}]. \quad (2)$$

DCT (Discrete Cosine Transform) described in [12]. The first step to calculate the DCT descriptor for a pixel i, j is get a matrix A^k with the values of the pixel (i, j) and its neighbors in the k band, as show in the equation 3.

$$A^k = \begin{bmatrix} \ddots & \vdots & \vdots & \vdots & \ddots \\ \cdots & x_{(i-1)(j-1)k} & x_{(i-1)jk} & x_{(i-1)(j+1)k} & \cdots \\ \cdots & x_{i(j-1)k} & x_{ijk} & x_{i(j+1)k} & \cdots \\ \cdots & x_{(i+1)(j-1)k} & x_{(i+1)jk} & x_{(i+1)(j+1)k} & \cdots \\ \ddots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \quad (3)$$

Then, the matrix B , that represents the Discrete Cosine Transformation of A , is calculate with the equation 4.

$$B_{pq}^k = \alpha_p \alpha_q \sum_{m=1}^M \sum_{n=1}^N A_{mn} \cos\left(\frac{\pi(2m+1)p}{2M}\right) \cos\left(\frac{\pi(2n+1)q}{2N}\right). \quad (4)$$

where the matrix B has the same size of A . M and N are the row and column size, p represents a row and q represents a column, so $1 \leq p \leq M$, $1 \leq q \leq N$. α_p and α_q are calculate with the equation 5 and 6 respectively:

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{M}} & p = 1 \\ \sqrt{\frac{2}{M}} & 2 \leq p \leq M \end{cases}, \quad (5)$$

$$\alpha_q = \begin{cases} \frac{1}{\sqrt{N}} & q = 1 \\ \sqrt{\frac{2}{N}} & 2 \leq q \leq N \end{cases}. \quad (6)$$

Finally the descriptor DCT is the vectorization of all the resulting matrices of the DCT in each band, as show in the equation 7:

$$D_{ij}^{DCT} = [B_{11}^1, B_{12}^1, \dots, B_{11}^k, B_{12}^k, \dots, B_{11}^n, B_{12}^n, \dots]. \quad (7)$$

GF (Gradient Fields) It is based on the gradient of each pixel, it means, the magnitud and orientation of the maximum difference between neighbor pixels. The first step is calculate two matrices for each band that represent the difference between horizontal neighbor pixels and vertical neighbor pixels, as show in the equation 8:

$$DX_{ijk} = x_{ijk} - x_{(i-1)jk}, DY_{ijk} = x_{ijk} - x_{i(j-1)k}. \quad (8)$$

Then calculate the magnitud and the orientation based in the DX and DY matrices, equation 9:

$$M_{ijk} = \sqrt{(DX_{ijk})^2 + (DY_{ijk})^2}, O_{ijk} = \tan^{-1}\left(\frac{DY_{ijk}}{DX_{ijk}}\right). \quad (9)$$

After, for each band and for each pixel two sub matrices centering in the pixel (i, j) are extracted, one matrix of magnitudes and other matrix for orientations, with the objective to calculate a normalized histogram of the gradient orientation, equation 10:

$$H_{ijk} = [H_{ijk}^{0^\circ-20^\circ}, H_{ijk}^{20^\circ-40^\circ}, \dots, H_{ijk}^{340^\circ-360^\circ}]. \quad (10)$$

Finally, the GF descriptor is formed with all the values of the histograms, equation 11:

$$D_{ij}^{GF} = [H_{ij1}, H_{ij2}, \dots, H_{ijn}]. \quad (11)$$

AD (Adjacency Matrix), the first step to calculate the AD descriptor consists in reduce the dimensionality of the pixels in the image to one band using PCA, described in [1]. Then the values are discretized in b bins to obtain I . After two matrices are calculate: horizontal adjacency matrix and vertical adjacency matrix, as show in the equation 12:

$$A_{rs}^H = \sum_{i=2}^M \sum_{j=1}^N \delta(I_{ij}, r) \delta(I_{(i-1)j}, s), A_{rs}^V = \sum_{i=2}^M \sum_{j=1}^N \delta(I_{ij}, r) \delta(I_{i(j-1)}, s), \quad (12)$$

where the $A^V = \{A_{rs}^V\}_{r,s=1,\dots,b}$, $A^H = \{A_{rs}^H\}_{r,s=1,\dots,b}$, M and N are the number of rows and number of columns in the image, δ is the Kronecher delta where the result is one if the parameters are equals or zero if are different.

Then the matrix exponential is used to increment the response of the adjacency, equation 13:

$$EA^V = I + A^V + \frac{(A^V)^2}{2!} + \frac{(A^V)^3}{3!}, EA^H = I + A^H + \frac{(A^H)^2}{2!} + \frac{(A^H)^3}{3!}. \quad (13)$$

Finally the descriptor AD of the pixel (i, j) is the vectorization of values in the sub matrix of EA^V centering in (i, j) and the sub matrix of EA^H centering in (i, j) , equation 14:

$$D_{ij}^{AD} = [\dots, EA_{(i-1)(j-1)}^V, \dots, EA_{ij}^V, \dots, EA_{ij}^H, \dots, EA_{(i+1)(j+1)}^H, \dots]. \quad (14)$$

2.2 Dimensionality Reduction

WPCA (Weighted PCA) described in [16], is a technique based on PCA that assume each dimension contributes in a different proportion to represent the information. We can see the segmentation problem as a classification problem, so, the contribution of each variable depends on its classification capability.

Asuming $X = \{x_{ij}\}, i = 1, 2, \dots, N$ and $j = 1, 2, \dots, D$ the goal of weighted PCA is project the data in a space of dimensionality $M < D$, the steps are:

1. Calculate a vector with the variables weight $W = [w_1, w_2, \dots, w_D]^T$, where D is the dimension number. The weights can be calculated with a dependency measurement as Pearson correlation coefficient between the variable and the class.
2. Normalize the vector W in order to $w_j > 0 \forall j$ and $\sum_{j=1}^D w_j = 1$,
3. Calculate the mean vector $\bar{X} = [\bar{x}_1, \bar{x}_2, \dots, \bar{x}_D]^T$ and the variance vector $s = [s_1, s_2, \dots, s_D]^T$,
4. Standardize the data using the means and variances $z_{ij} = \frac{x_{ij} - \bar{x}_j}{\sqrt{s_j}}$,
5. Weighted the data using the vector W , $z_{ij}^* = z_{ij} \sqrt{w_j}$,

6. Calculate the projection vectors P as the M bigger eigenvectors of $(Z^*)^T(Z^*)$, where $Z^* = z_{ij}^*, i = 1, 2, \dots, N$ and $j = 1, 2, \dots, D$,
7. Send the data to the origin $\hat{x}_{ij} = x_{ij} - \bar{x}_j$,
8. Calculate the main components projecting the data $Y = \hat{X}P$, where $\hat{X} = \hat{x}_{ij}$.

Finally, the new data are $Y = \{y_{ij}\}, i = 1, 2, \dots, N$ and $j = 1, 2, \dots, M$.

QPFS (Quadratic Programming Feature Selection) described in [15], is a feature selection method to classification problems using quadratic programming. It reduce the redundancy between variables and maximize the dependency between the variables and the class variable. The main goal is provide a method of reasonable complex for classification problems of high dimensionality. Consists in solve the optimization problem defined in the formula 15.

$$\begin{aligned} \min_w \frac{1}{2}(1 - \alpha)w^T Qw - \alpha F^T w \\ \text{s.t.} \\ w_i \geq 0, \sum_{i=1}^D w_i = 1. \end{aligned} \quad (15)$$

Q is a quadratic simetric matrix that represents the redundancy between variables. The size of Q is $D \times D$, where D is the variable number. $Q = \{q_{ij}\}$, where q_{ij} represents the dependency between the i and j variables.

q_{ij} is calculate with a dependency measurement as the mutual information [6], [15] for discrete variables, equation 16:

$$q_{ij} = I(X_i, X_j) = \sum_{h_i \in X_i} \sum_{h_j \in X_j} P(h_i, h_j) \log \left(\frac{P(h_i, h_j)}{P(h_i)P(h_j)} \right), \quad (16)$$

or Pearson correlation coefficient, described in [15], for continuos variables, equation 17:

$$q_{ij} = r_{ij} = \frac{\sum_{k=1}^N (X_{ki} - \bar{X}_i)(X_{kj} - \bar{X}_j)}{\sqrt{\sum_{k=1}^N (X_{ki} - \bar{X}_i)^2} \sqrt{\sum_{k=1}^N (X_{kj} - \bar{X}_j)^2}}. \quad (17)$$

F is a vector that represents the dependency between the variables and class variable. The size of F is $D \times 1$. f_i can be calculated with the mutual information or with the Pearson correlation coefficient depending on the variable types. $F = \{f_i\}$, each f_i represents the dependency between the i variable and the class variable.

α controls the importance of the relevance in front of the redundancy. It must be $0 \leq \alpha \leq 1$. Large values of α represents that the dependency is more important but the selected variables can be redundants. Small values represents reducing the redundancy is more important.

To guarantee a good solution the optimization problem have two restrictions: $w_i \geq 0$ all the variable weighted must be bigger than zero and $\sum_{i=1}^D w_i = 1$ the sum of all variable weights must be 1.

Finally, only the variables with the bigger weights are selected.

2.3 Classification

GMM (Gaussian Mixture Model) described in [1], allows to model complex distributions of data sets based on a linear combination of Gaussians. The parameters of a GMM are calculate with the EM algorithm. Formally the form to calculate the likelihood with a GMM is with the equation 18:

$$p(x) = \sum_{k=1}^K \pi_k N(x|\mu_k, \Sigma_k), \quad (18)$$

where x is a vector, k is the Gaussian number, π_k the proportion of the k Gaussian, μ_k and Σ_k are the mean vector and covariance matrix of the k Gaussian. $0 \leq \pi_k \leq 1 \forall k, \sum_k \pi_k = 1$.

RF (Random Forest) described in [3], is a set of random decision trees, each one created with a random subset of the training dataset. A decision tree, described in [11], is a prediction model based in a series of questions about of variable values to predict the class. In a decision tree the data are organized in rectangular regions, product of the questions, with the objective of the data in a region be as possible of the same class.

2.4 Segmentation

QMMF (Quadratic Markov Measure Field Models) described in [13], calculate the probability to set a label to the pixel, unlike hard segmentation set the label to the pixel. Once the likelihood of each pixel is calculated as a normalized vector $\hat{v}(r)$ that shows the membership to the r pixel to each class QMMF calculates the probability $p(r)$ as a normalized vector that shows the probability to belong to each class based on the likelihood, the neighbor probabilities, and the entropy. QMMF is based on the Quadratic Programming defined in the formula 19.

$$\min_p \frac{1}{2} \sum_r Q(p(r), \hat{v}(r)) - \frac{\mu}{2} \sum_r \|p(r)\|_2^2 + \frac{\lambda}{2} \sum_{\langle r,s \rangle} R(p), \quad (19)$$

where μ and λ control the contribution to each term. The first term relates the solution p with the likelihood \hat{v} , $Q(p(r), \hat{v}(r)) = p(r)^T D_r p(r)$, $D_r = \text{diag}(-\log(\hat{v}(r)))$. The second term controls the solution entropy with the objective it be small. Finally, the third term produce soft spatially solution and r and s are first neighbors. The optimal solution is calculate with Gauss-Seidel Projected described in [9].

3 Proposed Method

The proposed method makes the binary segmentation of an image based on user clues at the beginning of the process to establish the features of each class. The



Fig. 1. User clues for the segmentation. a) Image b) User clues, white pixels are a sample of pixels of the class one and black pixels are another sample of pixels of the class two.

clues consist into mark a sample of pixels as class one and mark another sample of pixels as class two. The figure 1 shows an example of user clues.

Our method is divided into two phases:

1. Training with the marked pixels,
2. Classification of the not marked pixels and Segmentation.

3.1 Training with the Marked Pixels

The first phase consists in recognize the features of each class based on the user clues, this is: found the descriptors that identify the features that separate the classes, calculate the parameters to an optimal dimensional reducing and calculate the parameters to the classification models.

In this phase we only use the information of the marked pixels, the user clues, to learn the models for the classification. The steps, shown in the figure 2, are the following:

1. Create the descriptors: Color, DCT, GF and AD for the marked pixels.
2. Reduce the descriptors dimensionality using: WPCA and QPFS. So we have eight reduce descriptors into two groups:
 - (a) Group 1: Color reduced with QPFS, DCT reduced with QPFS, GF reduced with QPFS and AD reduced with QPFS.
 - (b) Group 2: Color reduced with WPCA, DCT reduced with WPCA, GF reduced with WPCA and AD reduced with WPCA.
3. Create the models, as we have two dimensionality reduction techniques and two classification methods, we group the models as the next form:
 - (a) Group 1: GMM of the descriptors reduced with QPFS,
 - (b) Group 2: RF of the descriptors reduced with QPFS,
 - (c) Group 3: GMM of the descriptors reduced with WPCA,
 - (d) Group 4: RF of the descriptors reduced with WPCA.
4. Classification of the marked pixels with the models created in the previous step. We calculate 16 different likelihoods for each marked pixel because we have four groups and four descriptors in each group.

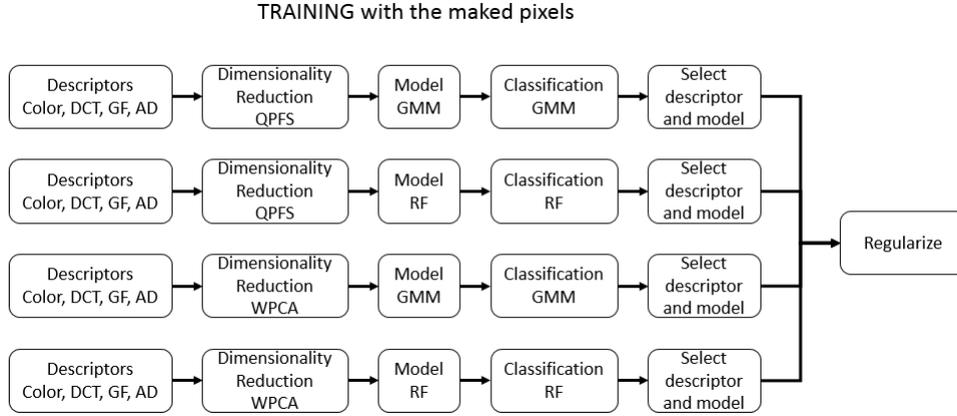


Fig. 2. Proposed method, first phase.

5. Select the descriptor and model for each group. For each group:
 - (a) Calculate the accuracy for each descriptor with the equation 20.

$$Accuracy = \frac{\sum_{\langle i \rangle} \varphi(V_i^1, V_i^2, C_i)}{Nmp}, \quad (20)$$

where

$$\varphi(V_i^1, V_i^2, C_i) = \begin{cases} 1 & \text{if } C_i = 1, V_i^1 \geq V_i^2 \\ 1 & \text{if } C_i = 2, V_i^2 \geq V_i^1 \\ 0 & \text{otherwise} \end{cases}, \quad (21)$$

where Nmp is the number of marked pixels and $\langle i \rangle$ represents only the marked pixels.

- (b) Select the descriptor with the bigger accuracy.
- (c) Select the model, RF or GMM, that corresponds to the selected descriptor.
- (d) Normalize the likelihoods of the descriptor selected, with the goal of $V_i^1 + V_i^2 = 1$, with the equation 22:

$$V_i^1 = \frac{V_i^1 + \varepsilon}{V_i^1 + V_i^2 + \varepsilon}. \quad (22)$$

6. Regularize, this is, calculate the weight of each group to the classification.
 - (a) Calculate the efficiency to the classification for each group. a^g , $g = 1, 2, 3, 4$, represents the classification efficiency of the group g , and it is calculate as the accuracy of the normalize likelihood.
 - (b) Regularize the accuracy, this is, $\hat{a}^1 + \hat{a}^2 + \hat{a}^3 + \hat{a}^4 = 1$, whit the equation 23:

$$\begin{aligned} \hat{a}^1 &= \frac{a^1}{a^1 + a^2 + a^3 + a^4} & \hat{a}^2 &= \frac{a^2}{a^1 + a^2 + a^3 + a^4} \\ \hat{a}^3 &= \frac{a^3}{a^1 + a^2 + a^3 + a^4} & \hat{a}^4 &= \frac{a^4}{a^1 + a^2 + a^3 + a^4} \end{aligned}. \quad (23)$$

3.2 Classification of the not Marked Pixels and Segmentation

The second phase consists in the classification of the no marked pixels and the segmentation of the results, it means: calculate the descriptor of the no marked pixels, reduce the descriptors dimensionality and calculate the likelihood of the not marked pixels with the models created in the first phase, combination of the four classifiers results and segmentation. The steps of this phase, show in the figure 3, are the following:

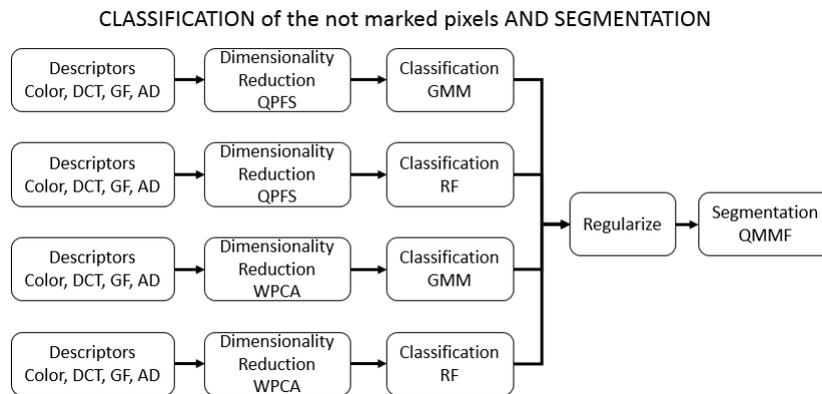


Fig. 3. Proposed method, second phase.

1. Create the selected descriptors for the no marked pixels.
2. Reduce the descriptors dimensionality using: WPCA and QPFS.
3. Classify the no marked pixels using the reduce descriptors and the models created and selected in the first phase. We must to have four likelihoods, one for each group:
 - (a) Group 1: QPFS and GMM,
 - (b) Group 2: QPFS and RF,
 - (c) Group 3: WPCA and GMM,
 - (d) Group 4: WPCA and RF.
4. Calculate the general likelihood using the accuracy of each group.
 - (a) Normalize the likelihood of each group, as mentioned in the equation 22.
 - (b) Combine the classifiers results 24:

$$\hat{V}_i^1 = \sum_{g=1}^4 \hat{a}^g V_{gi}^1. \quad (24)$$

5. Segment the likelihood using QMMF.

Table 1. Percentage of error in the classification of the Statlog dataset.

| Class | QPFS | QPFS | WPCA | WPCA | Combination |
|-------|---------|--------|-------------|--------|-------------|
| | and GMM | and RF | and GMM | and RF | |
| 1 | 2.97 | 4.93 | 1.97 | 1.94 | 1.18 |
| 2 | 3.75 | 2.36 | 1.12 | 1.46 | 1.43 |
| 3 | 7.04 | 5.56 | 5.28 | 4.55 | 3.82 |
| 4 | 14.39 | 8.81 | 9.29 | 7.94 | 6.53 |
| 5 | 7.32 | 5.19 | 7.52 | 4.72 | 3.43 |
| 7 | 11.62 | 8.86 | 8.17 | 7.02 | 5.87 |
| Mean | 7.85 | 5.95 | 5.56 | 4.61 | 3.71 |

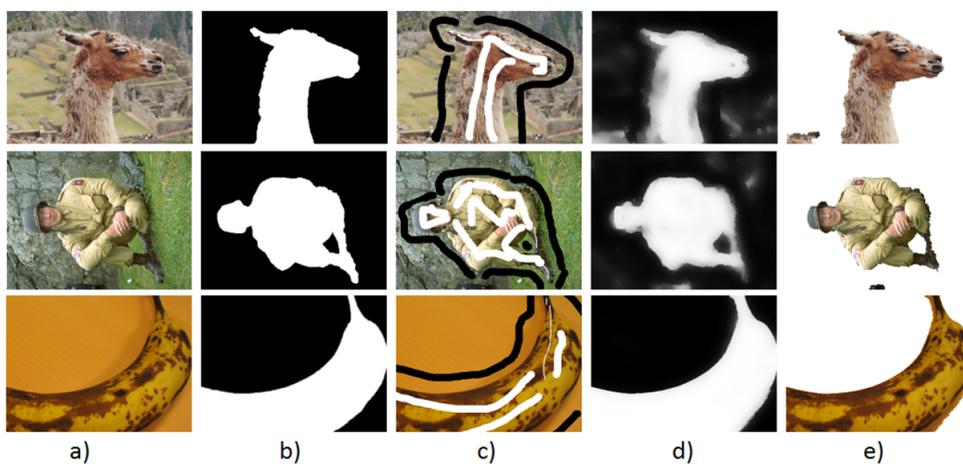


Fig. 4. Results of binary segmentation of images with real textures a) Image b) Real segmentation c) User clues d) Proposed method segmentation e) Main object, calculated with the proposed method

4 Experiments and Results

4.1 Statlog Dataset

The Statlog dataset [7] has information of satellite images of the Landsat satellite. This dataset consists in multivariate data of pixels in 3x3 neighborhood and the classification of the central pixel. The objective is predict the classification. The pixel class is coded with a number that represents: 1 red soil, 2 cotton crop, 3 grey soil, 4 damp grey soil, 5 soil with vegetation stubble, 6 mixture class and 7 very damp grey soil. Each neighborhood is represented by 36 variables more the class, the data set is composite by 6,435 data. The objective of this experiment is verify the classifiers combination is better than each classifier by separated, the results are show in table 1.

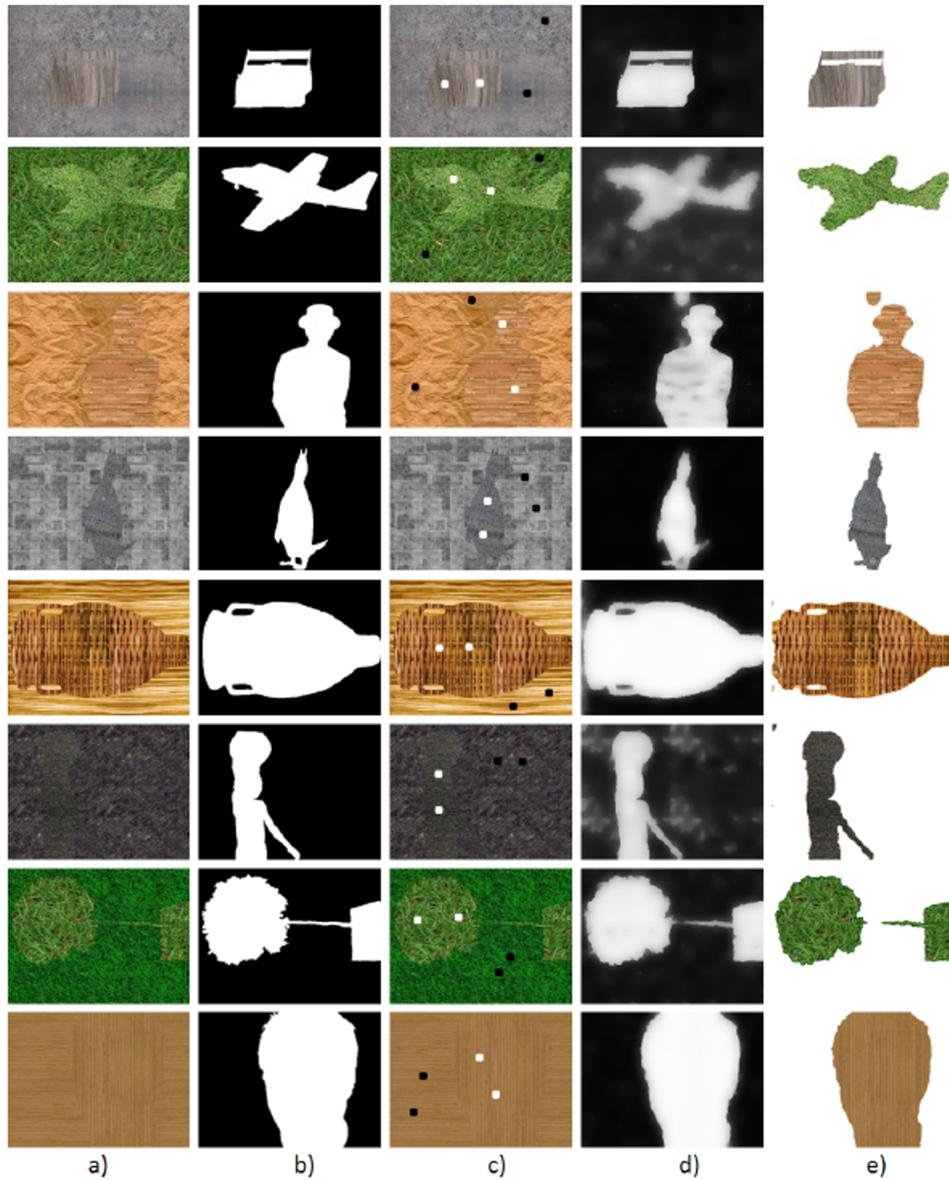


Fig. 5. Results of binary segmentation of images with different textures. a) Image b) Real segmentation c) User clues d) Proposed method segmentation e) Main object calculated with the proposed method

4.2 Images with Real Textures

The images with real textures was obtained of the Microsoft Research Cambridge dataset [8]. The objective to this experiment is measure the efficiency of the

proposed method with real images in RGB format. The dataset consists in 50 images as the images shown in the figure 4. The mean percentage of error obtained with the proposed method is 3.6%.

4.3 Images with Different Textures

The images with different textures was created based on the Microsoft Research Cambridge dataset [8] replacing the main object and the background with texture images. The objective to this experiment is measure the efficiency of the proposed method to segmentation by color, texture and orientation. The results are show in the figure 5.

5 Conclusions

The use of four descriptors: Color, DCT, GF and AD creates a robust segmentation by color, texture and orientation. Descriptor selection improvements the results because only the features that maximize the difference between classes are used, so this makes easy the models creation because we have a moderate dimensionality instead of use the information of all descriptors. With the selection of the better descriptors we can say that the method is adapted to each image.

The lineal combination of four classifiers: (1) WPCA and GMM, (2) WPCA and RF, (3) QPFS and GMM, and (4) QPFS and GMM produces better results than the use of only one classifier.

The proposed method can be implemented in parallel form so can be applied efficiently in satellite images.

References

1. Bishop, C.: Pattern Recognition and Machine Learning. Springer (2006)
2. Boykov, Y., Jolly, M.: Interactive graph cut for optimal boundary and region segmentation. ICIIP (2001)
3. Breiman, L.: Random forest. Machine Learning pp. 5–32 (2001)
4. Caetano, M.: Image classification. ESA Advanced training course on Land Remote Sensing (2009)
5. Grady, L.: Random walks for image segmentation. ECCV 1 (2004)
6. L., H.P.F., Ding, C.: Feature selection based on mutual information: criteria of max-dependency, max-relevance, and min-redundancy. IEEE Trans pp. 1226–1238 (2005)
7. Machine learning repository. statlog dataset (2012), "[http://archive.ics.uci.edu/ml/datasets/Statlog+\(Landsat+Satellite\)](http://archive.ics.uci.edu/ml/datasets/Statlog+(Landsat+Satellite))"
8. Microsoft research cambridge. dataset of test image (2012), "<http://research.microsoft.com/en-us/um/cambridge/projects/visionimagevideoediting/segmentation/grabcut.htm>"
9. Nocedal, J., W., S.J.: Numerical Optimization. Springer (2006)

10. Ocegueda, O., Rivera, M., Marroquín, J.: Entropy-controlled quadratic markov measure field models for efficient image segmentation. *IEEE Trans. Image Processing* (2007)
11. Olshen, C.R.A., Stone, J., Breiman, L.: *Classification and regression trees*. CRC Press (1998)
12. Pennebaker, W., Mitchell, J.: *Jpeg: Still image data compression standard*. Van Nostrand Reinhold (1993)
13. Rivera, M., Dalmau, O.: Variational viewpoint of the quadratic markov measure field models: Theory and algorithms (2010)
14. Rivera, M., Dalmau, O., Mio, W., Ramirez, A.: Spatial sampling for image segmentation. *The computer Journal* (2010)
15. Rodríguez, L.I.: *Selección de variables mediante programación cuadrática*. Universidad Autónoma de Madrid (2009)
16. Thomaz, C.: A simple and efficient supervised method for spatially weighted pca in face image analysis. Imperial College London, Technical Report (2010)
17. Viana, H., Aranha, J., Rodrigues, R.: Vegetation classification and quantification by satellite image processing. a case study in north portugal. *International Conference and Exhibition on Bioenergy* (2008)

Fuzzy Logic Applied to Improvement of Image Resolution using Gaussian Membership Functions

Samuel Souverville, Jorge A. Rosales, Francisco J. Gallegos, Mario Dehesa, Isabel V. Hernández, and Lucero V. Lozano

Instituto Politécnico Nacional,
Escuela Superior de Ingeniería Mecánica y Eléctrica Unidad Zacatenco,
S.E.P.I. Maestría en Ciencias en Ingeniería Electrónica,
México DF, Mexico

samuel_souverville@hotmail.com, arosales@ipn.mx

Abstract. The resolution in images is a perceptible detail measure. If the resolution increases, perception of fine details, edges, clearness of the objects and image quality increases too. Video surveillance cameras usually have a standard resolution for video surveillance applications, commonly in VGA resolution (640 x 480 pixels). This video image in most of the cases does not provide enough information to identify a person or an object, the cameras with low resolution deliver poor data information and poor information in detailed images to maximize its size. If an area needs more resolution, it is necessary an algorithm that achieve this without the loss of inherent characteristics. We selected the fuzzy logic theory to solve these problems. This technique is used to improve image resolution. It helps in processes where ambiguity and vagueness in the data interpolation are present, this is due to the non-linearity of image information (edges, fine details, textures, etc.). The proposed Gaussian membership functions have non-linear characteristics, so they obtain good results in interpolation process.

Keywords: Super resolution, Gaussian functions, color images, fuzzy interpolation

1 Introduction

Video surveillance is one of the most important applications in the security systems, it helps to detect intruders, identify and prevent crimes, it is useful to deliver evidence of crimes. This technique is known as “Super resolution”. It is also used in:

- Traffic Monitoring.
- Land traffic characteristics such as speed, and acceleration.
- Industrial processes improving.
- Business management.
- Medical activities.

The super resolution is needed in applications like recognition, image analysis, medical imaging for a better diagnosis, and applications where a zoom is required, this

for a specific area of interest and it is where the super resolution becomes essential, for example, video surveillance, satellite imagery and more [1-6].

However, the high resolution images are not always available, this is because it is often costly to obtain a high resolution image and sometimes may not be feasible due to the limitations of the sensor, among others. These drawbacks can be solved using mathematical image processing algorithms, which are relatively inexpensive, leading to the concept of super-resolution. This gives us an advantage because it can cost less and systems of existing low-resolution images are still used [7].

Agree to the surveillance monitoring respect to a large open area using a camera, means losing important details to enable the people feature analysis or identification. The images are important in the analysis of crime as well as evidence of abuse. A possible solution to these problems is to enhance low resolution video surveillance systems with advanced algorithms that realize complex activities, such as increasing resolution in the images.

Phenomena found every day are imprecise, i.e., have the ambiguity and vagueness implied in the scene captured. This imprecision can be associated in its shape, position, time, color, texture, or even semantic in the scene. This is the behavior of the images [8].

Fuzzy logic is conceptualized as a generalization of classical logic. It is a branch of artificial intelligence that allows specification vagueness to handle information. The first logic of vagueness was developed in 1920 by the philosopher Jan Lukasiewicz, visualized with possible joint membership degree values of 0 and 1, then the un-extended to an infinite number of values between 0 and 1 [9].

In 1960, Lofti Zadeh creates a powerful tool, known as fuzzy logic to model imprecise data in which the inference rules are formulated in a very general way making use of fuzzy categories, Lukasiewicz combines the concepts of logic and sets defining by membership degrees.

2 Method

The edge-based line average (ELA) algorithm is a well-known interpolation method in the spatial domain. Linear interpolation is the most commonly used method for de-interlacing. The edge-based line average (ELA) algorithm uses directional correlation among pixels to perform linear interpolation. There are three detection directions as shown in Fig.1, which are vertical, and diagonal. In each direction, the difference is calculated [10].

ELA looks for the possible edge direction and then applies the line average along the selected direction. This algorithm works well when the edge directions are estimated correctly but, otherwise, it introduces errors and degrades the image quality. In this paper is presented a new membership function is presented which improves the robustness of the original ELA algorithm, see Fig 4 Gaussian membership functions.

The inputs of the algorithms are computed as the absolute difference values of the luminance differences in the three directions (a, b, c) shown in Fig. 1.

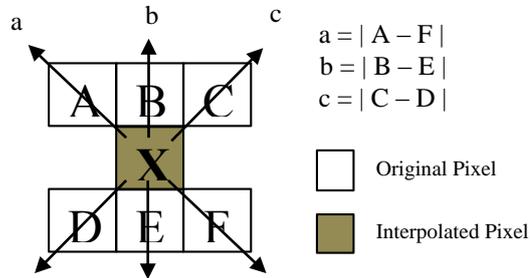


Fig. 1. Pixels involved in ELA 3+3 algorithm.

The fuzzy rules to connect the fuzzy values are found in Table 1.

Table 1. Fuzzy Rule for ELA 3 + 3 [5]

| IF | ANTECEDENTS | THEN | CONSEQUENT |
|----|---|------|-----------------------|
| 1 | a is <i>medium</i> , b is <i>big</i> and c is <i>big</i> | | $(A + F) / 2$ |
| 2 | a is <i>big</i> , b is <i>big</i> and c is <i>medium</i> | | $(C + D) / 2$ |
| 3 | a is <i>small</i> and b is <i>big</i> and c is <i>small</i> | | $(A + C + D + F) / 4$ |
| 4 | otherwise | | $(B + E) / 2$ |

The fuzzy rules 1 and 2, delivers values near to 1 (≈ 1) when the correlation is big in one direction while deliver small values (≈ 0) in the opposite directions. In both cases, the result is obtained by interpolating the average value of the luminance $(A+F)/2$ or $(C+D)/2$.

The fuzzy rule 3, estimates the fuzzy value of an edge because of the correlation, in this case, if big (≈ 1) for both directions illustrated in Fig. 1. In this case, we get a result interpolating the four pixels $(A+C+D+F)/4$. Finally, in the fuzzy rule 4, the otherwise antecedent parameter is obtained interpolating in vertical direction agree to $(B+E)/2$.

This method works using an amplification factor equal to 2 as shown in Fig. 2.

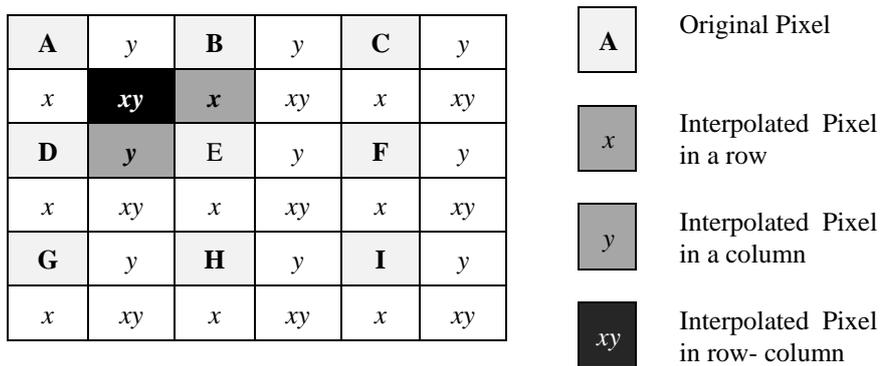


Fig. 2. Pixels involved to resolution increase with an amplification factor of two.

According to Fig. 2 are used eight pixels from the original image, these pixels are labeled as: **A, B, C, D, E, F, G, H, F**; first is interpolate the **pixel “x”** in the row, this is achieved using **A, B, C, D, E and F** pixels, second interpolate the **pixel “y”** in the column **A, B, D, E, G and H**, and finally interpolate the **pixel “xy”** row-column using the eight pixels that are located around the pixel “xy”. Four of the pixels are from the original image and the other ones were previously interpolated.

The ELA module increases the processing window up to 5+5 pixels. The **ELA 5 + 5** algorithm consider the closest pixels to the external ends (**A', C', D', F'**) as shown in Fig. 3 which includes two new directions (**a'** and **c'**).

Consequently, the fuzzy inference system has six fuzzy rules instead of 4 as in the ELA 3 + 3 algorithm, these fuzzy rules are shown in the Table 2 [11].

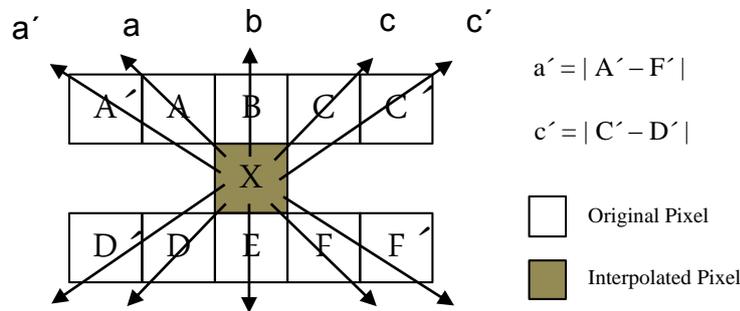


Fig. 3. Pixels used in the ELA 5+5 algorithm.

Table 2. Fuzzy Rule for ELA 5 + 5

| IF | ANTECEDENT | THEN | CONSEQUENT |
|----|--|------|-----------------------|
| 1) | a' is <i>medium</i> and a is <i>big</i> and b is <i>big</i> and c is <i>big</i> and c' is <i>big</i> | | $(A' + F') / 2$ |
| 2) | a' is <i>big</i> and a is <i>big</i> and b is <i>big</i> and c is <i>big</i> and c' is <i>medium</i> | | $(C' + D') / 2$ |
| 3) | a' is <i>medium</i> and a is <i>medium</i> and b is <i>big</i> and c is <i>big</i> and c' is <i>big</i> | | $(A + F) / 2$ |
| 4) | a' is <i>big</i> and a is <i>big</i> and b is <i>big</i> and c is <i>medium</i> and c' is <i>big</i> | | $(C + D) / 2$ |
| 5) | a is <i>small</i> y b is <i>big</i> y c is <i>small</i> | | $(A + C + D + F) / 4$ |
| 6) | otherwise | | $(B + E) / 2$ |

Because of the images do not have linear behavior, it is proposed nonlinear membership functions, so Gaussian membership functions solve this problem (Eq. 1), to take into account the mean and the variance values of the sample processed in the image. This allows adaptability of the algorithm to texture changes, and produce good interpolation results.

$$f(x) = ae^{-\frac{(x-b)^2}{2c^2}}, \quad (1)$$

where

$$a = \frac{1}{c\sqrt{2\pi}} \quad b = \mu \quad c = \sigma.$$

The Fig. 4 shows the three groups were formed (small, medium and large) where the *x-axis* represents the luminance and the *y-axis* represents the fuzzy value.

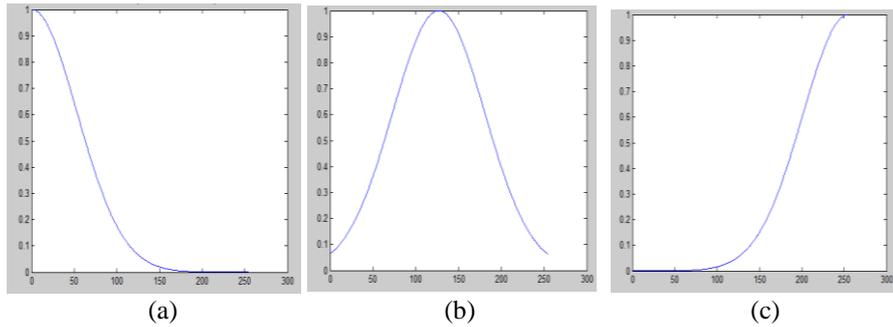


Fig. 4 Membership Functions (a) small, (b) medium, (c) big

The equations (2) and (3) show us how we can find the weight of ELA 3+3 and ELA 5+5 algorithms. For ELA 3+3 algorithm [5]:

$$\begin{aligned} \alpha_1 &= \min[\mu_{medium_a}(h), \mu_{big_b}(h), \mu_{big_c}(h)], \\ \alpha_2 &= \min[\mu_{big_a}(h), \mu_{big_b}(h), \mu_{medium_c}(h)], \\ \alpha_3 &= \min[\mu_{small_a}(h), \mu_{big_b}(h), \mu_{small_c}(h)], \\ \alpha_4 &= 1 - \alpha_1 - \alpha_2 - \alpha_3, \end{aligned} \quad (2)$$

where for ELA 5+5 algorithm [5],

$$\begin{aligned} \alpha_1 &= prod[\mu_{medium_{a'}}(h), \mu_{big_a}(h), \mu_{big_b}(h), \mu_{big_c}(h), \mu_{big_{c'}}(h)], \\ \alpha_2 &= prod[\mu_{big_{a'}}(h), \mu_{big_a}(h), \mu_{big_b}(h), \mu_{big_c}(h), \mu_{medium_{c'}}(h)], \\ \alpha_3 &= prod[\mu_{big_{a'}}(h), \mu_{medium_a}(h), \mu_{big_b}(h), \mu_{big_c}(h), \mu_{big_{c'}}(h)], \\ \alpha_4 &= prod[\mu_{big_{a'}}(h), \mu_{big_a}(h), \mu_{big_b}(h), \mu_{medium_c}(h), \mu_{big_{c'}}(h)], \\ \alpha_5 &= \min[\mu_{small_a}(h), \mu_{big_b}(h), \mu_{small_c}(h)], \\ \alpha_6 &= 1 - \alpha_1 - \alpha_2 - \alpha_3 - \alpha_4 - \alpha_5. \end{aligned} \quad (3)$$

The defuzzification processes of algorithms ELA 3+3 and ELA 5+5 are illustrated in equations (4) and (5):

$$X = \alpha_1 \left(\frac{A+F}{2} \right) + \alpha_2 \left(\frac{C+D}{2} \right) + \alpha_3 \left(\frac{A+F+C+D}{4} \right) + \alpha_4 \left(\frac{B+E}{2} \right), \quad (4)$$

$$X = \alpha_1 \left(\frac{A'+F'}{2} \right) + \alpha_2 \left(\frac{C'+D'}{2} \right) + \alpha_3 \left(\frac{A+F}{2} \right) + \alpha_4 \left(\frac{C'+D'}{2} \right) + \alpha_5 \left(\frac{A+F+C+D}{4} \right) + \alpha_6 \left(\frac{B+E}{2} \right). \quad (5)$$

3 Evaluation of Results

3.1 Pick Signal to Noise Ratio (PSNR)

The PSNR criterion used to compare the performance of different algorithms Eq. 6:

$$PSNR = 10 * \log \left[\frac{(255)^2}{MSE} \right]. \quad (6)$$

3.2 Mean Absolute Error (MAE)

The MAE is the criterion for assessing the preservation of contours and fine details because of this was suggested for the correlation with the human visual system; Eq. 7 computes the MAE:

$$MAE = \frac{\sum_{i,j} (|In(i,j) - Iorg(i,j)|)}{M \times N}, \quad (7)$$

where $In(i, j)$ represents the values of the original image, and $Iorg(i, j)$ represents the values of the restored Image.

3.3 Mean Square Error (MSE)

The MSE is the approach that presents an objective measure of the average square deviation to find the estimate of the true value and it's calculated by the equation (8) (it is the most common objective measure to compare the quality of the filter between the original image and the filtered one).

$$MSE = \frac{\sum_{i,j} (In(i,j) - Iorg(i,j))^2}{M \times N}, \quad (8)$$

where $In(i, j)$ represent the values of the original image planes, and $Iorg(i, j)$ represent the values of the pixels for the restored image [12].

Criteria evaluation results are achieved using original image dimensions the same as the interpolated one, that mean that we must have original non interpolated image and the same original non interpolated image but with the double in its size to compare pixel by pixel with the interpolated image.

4 Results

The algorithms described before (ELA 3 + 3 and ELA 5 + 5) were applied to well-known images as "Lena", "Peppers" and "Baboon" see Fig. 5, 6, and 7, because they emulate different environments like colors changes, textures etc.

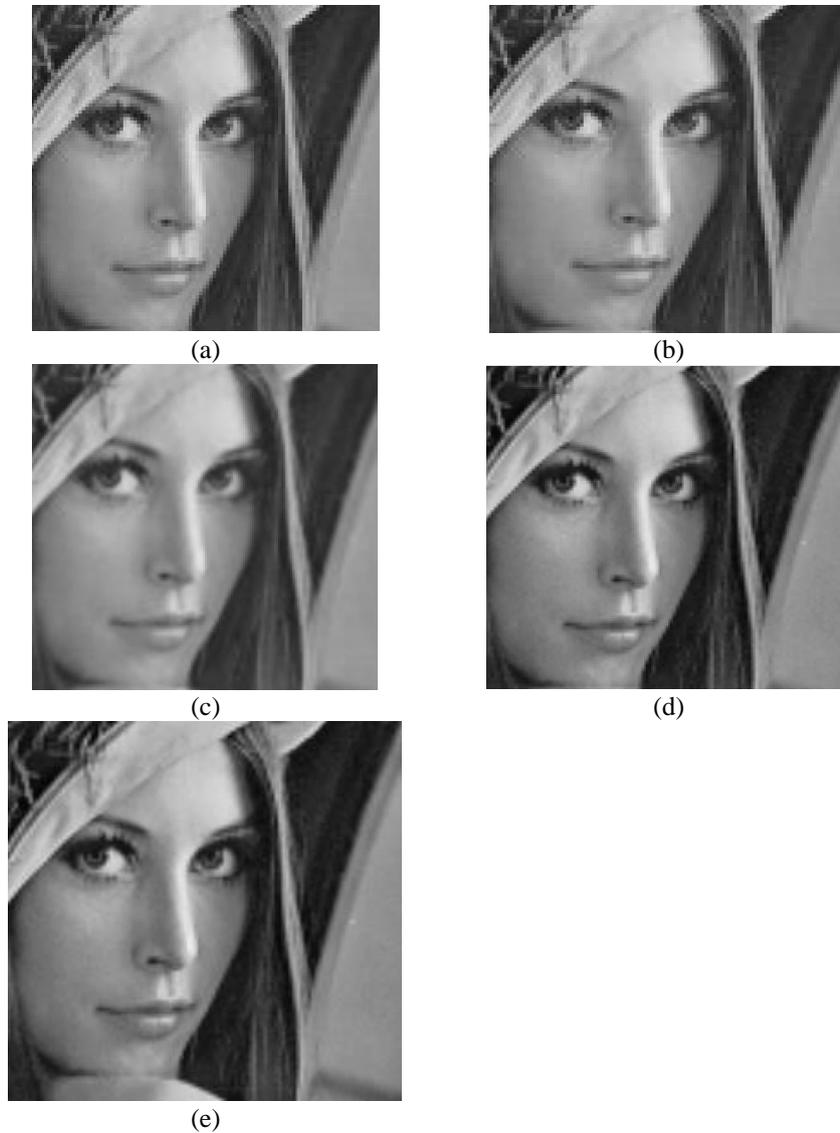


Fig. 5. Lena (a) Original image 256x256, (b) KNN 512x512 interpolated image, (c) Bilinear 512x512 interpolated image, (d) ELA 3 + 3 Interpolated Image, (e) ELA 5 + 5 Interpolated Image.

The Lena, Baboon and Peppers images had a good preservation in details and edges. The images show a significant improvement results in a qualitative and quantitative way, due to the algorithm ELA. It delivers better results compared with other methods because of nonlinear membership functions applied to identify edges and details in interpolation algorithm to preserve them.

In Table 3 we can see the quantitative results of the suggested interpolations, the image “Peppers” delivers the highest peak signal to noise ratio because the image contains large homogeneous areas, Baboon image deliver the highest mean absolute error due to the color changes in a sharply way.

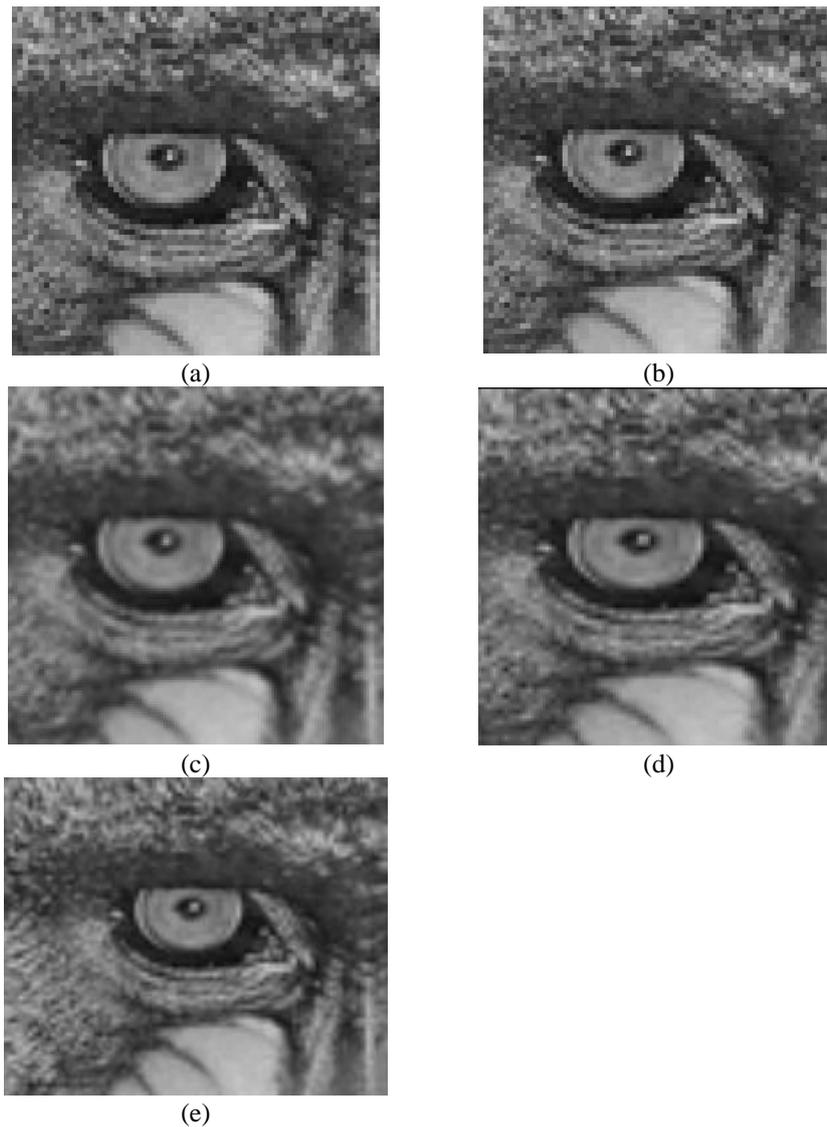


Fig. 6. Baboon (a) Original image 256x256, (b) KNN 512x512 interpolated image, (c) Bilinear 512x512 interpolated image, (d) ELA 3 + 3 Interpolated Image, (e) ELA 5 + 5 Interpolated Image.

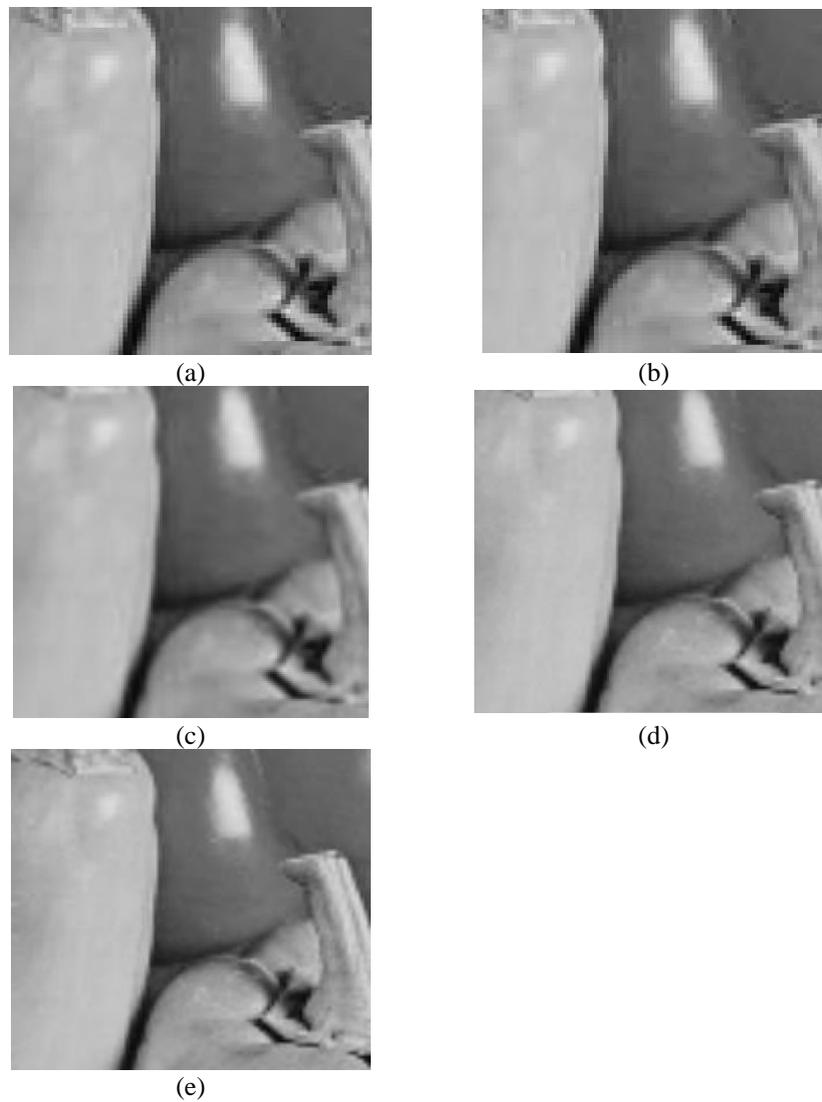


Fig. 7. Peppers (a) Original image 256x256, (b) KNN 512x512 interpolated image, (c) Bilinear 512x512 interpolated image, (d) ELA 3 + 3 Interpolated Image, (e) ELA 5 + 5 Interpolated Image.

The Figures 8 and 9 show an application of the algorithms ELA 3 + 3 and ELA 5 + 5 in video surveillance images, where the image resolution increases in the way that the fine details do not lose as well as the preservation of edges and details.

Table 3. Criteria results for Lena, Baboon and Peppers.

| KNN Interpolation | | | | Bilinear Interpolation | | |
|-----------------------|-------|--------|---------|------------------------|--------|---------|
| Judgments | Lena | Baboon | Peppers | Lena | Baboon | Peppers |
| MAE | 3.73 | 4.74 | 2.55 | 3.39 | 4.60 | 2.45 |
| MSE | 98.49 | 143.96 | 68.28 | 72.86 | 129.45 | 48.96 |
| PSNR (db) | 28.20 | 26.55 | 29.79 | 29.51 | 27.00 | 31.57 |
| ELA 3+3 Interpolation | | | | ELA 5+5 Interpolation | | |
| Judgments | Lena | Baboon | Peppers | Lena | Baboon | Peppers |
| MAE | 2.85 | 4.03 | 2.16 | 2.84 | 4.04 | 2.14 |
| MSE | 44.75 | 117.43 | 32.25 | 44.70 | 158.09 | 30.25 |
| PSNR (db) | 31.62 | 27.16 | 33.04 | 31.63 | 26.15 | 33.24 |



(a)



(b)



(c)

Fig. 8. Image Video surveillance, (a) Original Image, (b) Interpolated Image by ELA 3+3, (b) Interpolated Image by ELA 5+5.

Performing zoom of the face of the robber we can identify the details and edges preserved agree to the interpolated techniques used.

5 Conclusions

Different interpolation Methods were analyzed, where the method of nearest neighbor interpolation is a basic method that require a low time processing compared to the other

methods used because only is considered one a pixel, which is the closest to the interpolated point. Disadvantage with this method, is the loss of the image details such as preserving edges.

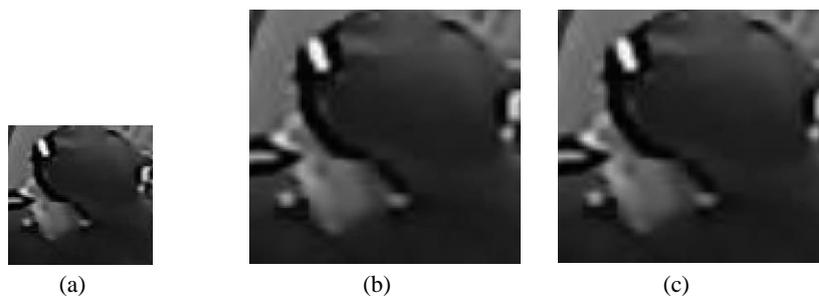


Fig. 9. Image Video surveillance (a) Original Image zoom 400%, (b) Interpolated Image by ELA 3+3 zoom 400%, (c) Interpolated Image by ELA 5+5 zoom 400%.

Contrary to this, the bilinear algorithm takes into account the pixel values surrounding the pixel to be interpolated, a window of 2x2 pixels is used, the result is an image with soft edges, but it requires more processing time compared to the nearest neighbor interpolation.

Algorithms that use fuzzy logic techniques (ELA 3 + 3 and ELA5 + 5) waste more processing time, due to the steps that must be performed to fuzzify and defuzzify the values to interpolate pixels, the main advantage presented is to have an image with more edges delineated and defined.

Acknowledgement. The authors would thank to the Instituto Politécnico Nacional de Mexico (IPN) and CONACyT for their help and support to develop this research work.

References

1. H. Demirel, G. Anbarjafari, S. Izadpanahi: Improved motion-based localized super resolution technique using discrete wavelet transform for low resolution video enhancement. In: Proc. 17th EUSIPCO, Edinburgh, U.K., pp. 1097–1101 (Aug. 2009)
2. T. Celik, C. Direkoglu, H. Ozkaramanli, H. Demirel, M. Uyguroglu: Region-based super-resolution aided facial feature extraction from low-resolution video sequences. In: Proc. IEEE ICASSP, Philadelphia, PA, vol. II, pp. 789–792 (Mar. 2005)
3. H. Demirel, G. Anbarjafari: Satellite image resolution enhancement using complex wavelet transform. IEEE Geosci. Remote Sens. Lett., 7(1), 123–126 (Jan. 2010)
4. L. Yi-bo, X. Hong, Z. Sen-yue: The wrinkle generation method for facial reconstruction based on extraction of partition wrinkle line features and fractal interpolation. In: Proc. 4th ICIG, Aug. 22–24, pp. 933–937 (2007)
5. Y. Rener, J. Wei, C. Ken: Downsample-based multiple description coding and post-processing of decoding. In Proc. 27th CCC, Jul. 16–18, pp. 253–256 (2008)
6. C. B. Atkins, C. A. Bouman, J. P. Allebach: Optimal image scaling using pixel classification. In: Proc. ICIP, Oct. 7–10, vol. 3, pp. 864–867 (2001)

Samuel Souverville, Jorge A. Rosales, Francisco J. Gallegos, Mario Dehesa, et al.

7. Nancy G. La Vigne, Samantha S. Lowry: Evaluating the uses of public surveillance cameras for crime control and prevention. Urban Institute (September 2011)
8. E.E. Kerre, M. Nachtgael: Fuzzy Techniques in Image Processing. 3 ed, Editorial Alfaomega Ra-Ma, Mexico (June 2011)
9. Transactions D: Computer Science & Engineering and Electrical Engineering, Scientia Iranica, International Journal of Science and Technology (February 2011)
10. H. H. Hsiao, J. H. Jeng: Modified De-interlacing method based on edge Direction. (2006)
11. P. Brox: Progressive scan conversion based on edge-dependent interpolation using fuzzy logic. (2007)
12. A. Rosales, F. Gallegos: Procesamiento de Imágenes y Video Multiespectrales y Multicanales. (2011)

Clustering Ensemble Selection Considering Quality and Diversity

Roham Ranjbar, Hamid Parvin*, and Farhad Rad

Department of Computer Engineering, Yasooj Branch,
Islamic Azad University, Yasooj,
Iran

parvin@iust.ac.ir

Abstract. Information clustering means classifying information or partitioning some samples in clusters such that samples inside each cluster have maximum similarity to each other and maximum distance from other clusters. As clustering is unsupervised, selecting a specific algorithm for clustering of an unknown set may fail. As a consequence of problem complexity and deficiencies in basic clustering methods, most of studies have focused on ensemble clustering methods in recent years. Diversity in initial results is one of the most important factors which may affect final quality of the results. Moreover, the quality of primary results affects the quality of final results. Both factors have been investigated in recent studies on clustering. Here, a new framework is proposed which is used for improving clustering efficiency and it is based on use of a subset of initial clusters. Selection of this subset plays a significant role in performance of the scheme. The subset is selected using two intelligent methods. The main idea in these methods is utilizing stable clusters through intelligent search algorithms. Two stability factors are utilized for cluster evaluation. One of these two stability factors is based on mutual information and the other one is based on Fisher measure. Finally, the selected clusters are added using several final combining methods. Practical results of several standard data sets demonstrate that the proposed method may improve combination clustering method significantly.

Keywords: Clustering combination, local optimization, diversity.

1 Introduction

Clustering is a branch of unsupervised learning. It is an automatic process through which samples are divided into groups with similar members which are called clusters. Thus, cluster is a set of objects which are similar to each other while they are different from objects inside other clusters. Various criteria might be considered for

*Corresponding author.

similarity, for instance, distance could be used for clustering and objects which are closer could be clustered as one cluster; it is called distance based clustering. In unsupervised methods no objective variable is defined and data-mining algorithm searches correlations and structures of all variables. Clustering is the most prominent example of unsupervised data mining.

As mentioned before clustering is putting similar objects together; however, it must be found out how a clustering system is evaluated. As a matter of fact there is not any absolute measure for determining the best clustering method and it depends on the problem and user's opinion; nevertheless, there are various measures to determine a good clustering which may help the user to achieve a proper clustering. Some of these measures are explained in efficiency measures section.

In statistics and machine learning, clustering or cluster analysis is the procedure of grouping similar objects. The clustering problem might be introduced in two ways: 1) a $n \times n$ dissimilarity (similarity) matrix is given, 2) a $n \times d$ matrix is given where each row defines an object. The output of the algorithm could be in two forms: 1) grouping the objects to separated sets 2) hierarchical clustering which finds a tree for division of objects. The algorithms of first group are faster. It must be noticed that this paper does not deal with hierarchical clustering. Each clustering algorithm cluster the data in a unique way as it focuses on a specific aspect of the data. Thus, it is necessary to combine such algorithms, take advantage of a few algorithms and provide optimum results. Actually, the main goal of ensemble clustering is searching for the best clusters obtained via combining other algorithms [1, 2]. Combination clustering may provide better results from stability, flexibility and robustness perspectives [2-4]. To sum up, combination clustering includes the following steps: a) generating different subsets from whole given data, b) initial clustering based on applying various clustering algorithms on the subsets generated from main samples. c) combining the results obtained from primary clustering methods to achieve final clustering. There are two important issues regarding combination clustering: 1) diversity of various clustering algorithms such that each of them focuses on a specific characteristic of the data, 2) the combining algorithm which provides the final results. To address the first issue the following methods might be exploited: 1) using different clustering algorithms [5], 2) changing initial values or other parameters of the clustering algorithm [4, 6], 3) selecting some data features or generating new features [1, 3, 7], 4) dividing the main data to different and separated subsets [8, 9, 10, 11, 12, 13]. The second issue is vastly investigated to find algorithms for combining the results [14, 15, 16]. But the proposed methods have been static ones so far rather than dynamic methods. In this paper a dynamic approach is provided. Despite, information classification which has a supervisor and training set, in information clustering the data set is completely unknown. Lack of supervisor and training set makes it difficult to introduce modern and smart clustering methods with high efficiency. One of the methods which might be used to achieve smart information classification is considering diversity concept.

Diversity in information classification means that if one classifier has errors in some samples, we look for other classifier which has errors in samples different from errors of the first classifier. As a result the classifiers provide better results. Lack of training set has deprived information clustering methods from such technique. Here, it is tried to include diversity concept in information clustering [13, 21, 22]. Diversity

concept has been utilized widely in recent research works [3, 13, 23, 24]. The main goal of recent combination clustering methods is examining data set from different perspectives and it has not been investigated whether the generated diversity is useful or not. Indeed, it is difficult due to unsupervised nature of clustering problem. However, practical results have demonstrated that generating diversity in primary clusters usually leads to better results [25]. Azimi [26] has shown that in some data sets more diversity does not necessarily increase final precision. In this paper, diversity and clustering quality have been simultaneously emphasized.

Routing in computer networks has played a special role in recent years. The cause of this is the role of routing in a performance of the networks. The quality of service and security is one of the most important challenges in routing due to lack of reliable methods.

2 Heuristic Search Methods

Increasing complexity of optimization problems has necessitated novel search methods. To address this issue, heuristic methods have been developed in many fields as a powerful optimization and search tool in recent decades. Their wide ranged applications, simple use and the capability of obtaining near to optimum solution has made these methods successful ones. In this section two heuristic algorithms which are utilized in this paper are briefly explicated.

Genetic algorithm is a scheme which considers the natural evolution of creatures [4]. It tries to imitate evolution process using computer algorithms. The most essential principle of evolution is inheritance. John Holland innovated genetic algorithm for the first time during 70s according to evolution theory. The algorithm exploits the same principles that natural evolution uses [4] to improve solutions of an optimization problem. There are two important operators in genetic algorithms which make the solutions chaotic to exit probable local optimums. One of these operators is crossover through which genetic algorithm generates solutions. Another operator called mutation is able to provide new values for bits which do not exist in the parents. Mutation guarantees genetic diversity and pushes the search to new domains.

Simulated annealing is an optimization method which is similar to the process through which metals are heated and then slowly annealed [7]. It is suitable for simple objective functions with one local bound point (minimization or maximization problems). For complex functions (for example for minimization problems) the local optimum points might be completely different from global optimum point. In such cases the optimization model will not be able to provide optimum solution. SA utilizes stochastic release so that it exits local minima points.

SA procedure starts from a possible solution such as q_0 (a real vector which shows all decision variables) and its corresponding objective function $J_0 = J(q_0)$. A new solution q_1 with objective function $J_1 = J(q_1)$ is randomly selected and evaluated among the neighboring area of the initial solution. The amount of variation in decision variable is usually known. Random nature is due to direction or dimension of changes (e.g. x variation might be known but its direction might be unknown). If the new solution has lower objective function $J_1 < J_0$ (for minimization problem), it is accepted and the search process is transferred to point q_1 . If the new solution is not

better than current solution ($J_1 \geq J_0$) it may be selected or rejected which depends on the following acceptance probability.

$$p_{acc} = e^{-\frac{J_1 - J_0}{T}}$$

3 Literature Review

Combining clustering methods is more difficult than combining supervised classifications. In clustering data set is not known. It is difficult to propose high efficiency smart clustering methods due to lack of supervisor and training set. Combination clustering is a method of clustering which is resulted from combining different clustering methods. Two main steps of generating a combination of initial clustering methods are generating each clustering method and using a function or mechanism to combine their results to obtain the final results.

Since the final result is a combination of initial clustering results, the more different initial results lead to better final result. As a matter of fact, if data are investigated from more different aspects the final result would more precise. There are various methods for generating diversity in combination clustering including different clustering algorithms, changing initial values or other clustering parameters, selecting some data features or generating new features and dividing main data to different and separated subsets. In the presented methods the main goal is to examine data set from different points of view. They have not investigated whether the generated diversity is useful or not [17].

Usually, most combination clustering methods use k-mean algorithm as their initial clustering method [12, 13, 18]. It has been shown that in some cases other clustering methods might be more beneficial considering behavior of each data set [1, 3]; nonetheless, k-mean algorithm has been the first choice due to its simplicity and appropriate ability.

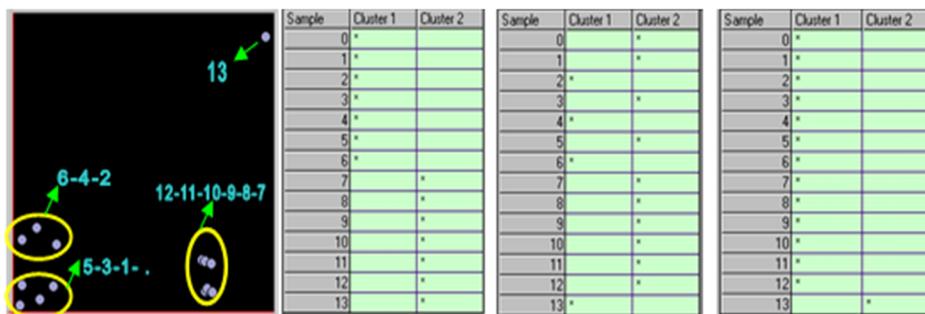


Fig. 1: Initial samples in k-mean algorithm. Figures from left to right: 1) space view of 14 samples, 2) results obtained using 1 and 8 initial samples, 3) results obtained from 2 and 3 initial samples and 4) results obtained from 1 and 13 initial samples

Another method for increasing diversity is changing initial parameters of clustering algorithms. For instance changing number of clusters in k-means or changing seed points significantly affect diversity [15]. In the following figure, the

effect of initial samples on final clustering is clear. In this figure, the distribution of samples is shown and the results of three different runs of algorithm with three different starts are depicted [1].

Feature selection might be considered as a method for adding diversity to combination clustering as well. Thus, another solution to increase diversity in combination clustering is using some features of total data set space or generating new features [13]. However, in information clustering selecting a subset of features has not been noticed due to the unsupervised nature of the problem. It is mostly tried to generate new features. There are several methods for feature generation in ensemble clustering [19] among which the simplest is data normalization. In fact it is shown that each data set achieves better behavior using one normalization method. As a result in many methods presented for information clustering, raw value of results are reported.

4 Consensus Function

Combining the first ensembles and obtaining the final result is one of the most important steps of combination clustering. There are diverse methods for combining the results of initial ensembles some of which are introduced in this section. Then, the proposed method is presented.

1- Hyper-graph based method

In hyper-graph partitioning, first off, the combination clustering problem is converted to a graph partitioning problem. Afterwards, the problem is solved using graph partitioning algorithms. Clusters are denoted by hyper edges of a graph. Graph vertexes correspond to samples which must be clustered. The problem is dividing this graph and generating k separated partitions each of which belongs to a cluster. There are three different algorithms in this group of methods including HGPA, CSPA and MCLA [1, 3].

1-1- CSPA

In CSPA feature space of data points is mapped to correlation feature space of hyper-graph. Then, a hyper graph minimum cut algorithm similar to METIS is applied to the data points. According to assumptions of this method more data points in one cluster in primary portioning means that data points are more probable to naturally belong to one cluster. CSPA is the simplest heuristic method. Its computational complexity is $O(kN^2M)$ where k is the number of clusters, N is the number of data points and M is the number of domains. The computational complexity of two other methods is lower than CSPA.

1-2- HGPA

HGPA consider vertexes as data points. Also, clusters which are resulted from initial portioning are assumed to be hyper edges. Now a hyper-graph minimum cut algorithm such as METIS is applied to hyper-graph for separating vertexes to k different components. Its computational complexity is $O(kNM)$ where k is the number of clusters, N denotes the number of data points and M is the number of domains.

1-3- MCLA

MCLA algorithm partitions the cluster resulted from initial partitioning. Afterwards, it utilizes a voting mechanism to generate set partitions. Clustering is

done using METIS. Its computational complexity is $O(k^2NM^2)$ where k , N and M are similar to previous methods. For more details regarding hyper graph based methods interested reader is referred to [3].

2- Voting method

This is actually majority of vote method. The cluster to which each sample belongs is determined according to majority votes. The main problem of this method is matching of cluster numbers in different runs which imposes heavy computational overhead on the algorithm. This computational overhead has caused this method to be unpopular among various consensus function methods [2, 8, 23].

3- Co-association matrix

Consider D as a data set consisting of N points (samples) in a d dimensional space. The input data might be considered either as a $N \times d$ pattern matrix or a dissimilarity $N \times N$ matrix. Assume that $X = \{X_1, X_2, \dots, X_{B_1}\}$ is a subset of available samples extracted from initial samples. All algorithms generate $P = \{P_1, P_2, \dots, P_{B_1}\}$ when they are applied to samples inside X . Each P_i is a set of clusters i.e. $P_i = \{C_1^i \cup C_2^i \dots \cup C_{k(i)}^i\}$ and $X_i = C_1^i \cup C_2^i \dots \cup C_{k(i)}^i$ such that $k(i)$ is the number of clusters in i th ensemble.

The first base algorithm which is utilized is k -means algorithm. At first step, k -means algorithm is executed on $X = \{X_1, X_2, \dots, X_{B_1}\}$ so that the co-association matrix could be derived as follows using generated P_i s.

$$Co - associatio n(xy) = \sum_{i=1}^{B_1} \lambda(P_i(x), P_i(y))$$

where

$$\lambda(a, b) = \begin{cases} 1 & \text{if } a = b \\ 0 & \text{if } a \neq b \end{cases}$$

$\lambda(P_i(a), P_i(b))$ acquires 1 if in P_i combination a and b are located in the same cluster and it would be zero otherwise. B_1 denotes the number of subsets i.e. the number of times that k -mean base algorithm is repeated. When co-association matrix is obtained, final clusters are extracted from co-association matrix employing a simple hierarchical algorithm such as average link (AL).

5 Proposed Procedure

Clustering combination is more difficult than combining supervised classifications. Despite classification problem which has supervisor and training set, in clustering there is not any information available about data set. It is difficult to present high efficiency modern and intelligent methods due to lack of supervisor and training set. Furthermore, when labeled training data is not available, problem of correspondence between cluster labels in different partitions of a combination arises. Recent clustering methods mainly try to examine dataset from different perspective while they mostly ignore whether the diversity is useful or not. Indeed, it is difficult to do so as a result of unsupervised nature of clustering. Although experimental results have

shown that diversity improves clustering in most cases [29], Azimi [30] demonstrated that in some data sets more diversity does not necessarily increase precision. Since there is not any true labels (supervisor), clustering is one of the most difficult and ambiguous concepts in artificial intelligence; as all partitions of data might be considered to be correct. One of the problems is shortage of a precise and absolute measure for clustering which could be optimized to obtain the best clustering.

In this section a novel scheme is proposed which optimizes diversity while taking precision into account. For this purpose a set (combination) of initial clusters called reference set or RefSet is generated. The size of RefSet combination is |RefSet| which denotes the number of its elements. It is worth mentioning that RefSet_{*i*} denotes the *i*th member of this combination. Afterwards, another combination called main combination or combination is generated. It must be mentioned that combination_{*i*} denotes *i*th member of combination. Then, stability is calculated for each combination_{*i*} where *i* changes from 1 to B. The stability of partition combination_{*i*} is the average of its similarity in reference set. The similarity of two partitions is calculated via Fisher measure equation. This measure which is utilized here to assess a partition is called F-measure.

$$FM(P, L) = \max_{\tau} \sum_{i=1}^{K_P} \frac{2 \times N_i^P \times \left(\frac{N_{i\tau(i)}^{PL}}{N_i^P} \times \frac{N_{i\tau(i)}^{PL}}{N_{\tau(i)}^L}\right)}{N \times \left(\frac{N_{i\tau(i)}^{PL}}{N_i^P} + \frac{N_{i\tau(i)}^{PL}}{N_{\tau(i)}^L}\right)}$$

where K_P is the number of clusters in partition P; N_i^P denotes the number of data existing in *i*th cluster of partition P; N_j^L represents the number of data in *j*th cluster of partition L; N_{ij}^{PL} is the number of data which are in both *i*th cluster of P and *j*th cluster of L. N is the total number of data and τ is a permutation of numbers from 1 to N. If partition P and label L are completely similar, FM has its maximum value i.e. 1 and it is zero in case of complete dissimilarity.

The stability of partition combination_{*i*} is derived as follows.

$$Stability(\text{combination}_i) = \frac{1}{|RefSet|} \sum_{j=1}^{|RefSet|} FM(\text{combination}_i, RefSet_j)$$

Then, clustering ensembles are searched with respect to stability and diversity so that the most stable and diverse clustering is found.

When combinations are selected according to their stability, an evolutionary algorithm is employed to select a subset of initial combinations. This algorithm is explicated in the following. These evolutionary algorithms include a bit-string chromosome whose length is the total number of combinations inside second combination. Each genes of this chromosome may acquire 1 or 0. When a gene is 1 it means that the combination with corresponding number is selected and zero means the combination is not selected. In other words a 1 in *i*th gene means that SE_{*i*} is selected while a 0 is *m*th gene means that *m*th combination is not selected. To calculate fitness function of evolutionary algorithm, diversity of selected combinations needs to be derived.

To calculate efficiency of the chromosome, the following equation is used which demonstrates the amount of diversity.

$$FitnessFunction = 0.5 - \frac{\sum_x \sum_y abs(Co(x, y) - 0.5)}{N^2}$$

6 Simulation Results

In this paper combination clustering was investigated from other aspects. Contrary to previous methods which suggest a constant method with constant characteristics for all types of datasets (even if diversity was considered) [26], our proposed method changes its behavior dynamically according to the data set. A dynamic approach changes its behavior with respect to sample distribution in each dataset. As each clustering method has its pros and cons, it is not possible to choose a specific method for a specific data set [1]. Our proposed method tries to choose those initial combinations which lead to the best initial results for a specific data set. K-means algorithm is considered as base clustering algorithm. Besides, in different runs of this algorithm the number of clusters is assumed to be integer. Furthermore, this number is considered as a parameter. The experimental results provided in next section confirm the performance of our method.

Table 1. Results

| | NMI | | FM | | AR | |
|--------------|-------------------|---------------|-------------------|---------------|-------------------|---------------|
| | Proposed Ensemble | Full Ensemble | Proposed Ensemble | Full Ensemble | Proposed Ensemble | Full Ensemble |
| Glass | 15.82 | 15.77 | 28.73 | 28.17 | 8.72 | 8.62 |
| BreastCancer | 39.90 | 35.58 | 48.39 | 47.34 | 44.26 | 40.69 |
| Wine | 21.97 | 21.44 | 35.71 | 35.48 | 18.58 | 18.56 |
| Iris | 38.91 | 37.1 | 46.01 | 44.61 | 39.3 | 35.82 |

In this section the results of applying the proposed algorithm to some datasets are reported. 4 datasets which are popular in literature are investigated; so, it would possible to compare the proposed method to other methods. The results are reported in Table 1.

7 Conclusion

The proposed method opens up new horizons in clustering algorithms. Including fuzzy concept in clustering combination is a promising idea. Data normalization is

necessary when Euclidean distance is exploited. Since using data normalization algorithm does not guarantee improvement in clustering, proposed clustering methods present their reports according to raw data. Therefore, another idea which might be considered in future studies, is finding a dynamic method for assigning proper normalization method to each data set. The most prominent factor which leads to considerable improvement in the proposed method is finding an intelligent method for generation of initial results. It must be able to generate initial results which cover deficiencies of other initial results.

Resources

1. Azimi J.: Investigating Diversity in combination Clustering, Master thesis, Iran University of Science and Technology (2007)
2. Alizadeh H.: Combination Clustering Based on a Subset of Initial Results. Master thesis, Faculty of Computer Engineering, Iran University of Science and Technology.
3. Jain A., Murty M. N., Flynn P.: Data clustering: A review. *ACM Computing Surveys*, 31(3):264–323 (1999)
4. Faceli K., Marcilio C.P. Souto D.: Multi-objective Clustering combination, *Proceedings of the Sixth International Conference on Hybrid Intelligent Systems* (2006)
5. Strehl A., Ghosh J.: Cluster combination - a knowledge reuse framework for combining multiple partitions. *Journal of Machine Learning Research*, 3(Dec):583–617 (2002)
6. Melanie M.: *An Introduction to Genetic Algorithms*. A Bradford Book The MIT Press, Cambridge, Massachusetts. London, England, Fifth printing (1999)
7. Davis, T.: *The Handbook of Genetic Algorithms*. Van Nostrand Reinhold Co., New York, NY (1991)
8. Muni, D., Pal, N., Das, J.: Genetic programming for simultaneous feature selection and classifier design. *IEEE Trans Syst Man Cybern B Cybern* 36(1):106–117 (2006)
9. Aarts E. H. L., Korst J.: *Simulated Annealing and Boltzmann Machines*, John Wiley & Sons, Essex, U.K. (1989)
10. Fred, A., Jain, A. K.: Data Clustering Using Evidence Accumulation. In: *Proc. of the 16th Intl. Conf. on Pattern Recognition, ICPR02, Quebec City*, pp. 276–280 (2002)
11. Parvin H., Alizadeh H., Minaei-Bidgoli B.: A New Method for Constructing Classifier combination. *International Journal of Digital Content: Technology and its Application, JDCTA* (2009)
12. Parvin H., Alizadeh H., Minaei-Bidgoli B.: Using Clustering for Generating Diversity in Classifier combination. *International Journal of Digital Content: Technology and its Application, JDCTA*, Vol. 3, No.1, pp. 51–57 (2009)
13. Alizadeh H., Minaei-Bidgoli B., Amirgholipour S.K.: A New Method for Improving the Performance of K Nearest Neighbor using Clustering Technique. *International Journal of Convergence Information Technology, JCIT* (2009)
14. Topchy, A., Jain, A.K., Punch, W.F.: Combining Multiple Weak Clusterings. In: *Proc. 3d IEEE Intl. Conf. on Data Mining*, pp. 331–338 (2003)
15. Fred A., Lourenco A.: Cluster combination Methods: from Single Clusterings to Combined Solutions. *Studies in Computational Intelligence (SCI)*, 126, 3–30 (2008)

16. Ayad H.G., Kamel M.S.: Cumulative Voting Consensus Method for Partitions with a Variable Number of Clusters. *IEEE Trans. on Pattern Analysis and Machine Intelligence* 30(1):160–173 (2008)
17. Minaei-Bidgoli B., Topchy A., Punch W.F.: Combinationes of Partitions via Data Resampling. In: *Proc. Intl. Conf. on Information Technology, ITCC 04, Las Vegas (2004)*
18. Alizadeh H., Amirgholipour S.K., Seyedaghaee N.R., Minaei-Bidgoli B.: Nearest Cluster combination (NCE): Clustering combination Based Approach for Improving the performance of K-Nearest Neighbor Algorithm. In: *11th Conf. of the International Federation of Classification Societies, IFCS09, March 13–18 (2009)*
19. Mohammadi M., Alizadeh H., Minaei-Bidgoli B.: Neural Network combination using Clustering combination and Genetic Algorithm. In: *Intl. Conf. on Convergence and hybrid Information Technology, ICCIT08, Nov. 11-13, IEEE CS (2008)*
20. Barthelemy J.P., Leclerc B.: The median procedure for partition. In: *Partitioning Data Sets, AMS DIMACS Series in Discrete Mathematics, Cox, I. J. et al. (eds.), 19, pp. 3–34 (1995)*
21. Fern, X., Brodley, C. E.: Random Projection for High Dimensional Data Clustering: A Cluster combination Approach. In: *Proc. 20th Int. conf. on Machine Learning, ICML (2003)*
22. Dudoit S., Fridlyand, J.: Bagging to improve the accuracy of a clustering procedure. *Bioinformatics*, 19(9):1090–1099 (2003)
23. Fischer B., Buhmann J.M.: Bagging for path-based clustering. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp.1411–1415 (2003)
24. Fred A., Jain A.K.: Robust data clustering. In: *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR, USA, vol. II, pp. 128–136 (2003)*
25. Fred A.L., Jain A.K.: Combining Multiple Clusterings Using Evidence Accumulation. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 27(6):835–850 (2005)
26. Fred A., Jain A.K.: Learning Pairwise Similarity for Data Clustering. In: *Proc. of the 18th Int. Conf. on Pattern Recognition (ICPR'06) (2006)*
27. Kuncheva L.I., Whitaker C. J.: Measures of diversity in classifier combinations. *Machine Learning (2003)*
28. Kuncheva L.I., Hadjitodorov S.: Using diversity in cluster combinations. In: *Proc. of IEEE Intl. Conference on Systems, Man and Cybernetics*, pp. 1214–1219 (2004)
29. Baumgartner R., Somorjai R., Summers R., Richter W., Ryner L., Jarmasz M.: Resampling as a Cluster Validation Technique in fMRI. *Journal of Magnetic Resonance Imaging* 11:228–231 (2000)
30. Law M.H.C., Topchy A.P., Jain A.K.: Multiobjective data clustering. In: *Proc. of IEEE Conference on Computer Vision and Pattern Recognition, volume 2, pp. 424–430, Washington D.C. (2004)*
31. Shamiry O., Tishby N.: Cluster Stability for Finite Samples. In: *21st Annual Conference on Neural Information Processing Systems (NIPS07) (2007)*
32. Breckenridge J.: Replicating cluster analysis: Method, consistency and validity. *Multivariate Behavioral research* (1989)
33. Fridlyand J., Dudoit S.: Applications of resampling methods to estimate the number of clusters and to improve the accuracy of a clustering method. *Stat. Berkeley Tech Report. No. 600 (2001)*
34. Levine E., Domany E.: Resampling Method for Unsupervised Estimation of Cluster Validity. *Neural Computation* 13:2573–2593 (2001)

35. Roth V., Lange T., Braun M., Buhmann J.: A Resampling Approach to Cluster Validation. In: Intl. Conf. on Computational Statistics, COMPSTAT (2002)
36. Roth V., Braun M.L., Lange T., Buhmann J.M.: Stability-Based Model Order Selection in Clustering with Applications to Gene Expression Data. In: ICANN 2002, LNCS 2415, pp. 607–612 (2002)
37. Lapointe F.J., Legendre P.: The generation of random ultrametric matrices representing dendrograms. *Journal of Classification* 8(2):177–200 (1991)

Feature Selection for Improvement the Performance of an Electric Arc Furnace

Amado Sánchez Sánchez¹, José Crispín Hernández Hernández¹, Haydee Patricia Martínez Hernández¹, David Ibarra Guzmán², Arturo Contreras Juárez², Arturo Aguila Flores², and Perfecto Malaqías Quintero Flores¹

¹Instituto Tecnológico de Apizaco,
Laboratorio de Investigación en Tecnologías Inteligentes, Apizaco, Tlaxcala,
Mexico

²Universidad Politécnica de Tlaxcala Región Poniente,
Hueyotlipan, Tlaxcala,
Mexico

amado.sanchez1982@gmail.com, josechh@yahoo.com

Abstract. Feature selection has as principal goal to find a representative space of minimal size from original set of larger size. Several research works have been developed on this problem. This paper presents Support Vector Machine-Recursive Feature Elimination (SVM-RFE), Genetic Algorithms (GA), and Differential Evolution (DE) algorithms for feature selection from a database of an Electric Arc Furnace (EAF) for locating variables related to energy consumption. The proposal suggests merging the coefficients generated by LDA and SVM, employing them in RFE to obtain the ranks for each discriminant variables. The measure of accuracy and error rate for each algorithm is presented like a decisive score for choosing the subset obtained by the algorithm with the best performance. The variables selected were adjusted for the EAF control system achieving the reduction of the energy consumption to 3.5% in a steel castings and 1 minute reduction of the connected EAF in a steel casting.

Keywords: Feature selection, electric arc furnace, DEFS, GA, SVM-RFE, LDA, energy consumption

1 Introduction

The impact of feature selection methods applied to data analysis process has been exposed on several research such as medical diagnosis, image recognition, credit to a bank, text classification, where the results allow given a dataset described by N features to find the minimum number n of relevant features for describing the data as well as the original set of features [5, 3].

In this paper, the application of feature selection algorithms on data of an Electric Arc Furnace (EAF) is presented in order to identify the variables that

discriminate the merger process and adjust them to the melting process for improvement of the performance in steel casting.

The steel melting process in electric arc furnace is too complex and its nature consists of N number of variables that require a precise control to achieve high performance. Although electrical energy is the principal resource in EAF, additional chemical energy is supplied; this is added by injecting oxygen, natural gas and graphite in the EAF through multiple injection units adapted to the wall to achieve thermal homogeneity within the furnace. The release of heat generated by the chemical energy ($CO + CO_2 + \Delta E$) is exploited to get to the parts where the arc is less (cold spots).

The electricity needed to generate the arc represents a high cost for the steel-making process, that is, the 80% of the steel mill consumption total. Therefore decreasing by at least 1% of the electricity consumption, the involved cost would be positively impacted.

2 Related Works for Improvement of the Performance of EAF

Some soft computing methods, focused on industrial and biological processes have been applied in EAF. Bernardo et al [2], propose the application of multi-objective algorithm for controlling temperature in a steelmaking process. Erik et al [13], propose using multivariate prediction models (Principal Component Analysis, Multiple Linear Regression, Partial Least Squares), to estimate the properties of scrap and achieving adequate chemical conditions of steel. Miroslaw et al [10], propose a combination of a regression tree with a neural network to optimize the input data, and the use of evolutionary algorithms to find optimal weights to be used in the neural network; both were to predict the temperature of an EAF. Khan et al [8], they offer excellent proposal for the analysis of information from a EAF using data mining in order to find rules that could be used for the development of automated intelligent systems, at the same time they introduced the concept of chemical energy and perform an analysis on the behavior of carbon, gas and oxygen through the development of an energy balance model, achieving at various stages of the merger process, determine the average gas required into the EAF.

Each work has contributed through the passage of time to improve the performance of the EAF's as shown in figure 1.

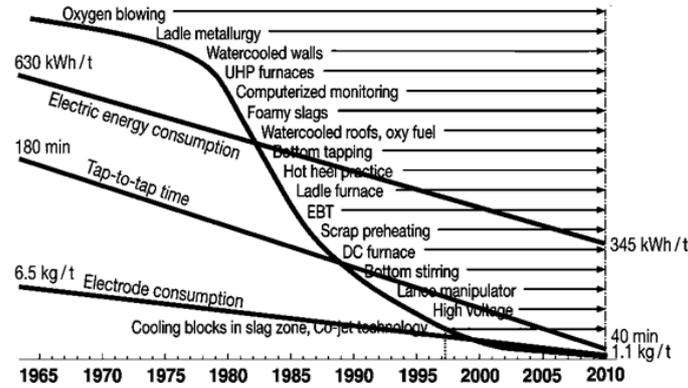


Fig. 1. Several innovations have been introduced on EAF to improve its performance. (Extracted from [8])

The figure 1 depicts the performance of EAFs. During previous years, different innovations were introduced to EAFs. These technological developments caused significant improvements in different key performance indicators.

3 Methods for Feature Selection Used

There are two main approaches to data reduction: feature transformation and features selection. The feature selection algorithms are organized into three main categories: filter methods, envolventes methods, and embedded methods [9].

The first method, select the features without running any learning algorithm, they are evaluated through the intrinsic properties of the data. In envolventes methods, learning algorithms are used to evaluate the selected subset, based on learning classifier for each of the candidates of the subset. Embedded methods incorporate learning algorithms and search features in an optimization problem. Feature selection known as the selection of variables, is the problem of selecting a subset of the original features, in contrast to methods based on the transformation wich, allowing the modification of the input characteristics to a new feature space; in features selection, the original representation variables do not change [9, 6, 11].

According with the literature, the most popular methods for feature selection are: SVM-RFE, GA, DE and LDA for reduction data.

3.1 SVM-RFE Algorithm

The *SVM – RFE* algorithm is a technique that combines Support Vector Machine and Recursive Feature Elimination, methods developed by V. Vapnik et al, and I. Guyon et al, respectively. These predictive models have the capacity

to generate knowledge for the interpretation of the obtained solution [16, 7]. The general esqueme of the algorithm to SVM-RFE is illustrated in figure 2.

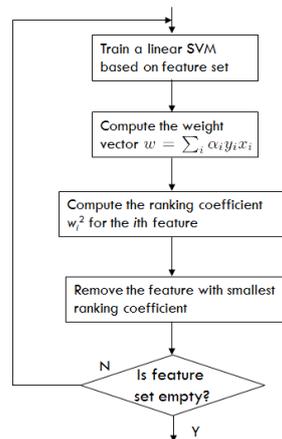


Fig. 2. SVM-RFE general scheme.

Where x is the feature expression vector of a sample, x_i is that of sample i in the training set ($i = 1, 2, \dots, n$), $y_i \in \{+1, -1\}$ is its corresponding class label, $w = \sum_{i=1}^n \alpha_i y_i x_i$, is the vector of weights of the features, and b is a scalar offset. The α_i 's and b are estimated from the training set. Only those samples closest to the separating boundary (called support vectors). The criterion (w_i), estimates the effect of eliminating a feature in an objective function, but it becomes a suboptimal procedure when it decides to eliminate many features. Performing these reductions are necessities in some cases for obtain a small subset of features. For more details see [7].

3.2 Genetic Algorithms

Genetic algorithms (GA) are adaptive search techniques, based on the analogy with biology, in which a set of possible solutions evolves via natural selection. Genetic Algorithms were introduced by John Holland in 1975, based on natural selection proposed by Charles Darwin; ie, they mimic the mechanisms of gene duplication and natural selection. In nature individuals compete among themselves to achieve survival, genes of the fittest individuals are propagated to subsequent generations, including on occasion the children adapt better parents. So species evolve generation after generation. The set of all genes are encoded in a string of values, called chromosome. In the early works of John Hollan coding is performed with a string of zeros and ones. And in fact currently representations

are coded with an integer, real or float value which allows the development of genetic operators can be made more specific [3, 12, 18].

The general scheme of an simple genetic algorithm is depict in the figure 3:

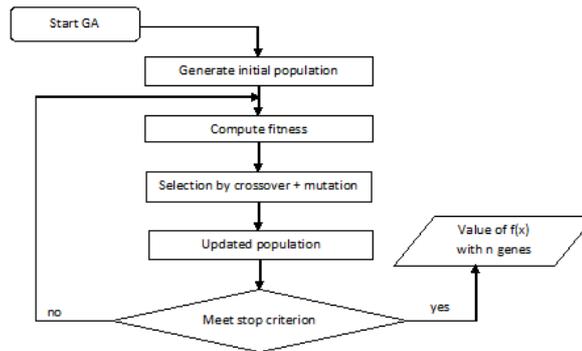


Fig. 3. General scheme of Genetic Algorithm

In carrying out the genetic algorithm for selecting features, it must; firstly, ensure that the strings are randomly generated to build the initial population. Each string represents a subset of features and values in every position in the string are coded as either the presence or absence of a particular feature. Then fitness must be calculated, which is a measure indicating how well a set of characteristics survives evaluation criteria specified for each subset. Best feature subsets are more likely to be selected to form a new subset through a crossing or mutation. The mutation changes some values (by adding or removing features) in a random subset. Crosses made to the different characteristics of a pair of subassemblies are combined into a new subset. This feature selection based on genetic algorithm is an iterative process in which; each successive generation is generated by applying genetic operators for members of the current generation. Thus, good subsets are evolved over time until stopping criteria are met.

3.3 Differential Evolution

Differential evolution (DE) is a simple optimization method that has parallel, direct search, easy to use, good convergence, and fast implementation properties. The first step in the DE optimization method is to generate a population of NP members each of D-dimensional real-valued parameters, where NP is the population size, and D represents the number of parameters to be optimized [15].

The general squeme is depict in the figure 4.

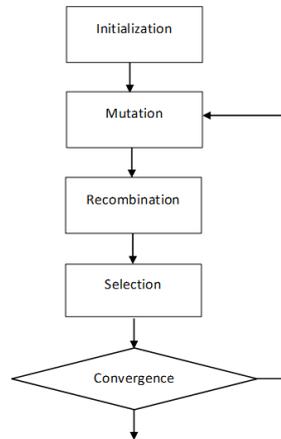


Fig. 4. General esqueme of Differential Evolution

Feature selection with differential evolution (DEFS) is rooted in the general DE algorithm . The main goal of DEFS is to select a subset of features with a low rate of misclassification or high precision in the classification [17]. Therefore the error rate is used as a fitness function; ie, a minimization function:

$$Fit = errorRate = \frac{FP + FN}{TP + TN + FP + FN} \quad (1)$$

where FP, FN, TP and TN are the measurements for false positives, false negative, true positives and true negatives, respectively.

Feature selection plays a central role in the data analysis process since irrelevant features often degrade the performance of algorithms devoted to data characterization, rule extraction and construction of predictive models, both in speed and in predictive accuracy.

4 Methodology Proposal

The propose is a method to reduce the initial dimension of EAF database to select a subset of discriminant variables related with the energy consumption. The model involves the SVM-RFE that guides the feature elimination process. The proposal suggested to merge the coefficients generated by LDA and SVM and employ them in RFE to obtain the ranks for each discriminant variables in the database EAF.

4.1 Linear Discriminant Analysis

LDA is one of the most commonly used technique for data classification and dimension reduction [1].

LDA considers maximizing the following objective:

$$J(w) = \frac{w^T S_B w}{w^T S_W w} \quad (2)$$

where S_B is the between classes scatter matrix and S_W is the within classes scatter matrix. Due to the fact that scatter matrices are proportional to the covariance matrices, J is defined using covariance matrices. The definitions of the scatter matrices are:

$$S_B = \sum_c (\mu_c - \bar{x})(\mu_c - \bar{x})^T \quad (3)$$

$$S_W = \sum_c \sum_{i \in c} (x_i - \mu_c)(x_i - \mu_c)^T \quad (4)$$

where \bar{x} is the overall mean of the data cases, c represent the classes, S_B is the scatter of class 1 with respect the scatter of class 2 and hence corresponds to computing the scatter relative to a different vector. An important property to notice about the objective J is that is is invariant w.r.t. rescalings of the vectors $w \rightarrow \alpha w$. Hence, is possible choose w such that the denominator is simply $w^T S_W w = 1$, since it is a scalar itself [4, 14].

The coefficients of the eigen vector calculated by LDA are used to evaluate the relevancy of each feature for class discrimination. LDA obtains the projection vector w , wich is used in RFE algorithm.

4.2 SVM Weight Vector for Classification

Support vector machines are an example of a linear two-class classifier. The data for a two-class learning problem consist of objects labeled with one of two labels corresponding to the two classes; for convenience we assume the labels are +1 (positive examples) or -1 (negative examples).

A concept required for defining a linear classifier is the dot product between two vectors, defined as $w^T = \sum_i w_i \cdot x_i$. A linear classifier is based on a linear discriminant function of the form:

$$f(x) = w^T x + b \quad (5)$$

The vector w is known as the weight vector, and b is called the bias. Consider the case $b = 0$ first. The set of points x such that $w^T x = 0$, are all points that are perpendicular to w and go through the origin to a line in two dimensions, a plane in three dimensions, and more generally, a hyperplane. The bias b translates the hyperplane away from the origin. The hyperplane divides the space into two according to the sign of the discriminant function $f(x)$ defined in Equation (5). This weigth vector w , guide the RFE method to remove the most discriminant features [16, 7].

4.3 LDA_SVM-RFE

LDA and SVM are proposed to train and to obtain the vector weights and eigen value of each feature by removing one by one with the smallest weight.

In each iteration is used a validation method to train SVM and LDA classifiers to calculate the eigen value vector definite as w_1 and weights vector definite as w_2 of each feature of the data, both are averaged to obtain a single vector w . Thus, RFE find a final subset that contains the most discriminant variables related with the energy consumption. The method is described in the figure 5:

Pseudocode 1: LDA_SVM-RFE

Given the original set with respective labels, set $X=\{x_1y_1, x_2y_2, \dots, x_ny_n\}$

Train LDA and SVM classifiers

Obtain discriminant coefficients of each feature from
LDA and SVM classifier

average the vector w_1 from LDA an vector w_2 from SVM
to obtain a single vector w

validate the new w vector en RFF method

Compute the ranking criteria

Find the feature with smallest ranking criterion

Update feature ranked list

Eliminate the feature with smallest ranking criterion

Feature ranked list r

Fig. 5. Pseudocode for feature selection with LDA_SVM-RFE.

This proposal is based on the SVM-RFE strategy proposed by Guyon et al. [7], where each feature is evaluated for its corresponding coefficient.

5 Experimental Results

This model was evaluated with EAF data, previously these were treated and normalized using the minimum and maximum expression values of each feature given as allows $y_i = x_i - \min(x_i) / (\max(x_i) - \min(x_i))$.

The performance of the algorithms is evaluated by 10-FOLD Cross Validation. The algorithms were coded on Matlab using a laptop intel® core™ i5-2540M CPU @ 2.60GHz y 4.00 GB en RAM.

The total data is 1254 examples with 75 variables. In the table 1 is shown the accuracy and error rate for each algorithm used.

Table 1. Comparison of the algorithms.

| Algorithm | Accuracy (%) | Error Rate (%) | Std. Des. |
|-------------|--------------|----------------|-----------|
| DEFS | 67.69 | 32.31 | 0.0729 |
| GA | 65.27 | 43.73 | 0.0380 |
| SVM-RFE | 84.31 | 15.69 | 0.0019 |
| LDA_SVM-RFE | 84.37 | 15.63 | 0.0043 |

We observe that we model obtain te best accuracy respect DEFS, GA and SVM-RFE. GA has the worse accuracy, barely 65.69% following by DEFS with 67.69%. SVM-RFE is very close with 84.31% and LDA_SVM-RFE offer an accuracy rate of 84.37%.

In this practice, all variables of the EAF were used to confirm the performance, given the dimension of the data. In the table 2 the top 5 ranking variables generated for each algorithm are listed.

Table 2. Top 5 ranking variables determined by the runs algorithms

| Algorithm | Top 5 ranking variables |
|-------------|--|
| DEFS | (1) Kilowatts hour per charge metal tons. (2) Amount of O_2 injected in Melting 1. (3) Time On of EAF during steel casting. (4) °C/minute in heating step. (5) Mega Watts in stage flat bath. |
| GA | (1) Time On of EAF during steel casting. (2) Amount of graphite (kg) injected in Melting 2. (3) Amount of oxygen (m^3) injected in Melting 2. (4) Tons of scrap charged to the EAF, to the Melting 2. (5) °C/minute in heating step. |
| SVM-RFE | (1) Ratio C_{Tot} , O_{2Tot} , CH_{4Tot} throughout the steel casting. (2) Amount of graphite (kg) injected in Melting 2. (3) Amount of oxygen (m^3) injected in Melting 2. (4) Tons of scrap charged to the EAF, to the Melting 2. (5) °C/minute in heating step. |
| LDA_SVM-RFE | (1) Ratio C_{Tot} , O_{2Tot} , CH_{4Tot} throughout the steel casting. (2) Ratio O_2 , CH_4 in Melting 1. (3) Amount of CH_4 injected in Melting 1. (4) Amount of O_2 injected in Melting 1. (5) O_2 required to burn the CH_4 effectively. |

Although the performance of each algorithm was different, they share some common variables. With this information, the flow injection of the chemical energy in the EAF were adjusted so as to create conditions for effective combustion reactions, which allows the system to obtain thermal energy instead of losing by excess or lack of the elements reagents involved in the formation of chemical energy and chiefly in the formation of foamy slag.

5.1 Test Results

From previous information was determined the adjuste for the elements on the EAF system control. The adjustment was executed supported by the process melting team. This new flow allow chemical balance in the combustion reaction. That chemical energy helps to the electric energy supply by the transformer through the electrodes. The figure 6, depict the adjuste for the flows at the chemical elements.

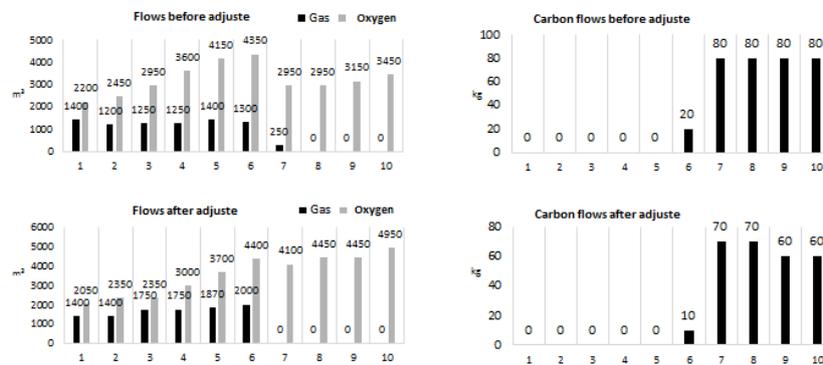


Fig. 6. Representative graphics for the adjusted on EAF system control

According with the graphics, we can see that the flows before the adjuste have irregular behavior since the phase 6, where the chemical energy is not constant. The flows adjusted claim supply the oxygen, natural gas and carbon in progressive form for generate the fusion after the heating of the scrap, this condition allow that the chemical energy stay agressive on the last steps. After the adjuste on the EAF control system were accused steel casting tests. The behavior for the kWh and time on for the EAF are presented in the figure7.

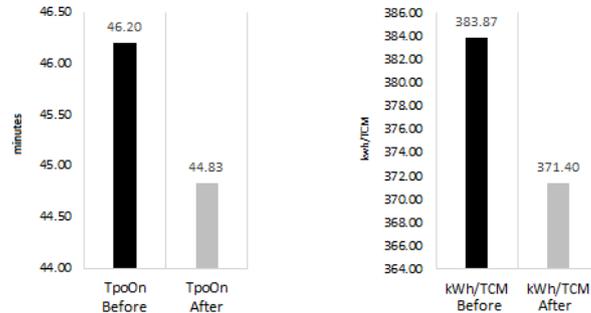


Fig. 7. Results obtained for the electric energy consumption (kWh/TCM) and the time connecting of the EAF(TpoOn)

The figure 7 depicts the results back to the settings made in the control system of EAF, where the predicted was decreased by at least 1% of electricity consumption and reduce time connected the EAF at least 1 minute. Achieving a savings of 3.5% in consumption and one minutes less on average than TpoOn.

6 Conclusions

In this paper, some algorithms for feature selection and the performance of each were presented. Similarly, a new it uses the eigen values generated by LDA and the SVM weight vector to guide the recursive feature elimination to obtain a subset containing the most discriminantes variables related to the energy consumption on an EAF. The LDA_SVM-RFE performance was better than the performance of the algorithms used for comparison. The results of test steel castings were completely satisfactory, achieving a 3.5% reduction in electricity consumption and 1 minute less for the EAF connected time in a steel casting.

References

1. Balakrishnama, S., Ganapathiraju, J.: Linear discriminant analysis for signal processing problems (2002)
2. Bernardo, P.R.d.C., Leonardo, C.d.A.: New trends of soft computing methods for industrial and biological processes (2010)
3. Chtioui, Y., Bertrand, D., Barba, D.: Feature selection method using genetic algorithm for the classification of small and high dimension data. In: First International Symposium on Information and Communications Technologies. pp. 1–4 (2004)
4. Chu, D., Zhang, X.: Sparse uncorrelated linear discriminant analysis (2013)
5. Claeskens, G., Croux, C., Kerckhoven, J.: An information criterion for variable selection in support vector machines pp. 541–558 (2008)
6. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. *Journal of Machine Learning Research* 3, 1157–1182 (2003)

7. Guyon, I., Weston, J., Barnhill, S.: Gene selection for cancer classification using support vector machines. *Machine Learning* 46, 389–422 (2002)
8. Khan Muhammad Badruddin, K., Yagi, I., Terano, T.: Applying data mining techniques to assess steel plant operation conditions. In: *Data Mining Found. and Intell. Paradigms*. pp. 343–361 (2011)
9. Kohavi, R., H. John, G.: Wrappers for feature subset selection. *Artificial Intelligence* 97, 273–324 (May 1996)
10. Kordos, M., Blachnik, M., Wiczorek, T.: Evolutionary optimization of regression model ensembles in steel-making process (2010)
11. Masaeli, M., Fung, G., G. Dy, J.: From transformation-based dimensionality reduction to feature selection (2010)
12. Mitchell, M.: *An Introduction to Genetic Algorithms*. Massachusetts Institute of Technology, London, England, 1st edn. (1996)
13. Sandberg, E.: *Energy and Scrap Optimisation of Electric Arc Furnaces by Statistical Analysis of Process Data*. Ph.D. thesis (2005)
14. Singh, S., Silakari, S.: Generalized discriminant analysis algorithm for feature reduction in cyber attack detection system pp. 173–180 (2014)
15. Storn, R., Price, K.: Differential evolution a simple and efficient heuristic for global optimization over continuous spaces. *Global Optimization* 11(4), 341–359 (1997)
16. Vapnik, V.N.: *The nature of statistical learning theory* (1995)
17. Xue, B., Fu, W., Zhang, M.: Differential evolution (de) for multi-objective feature selection in classification. In: *GECCO 14: Conference on Genetic and Evolutionary Computation*. pp. 83–84. ACM Press (2014)
18. Younes, C., Dominique, B., Dominique, B.: Feature selection by a genetic algorithm application to seed discrimination by artificial vision (1998)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Octubre de 2015
Printing 500 / Edición 500 ejemplares

