Advances in Artificial Intelligence

Research in Computing Science

Series Editorial Board

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 147, No. 4,** abril 2018. 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. 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 147**, **No. 4**, April 2018. Printing 500. The authors are responsible for the contents of their articles. Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 147(4)

Advances in Artificial Intelligence

María de Lourdes Martínez Villaseñor (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2018

ISSN: 1870-4069

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

Indexed in LATINDEX, DBLP and Periodica

Printing: 500

Printed in Mexico

ISSN 1870-4069

Editorial

Topics of interest in Artificial Intelligence are reflected in this volume of "Research in Computing Science". Applications and technique improvements in machine learning, evolutionary algorithms, fuzzy systems, multi-agent systems, wavelets, and Bayesian networks can be found in this volume. The applications domains are also diverse.

Machine learning techniques are used to identify songs as a cover version by Silva-Reyes et al. Pérez-Espinosa et al. developed an age and gender classifier for children using convolutional neural network. Musawenkosi et al. present a model to determine the propensity of a student to succeed.

Giusti et al. presented a multi-agent system applied to Virtual Learning Environments to assist distance education managers. Another application in learning domain can be found in Yannibelli, where a hybrid evolutionary algorithm to solve collaborative learning team formation problem in higher education is developed.

Bayesian Networks were used to consider contextualized preferences in a recommender system by González et al. Sánchez-Fleitas et al. presented a case based system to develop a model for the management of geospatial data.

A fuzzy speed controller was implemented in brushless DC motor drives by García-López et al, showing how the life time cycle of the power electronic converter is affected in a fuzzy speed controller. Siega described a method of automatic segmentation of the encephalon by MRI with relaxation time (T1), gradient echoes (GRE) and inversion recovery (IR).

Feature extraction improved methodology from images was presented in Barajas-García et al. They developed a wavelet local feature pattern recognition methodology in order to identify and locate key points that represent relevant information of an image. Moré et al. introduced a pure Multi-Objective Optimization approach with automatic contrast enhancement for color images.

The paper by Uribe et al. develops signal timing model for traffic intersection using agents technology. Jerónimo and Sossa describe a system for recognition of quasi-plane objects and discuss its specific implementation.

We are grateful to members of Mexican Society for Artificial Intelligence (SMIA) for their collaboration in the preparation of this volume.

María de Lourdes Martínez Villaseñor
Universidad Panamericana, Mexico
Guest Editor

March 2018

ISSN 1870-4069

Table of Contents

Page

ISSN 1870-4069

Cover Song Recognition Using Machine Learning Techniques

Andree Silva-Reyes, Fabiola Martínez-Licona, Alma Martínez Licona

Universidad Autónoma Metropolitana, Electrical Engineering Depto., Mexico

andree_sr@hotmail.com, {fmml, aaml}@xanum.uam.mx

Abstract. The task of recognizing a song as a cover version of another is relatively easy for the human being, when the song is known. However, making a machine to do this job is complex because of the number of variables involved in the development of a cover; these include variations in tempo, instrumentation, gender, and duration with respect to the original version. A methodology that aims to identify covers from the application and analysis of machine learning techniques, sparse codification, signal processing and second order statistics, in order to obtain the best configuration, is proposed. Acoustic features such as pitches and timbres, as well as beat information of the cover songs were obtained from the Million Song, a metadata database oriented to music information retrieval. Along the experimentation it was able to try different analysis configurations on the metadata and to appreciate the effects on the comparisons between original and cover versions. According to the results, a system that integrates a frequency processing on the pitches with beat alignment, a sparse codification and a clustering technique was obtained with correct cover identification similar to the state of the art results. It was also possible to get information about learning techniques combinations with different metrics that allows future experiments to improve the results.

Keywords: cover song recognition, clustering, sparse codification.

1 Introduction

The intersection among music, machine learning and signal processing has let to address a wide range of task such as automatic definition of melodies, chords and instruments, identification and characterization of long term times and structures or recognition of musical genres and covers [1]. There are organizations like the International Information Society for Recovery of Music Information (ISMIR), or the Music Information Retrieval Evaluation Exchange (MIREX), that have promoted the

use of these fields to the access, organization and understanding of musical information, focusing on the research and development of computational systems that aim to solve these series of tasks.

A musical version, or cover, is defined as a new interpretation, live or in studio, of a song previously recorded by another artist [2]. This implies that a musical cover may have shifts in the rhythms, tempo, instrumentation ranges, gender or duration with respect to the original version. As an example the song Summertime, originally performed by Abbie Mitchel in 1935, has up to 1200 musical covers to date according to the project Second Hand Songs[1]. Some of these are in general similar to the original song and some others are quite different; the Million Song Dataset[2] reports versions in the musical genres of jazz, rock-pop, rhythm & blues and even country. A Cover Identification System (CIS), is an automatic system that ideally determines if a song is a cover version of some musical piece located in a database. This problem has been addressed by applying several methods based in two stages: the first stage consists on the extraction and the analysis of the most important characteristics of the song such as its melodic representation, harmonic progression or pitch; the second stage aims to the measurement of the similarity degree between the features extracted from each piece of music.

Some previous work on the subject has aimed to solve these steps by proposing different methods. Lee showed an extraction method based on Hidden Markov Models applied to sequence of chords for each song and followed by a similarity degree measurement between chord sequences, using dynamic time warping [3]; the problem with this technique is that it needs a huge amount of time and computational resources. In [4] Jensen et.al calculated a Chroma-gram, a matrix from the Chroma vector sequences, that was not sensitive neither to instrumentation nor time changes, to obtain the minimum distance of the matrices using the Frobenius norm. Ravuri and Ellis [5], proposed to obtain the Chroma-gram and calculate three characteristics per song to classify by means of vector support machines (SVM), or multilayer perceptron (MLP), while Chuan proposed to calculate a Chroma-gram that saves the partial harmonics of the melody and maintain the volume invariance, to make a framework that measures similarity by means of a binary classifier [6].

In [7], a method inspired on the creation of digital fingerprints used to minimize the execution times in the search for covers is proposed by Bertin-Mahieux and Ellis. This research makes use for the first time of the database MSD (Million Song Dataset) which consists of characteristics and metadata for a million songs under the Creative Commons (CC) license; in [8] an introduction to MSD, as well as its creation process and possible uses are presented by the same authors plus Lamere. The search process in a database may be accelerated by dividing a song into small fragments that may be used as hashes; in [9] Grosche and Maller applied this technique, but with larger segments for each song in order to minimize el number of searches. In [10] Bertin-Mahieux and Ellis used the 2D Fourier Transform to procure a representation of the Chroma patches to obtain an efficient nearest neighbor algorithm; this scheme makes

[1] http://www.secondhandsongs.com/statistics

[2] https://labrosa.ee.columbia.edu/millionsong/

the nearest neighbors to have more probabilities of being related to the same song. A couple of modifications of this method are presented in [11] to improve the classification; the authors used data dispersion to enhance their separability followed by a dimensionality reduction.

Two methods that help making queries in a faster way on a large data base are proposed for cover identification: Basic Alignment Search Tool (BLAST), which is a bio-sequence indexation technique that Martin et. al used to increase the efficiency [12], and a data base prune method that Osmalskyj et. al used to reduce the set where the search is made [13]. The Locally-Sensitive Hashing (LSH), method was used by Khadkevich and Omologo in [14] to obtain similar chord songs and then to apply a progression method to refine the search ranking. On the other hand, Salamon et al focus on extracting tonal representations (melody, bass line and harmonic progression), by using state of the art algorithms, and a dynamic programming algorithm to measure the degree of similarity in [15]. From these works it is concluded that harmonic representations are more reliable for cover identification although the tonal representations better improves the recognition accuracy.

Van Balen et al use three descriptors to content-based music recovery: pitch bi-histogram, Chroma correlation coefficients and harmonization features [16]. Serrà explains the steps involved in a CIS: feature extraction, key invariance, tempo invariance, structure invariance and similarity calculation [17], while in [16], the authors group these steps into a two phases system: calculation of harmonic features and pitch for each song and comparison of the similarity.

This paper presents the results obtained from a CIS based on machine learning techniques. This is based on the state-of-the-art feature processing plus the introduction of a sparse codification in order to obtain a reduced size feature vector. The components to separate the main melody from the accompaniment were selected, the machine learning architecture for cover recognition was determined and the first test adjustments were carried out. Followed, a final test and the performance assessment were developed.

2 Methods

The CIS presented here was based on the work of Bertin-Mahieux and Ellis [10]. Although some other research is aimed to classify musical covers, these authors used the MSD database by applying a very precise methodology that allows obtaining a comparative framework with our results. In general, the tasks done in [10], are to get the Chroma features of MSD, to align the Chroma features with the beat, to apply the power law on the resulting matrix, to generate patches and to calculate the 2D Fourier transform, to calculate the median and finally to apply PCA. Figure 1, shows the methodology used here on each one of the songs.

In general, the characteristics of the songs were obtained from an adaptation of the Chroma and timbre data taken directly from the database. A spectral analysis was applied to the resulting matrices in order to get a 2-D Fourier Transform (2D-FFT), a Wavelet Transform (WT) and patterns generated from the algorithm named Sparse

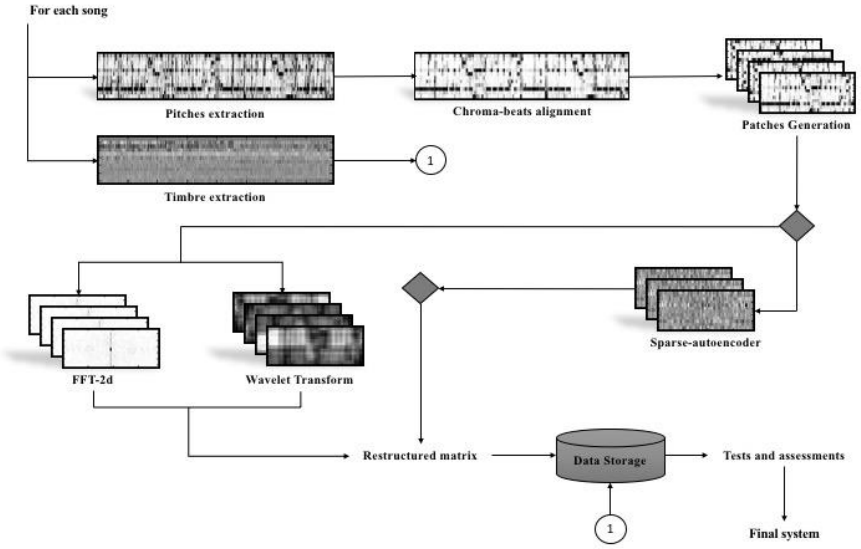


Fig. 1. Methodology applied to the individual song based on [10].

Autoencoder. WT and Sparse Autoencoder were added to the techniques showed in [10]. Cover recognition was made from metrics used to assess the ensemble clusters of each song.

2.1 Database

The Million Song Dataset was used to extract the songs and their covers. From this dataset it was extracted a subset named Second Hand Songs (SHS), that is oriented to original and cover songs[3] and is divided into training and test subsets [18].

2.2 Feature Extraction

Each musical piece included in MSD consists of a set of 55 attributes including song descriptors such as artist name, song title, labels with similar artists, and acoustic descriptors of the song such as pitches, timbre, volume and beat.

The first step was to determine the most representative attributes and the way to manipulate this information. This was done by selecting the most relevant features of the MSD songs listed on Table 1 through different classification methods including, Naive Bayes, Multilayer Perceptron and Support Vector Machine. Initially all attributes in the database were used, and they were subsequently removed one by one

[3] http://labrosa.ee.columbia.edu/millionsong/secondhand

in each new experiment in order to determine those that impacted the most on the correct classification of the songs. As a result, and in consistency with [7] and [19], pitches and timbre attributes were selected as features for the CIS development.

The pitches are the consequence of a perceptive property that allows ordering the sounds on a frequency-related scale [20]. In MSD pitch information is given by a set of Chroma vectors of length 12 that represents the notes with their semitones according to the Chromatic scale: C, C #, D, D #, E, F, F #, G, G #, A, A #, B. Each note is assigned a normalized numerical value, [0 1], which describes the relative dominance of each pitch on the time segment previously defined and named "musical event".

Timbre is the quality of a musical note or sound that distinguishes different types of musical instruments or voices [21]. In MSD timbres are derived from the spectral-temporal surface of a segment and it is represented by a vector of length 12, where the four first coefficients are related to the average segment volume, sound brightness, sound flatness and sound strength, while the remaining have no concrete meaning so they were left out of the final characteristic vector.

2.3 Chroma-Beats Alignment

Beat is a basic unit for time measurement in music. Time variability over the different cover songs are controlled when they are aligned with the beat, so this alignment was

Table 1. Song titles used to determine the set of attributes for CIS.

Song title	Number of Covers
Summertime	47
Silent Night	43
White Christmas	37
Body and Soul	19

applied on the pitches. The Chroma-beat vector was made taking the weighted average of the Chroma vectors included on the time segments of each beat. In order to increase the contrast in the Chroma values, the Chroma-beat vector was raised to 1.96 power according to [10].

2.4 Patches Generation

All the Chroma vectors belonging to the same song conform the Chroma matrix of this song. This matrix is segmented into a set of patches that were formed by taking 75 Chroma vectors and shifting by one Chroma vector, in order to obtain a number of [12x75], patches which value depends on the duration of the song. Next, the 2D-FFT and the WT as well as the Sparse Autoencoder, were applied to the patches data to get the song descriptors that will be used in the comparison among the covers.

2.5 Sparse Autoencoder

As described in [22], an autoencoder is a neural network that uses an unsupervised learning algorithm based on backpropagation in order to obtain the output values that most closely resemble the input ones, thus attempting to reproduce their own inputs. The architecture is similar to these of MLP, having input, output and hidden layers but the output layer has the same number of nodes as the input layer, to try to reproduce the values that enter the network, and the number of nodes of the hidden layer is less than the nodes of the input layer so the network is forced to turn the data representation into a smaller version. This feature makes Sparse Autoencoder (SA), to be used for dimensionality reduction purposes, so it allowed managing the amount of pitches data of each cover song, something "not covered" in [10].

The SA algorithm was applied on each song and on the set of songs with the same title in order to get one pattern that represents the individual song and other for the ensemble title. The set of patches for the song was restructured into a [900xN] array, where the [12x75], patches were converted to a [900x1], vector and N was the number of patches of the song. SA was applied to this array to achieve a [900xM], matrix, where M was the number of patterns calculated from the algorithm; after different testing configurations this value was set to 30, so the [900x30], matrix was restructured into a 30 [12x75], patterns that, according to the SA theory, represent the characteristics of the musical piece. The same procedure was applied to the ensemble of songs, so it was obtained a set of 30 [12x75], patterns per title.

2.6 2-D FFT and Wavelet Transforms

The set of patches obtained from each song were taken and transformed one by one by the 2-D FFT. Next, these Fourier transformed patches were integrated into an array of [Nx900], where each row represents a restructured patch from [12x75] to a [900x1] vector, and N continues to be the number of patches of the song. The same procedure was performed for the ensemble of songs with the same title. The set of 30 [12x75], patterns obtained from the application of the SA on the data was transformed with the 2-D FFT and restructured to a [30x900], M matrix

Wavelet Transform (WT), allows a signal to be represented as small time-frequency scale components and is suitable for the kind of signals that presents variations or abrupt discontinuities [23].

WT was applied, in addition to the 2-D FFT, to both the patches and the SA patterns of each cover song and the ensembles following the same procedure as in 2D-FFT. The wavelets used were Morlet, Sinc, Gauss, Meyer and Daubechies.

2.7 Cover Song Recognition Experiments

Different configurations were tested in order to find out the best approximation to a cover song identification system. The following experiments were carried out for this purpose:

EI. Cover song identification by comparison with a general ensemble bounded by a threshold. This experiment was divided in two stages. In the first stage a general descriptive ensemble of songs with the same title was obtained and the maximum value was calculated to use it as the threshold for the comparison of songs. The second stage consisted of making comparisons to determine whether a piece of music was a cover song or not. It was used the aligned array and the patch matrix, working with them separately. The general ensemble was obtained from the set of songs with the same title. The statistical measurements of mean, median and standard deviation (SD), were calculated from the set of patches and the aligned arrays of each song belonging the same group, and the descriptive ensemble was formed by the average of these statistical measures. The threshold was calculated by measuring the distances (subtraction, Mahalanobis distance and Euclidean distance), between each song and the general ensemble. The tests allowed to determine the degree of similarity between the general ensemble and other cover songs with the same title that were not included in its creation.

EII. Cover song identification by comparison with a general pattern bounded by a threshold. The same process as used in the previous section was applied to the set of SA patterns of each song. A stage of obtaining the general patterns of the group of songs with the same title was carried out with the statistical measurements of mean, median and standard deviation (SD), and the local entropy; next the average of these statistical measurements were computed. The distance measurements of subtraction, Mahalanobis and Euclidean, were used to obtain the threshold distance between each song patterns and the general pattern of the song.

EIII. Cover song identification using supervised (MLP) and supervised (K-nn) machine learning techniques and a representation of the data using 2D-FFT and WT. In pattern recognition k-nn is a non-parametric method used for regression and classification, based on measures of similarity or distance functions. The median of the [900xN] 2D-FFT matrix patches from the reference song and the other songs to compare, was calculated resulting in a [900x1], vector per song. Then, the k-nn algorithm was applied using the Euclidian, SEuclidian, City-block, Minkowski, Chebyshev, Cosine, Correlation and Spearman distance measurements. The same steps were repeated on the 2D-FFT of the patterns, the TW of the patches and the TW of the patterns. Comparisons were made in order to determine which characteristics allowed a better clustering of the data leading to the configuration that allowed to have a cluster of cover songs.

The procedure for the application of MLP was quite similar. The median of the matrices 2D-FFT and WT of both the patches and the patterns of the reference and other songs was calculated and then different configurations of MLP neural networks were tested in order to find the best cover song classification scheme.

EIV. Cover song identification using self-organized maps (SOM) on timbre features. SOM is a computational method for visualization and analysis of high-dimensional data [24]. They automatically organize the data so similar entries are mapped by being close to each other. SOM were applied on the first four coefficients of the timbre matrix directly obtained from MSD. The objective was to observe the way

Table 2. Best results for % of correct non-cover song identification by comparison with a general ensemble bounded by a threshold (EI) tests.

Subtraction (%)	Mahalanobis (%)	Euclidean (%)	Data Used	Statistical Meas
6.2	0.3	1.5	Aligned matrix	Mean
6.2	0.2	1.4	Patches	Mean
5.7	0	2.5	Patches	Median

Table 3. Best results for % of correct cover song identification by comparison with a general ensemble bounded by a threshold (EI) tests.

Subtraction (%)	Mahalanobis (%)	Euclidean (%)	Data used	Statistical Meas
56.8	29.1	42.9	Aligned matrix	Mean
54.8	25.6	45.8	Aligned matrix	Median
53.4	27.5	42.4	Patches	Mean

in which timbre coefficients were grouped in the case of having songs of the same title, and to identify if there were differences between cover and non-cover songs.

3 Results

The data set for the analysis and comparison was made from those SHS examples with at least 15 cover songs; the total number of cover songs was 417, corresponding to 19 different titles. As for the comparison tests, it was determined that each one was formed by five songs and an identifier of the test group: two covers of the same song and three other different songs. All songs were randomly selected, having a total of 1000 tests, which were used to assess the precision of the algorithms of each experiment.

The best results achieved in the cover song identification by comparison with a general ensemble bounded by a threshold experiment (EI), are shown in Table 2, for the non-cover song tests and in Table 3 for the cover song tests.

The best results obtained in cover song identification by comparison with a general pattern bounded by a threshold (EII), are shown in Table 4 for the non-cover song tests and in Table 5 for the cover song tests.

The Cover song identification using the supervised (K-nn), machine learning techniques and a representation of the data using 2D-FFT and WT (EIII), results are shown in Table 6. The best EIII correct results using the supervised MLP are shown in Table 7.

To perform the cover song identification using self-organized maps (SOM), on timbre features (EIV), the algorithm was trained with a cover and a non-cover songs of the group and the test was made with the remaining cover and non-cover songs. The overall result of correct assignation was 48.4% on the 1000 tests.

Table 4. Best results for % of correct non-cover song identification by comparison with a general pattern bounded by a threshold (EII) tests.

Cosine Distance (%)	Cityblock Distance(%)	Euclidean Distance(%)	Data used
0	2.4	2.4	Entropy SD
1	0.7	0.5	Pattern SD
0.1	0.3	0.1	Pattern median

Table 5. Best results for % of correct cover song identification by comparison with a general pattern bounded by a threshold (EII) tests.

Cosine Distance (%)	Cityblock Distance(%)	Euclidean Distance(%)	Data used
36.1	57.8	57.4	Entropy SD
31.7	34.8	34.6	Entropy median
31.9	34.6	34.9	Entropy median

Table 6. Results for cover song identification (%) tests using K-nn, (EIII).

Euclidean	Minkowski	Cosine	Correlation	Matrix	Wavelet
61.5	61.5	64	63.3	2D-FFT pattern	
60.2	60.2	63	56.5	2D-FFT patches	
42.4	42.4	40	32.9	WT pattern	Sinc
44.1	44.1	40.8	31.6	WT pattern	Gauss
45.3	45.3	40.5	34.6	WT patches	Morlet

Table 7. Results for cover song identification tests using MLP, (EIII).

% Correct classification	Matrix
49.5	2D-FFT pattern
47.5	2D-FFT patches
36.6	WT patches
36.2	WT pattern

4 Discussion and Conclusions

The results show several issues concerning to the analysis tools and learning techniques, as next presented.

1. *Cover song identification by comparison with a general ensemble bounded by a threshold*. Results in Table 2, show that it was not possible to separate the set of non-cover songs from the rest of the songs since most of them were below the threshold of the ensemble. For cover songs the best result was achieved with the mean of the aligned matrix using subtraction as a distance metric as shown in Table 3.
2. *Cover song identification by comparison with a general pattern bounded by a threshold*. As in the previous case, the results in Table 4 show that the threshold was not able to distinguish the non-cover songs from the rest while Table 5 shows that the best results for cover songs was achieved with the standard deviation of the data entropy using CityBlock distance as a metric. The pattern obtained by applying the Sparse Autoencoder allowed to reduce the data dimension and also to have another kind of representation for the song that might be of interest.
3. *Cover song identification using supervised (MLP), supervised (K-nn) machine learning techniques and a representation of the data using 2D-FFT and WT*. Tables 6 and 7 show that the supervised K-nn performed better and Fourier transform showed more relevant data than WT. Music as a signal can present a lot of events that change in time and frequency and the wavelet transform seemed to be a suitable signal processing tool to show them. Nevertheless the WT processing in metadata, as those of the MSD, did not achieve good results so this transform would be more useful if it is applied on raw data.
4. *Cover song identification using self-organized maps (SOM) on timbre features*. Timbres did not provide an improvement on the results. On the other hand, the SOM maps revealed some difficulties for the identification of the cover song group, but also showed keys to be identified that might be used in other schemes in order to increase the results.

Some other relevant aspects to consider are related to the data. MSD database contains a lot of records and a musical content variety that let the algorithms to be tested in a huge amount of data.

This is convenient since music databases exponentially grow every time. Concerning the features used in the experiments, timbres and pitches were the most relevant in terms of impact on the cover classification as is shown on a previous binary classification tests using Naive Bayes.

According to the results the best configuration achieved is shown in Figure 2. Likewise, the best result obtained was 64%, of correct cover song identification, which is close to the reported results (66-82%) in [10], although they are not the same tests.

In order to verify the effectiveness of the presented methodology, the experimentation performed in [10], was reproduced; 500 binary queries were carried out under the following scheme: given a song A that served as a query and two songs B and C, determine whether B or C was the cover of A. The results obtained in the reference were 82%, of covers identified correctly without applying PCA and 82.2%, with PCA with 50 major components on their data. In order to make the comparison the same scheme was used with other list of 500 queries taken from the same database. The results obtained were 75%, of covers identified correctly without applying PCA

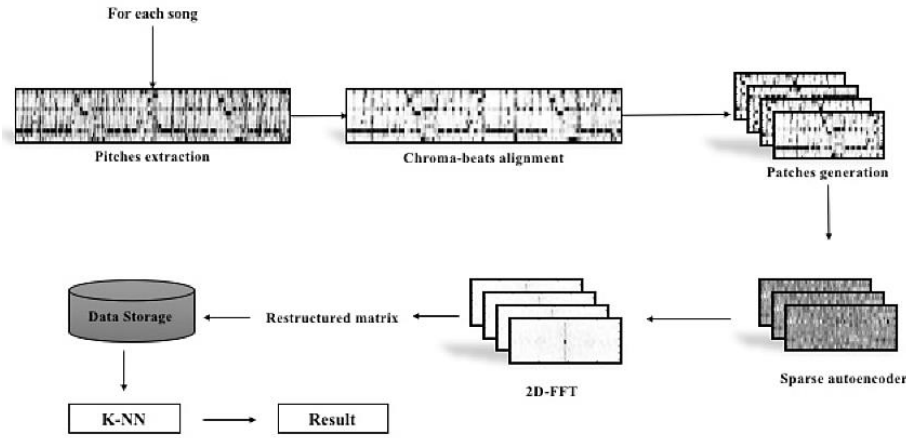


Fig. 2. Best configuration obtained of the CIS.

and 78.8% with PCA with 50 major components. By applying to the same data the CIS configuration shown in Figure 2, the best result was 73.8% (of correctly identified covers).

Further work is addressed on using raw audio signals for the analysis and the dealing with the unbalance issues that this task presents, the improvement on the parameter selection and preprocessing, and the experiment with other machine learning techniques such as Support Vector Machines or deep learning neural networks and apply inference systems like decision trees.

Acknowledgement. The first author would like to thank the Consejo Nacional de Ciencia y Tecnología (CONACYT), Mexico, for a grant given to him by them.

References

1. Humphrey, E., Bello, J. P., Le-Cun, Y.: Moving Beyond Feature Design: Deep Architectures and Automatic Learning. Music Informatics, Proceedings of 13th International Society for Music Information Retrieval Conference (ISMIR), pp. 403–408 (2012)
2. Oxford Dictionary: https://en.oxforddictionaries.com/definition/cover (2017)
3. Lee, K.: Identifying cover songs from audio using harmonic representation. MIREX task on Audio Cover Song Identification (2006)
4. Jensen, J. H., Christensen, M. G., Ellis, D. P. W., Jensen, S. H.: A Tempo-Insensitive Distance Measure for Cover Song Identification Based On Chroma Features. Proceedings of IEEE International Conference on Acoustics, (ICASSP), Speech and Signal Processing, pp. 2209–2212 (2008)

5. Ravuri, S., Ellis, D. P. W.: Cover Song Detection: From High Scores to General Classification. Proceedings of IEEE International Conference on Acoustics, (ICASSP), Speech and Signal Processing, pp. 65–68 (2010)
6. Chuan, X.: Cover song identification using an enhanced Chroma over a binary classifier based similarity measurement framework. International Conference on Systems and Informatics (ICSAI), pp. 2170–2176 (2012)
7. Bertin-Mahieux, T., Ellis, D. P. W.: Large-scale cover song recognition using hashed Chroma landmarks. Proceedings of IEEE Workshop on Applications of Signal Processing to Audio and Acoustics, (WASPAA), pp. 117–120 (2011)
8. Bertin-Mahieux, T., Ellis, D. P. W., Whitman, B., Lamere, P.: The million song dataset. Proceedings of the 12th International Society for Music Information Retrieval, Conference (ISMIR) (2011)
9. Grosche, P., Müller, M.: Toward characteristic audio shingles for efficient cross-version music retrieval. Proceedings of IEEE International Conference on Acoustics, Speech and Signal Processing, (ICASSP), pp. 473–476 (2012)
10. Bertin-Mahieux, T., Ellis, D. P. W.: Large-Scale Cover Song Recognition Using the 2D Fourier Transform Magnitude. Proceedings of the 13th International Society for Music Information Retrieval, Conference (ISMIR), pp. 241–246 (2012)
11. Humphrey, E., Nieto, O., Bello, J.: Data driven and discriminative projections for large-scale cover song identification. Proceedings of the 14th International Society for Music Information Retrieval, Conference (ISMIR), pp. 149–154 (2013)
12. Martin, B., Brown, D., Hanna, P., Ferraro, P.: BLAST for Audio Sequences Alignment: A Fast Scalable Cover Identification Tool. Proceedings of the 13th International Society for Music Information Retrieval, Conference (ISMIR), pp. 529–534 (2012)
13. Osmalskyj, J., Piérard, S., Van Droogenbroeck, M., Embrechts, J. J.: Efficient Database Pruning for Large-Scale Cover Song Recognition. Proceedings of IEEE International Conference on Acoustics, (ICASSP), Speech and Signal Processing, pp. 714–718 (2013)
14. Khadkevich, M., Omologo, M.: Large-scale cover song identification using chord profiles. Proceedings of the 14th International Society for Music Information Retrieval, Conference (ISMIR), pp. 233–238 (2013)
15. Salamon, J., Serrà, J., Gómez, E.: Tonal Representations for Music Retrieval: From Version Identification to Query-by-Humming. International Journal of Multimedia Information Retrieval, special issue on Hybrid Music Information Retrieval, 2(1), pp. 45–58 (2013)
16. Balen, J. V., Bountouridis, D., Wiering, F., Veltkamp, R. C.: Cognition-inspired Descriptors for Scalable Cover Song Retrieval. Proceedings of the 15th International Society for Music Information Retrieval, Conference (ISMIR), pp. 379–384 (2014)
17. Serrà, J., Gómez, E., Herrera, P.: Audio cover song identification and similarity: background, approaches, evaluation and beyond. Z. W. Ras, A. A. Wieczorkowska (eds.) Adv. In Music Information Retrieval, Studies in Computational Intelligence, 16, pp. 307–332 (2010)
18. Second Hand Songs dataset: http://labrosa.ee.columbia.edu/millionsong/secondhand (2017)
19. Patterson, R. D., Gaudrain, E., Walters, T. C.: The Perception of Family and Register in Musical Tones. Mari Riess Jones, Richard R. Fay, Arthur N. Popper. (eds.) Music Perception, pp. 37– 38 (2010)
20. Klapuri, A., Davy, M.: Signal processing methods for music transcription (2006)

21. Tristan, J., CSO, DesRoches, D.: Lead Audio Engineer. Analyzer Documentation, The Echo Nest Corporation (2016)
22. http://developer.echonest.com/docs/v4/_static/AnalyzeDocumentation.pdf (2014)
23. Andrew Ng: Sparse Autoencoder. CS294A Lecture notes, http://web.stanford.edu/class/cs294a/sparseAutoencoder.pdf (2017)
24. Meyer, Y., Bartram, J. F.: Wavelets and applications. The Journal of the Acoustical Society of America, 92(5), pp. 3023–3023 (1992)
25. Rojas, R.: Neural Networks (1996)

ISSN 1870-4069

Children Age and Gender Classification Based on Speech Using ConvNets

Humberto Pérez-Espinosa[1,2], Himer Avila-George[1,2], Juan Martínez-Miranda[1,2], Ismael Espinosa-Curiel[2], Josefina Rodriguez-Jacobo[2], Hector A. Cruz-Mendoza[2]

[1] CONACYT, Mexico

[2] CICESE-UT, Tepic, Nayarit, Mexico

{hperez, jmiranda, himerag}@cicese.mx

Abstract. In this paper, we present a study about building age and gender automatic classifiers for children at their first school years (between 6 and 11 years old). We created a speech corpus with 174 children interacting with a couple of robots in a Wizard of Oz scenario. The recorded speech was manually segmented and then characterized with low-level acoustic features. Next, we trained the classification models using a convolutional neural network architecture. Due to the complexity in the tuning process for the correct selection of the parameters used for this type of neural network, we integrated the use of a mathematical object called covering arrays to generate the set of optimal parameters for neural network architecture. Given the complexity of the classification of children speech, we obtained encouraging results. Our results indicate that it is difficult to achieve an accurate classification of children with very close ages. By grouping the subjects into two or three ages, the results improved significantly. On the other hand, the task of gender identification was less challenging, and we obtained higher classification performance measures.

Keywords: children speech, artificial neural networks, covering arrays.

1 Introduction

The automatic recognition of paralinguistic information can be useful to adapt better and personalize speech-based user interfaces. Paralinguistic phenomena can be extracted from the acoustic speech signal and then be used to identify, for instance, the speaker's identity, accent, gender, age, personality traits or emotional states. The automatic identification of this information facilitates the adaptation and personalisation of specific tasks in a human-computer interaction system or even in the assessment of the quality of the speech-based interaction between the user and the system [16].

The researchers have addressed the study of the correlation between paralinguistic information and individual's characteristics such as age and gender since the 1950s [15]. More recently, researchers have applied machine learning techniques in the construction of automatic classifiers which can recognize age and gender from adult's speech. An example of these works is [12], where the authors used Mel-frequency cepstral coefficients (MFCC) and delta regression coefficients to classify the age of Japanese speakers automatically. In [14], the authors compared different classification methods in a study of automatic classification of age group and gender. Moreover, in [13], the author refined and measured the significance of the long-term features to the age classification task.

The paralinguistic challenge organized at INTERSPEECH 2010, included two sub-challenges where the age and gender of speakers need to be identified using the "*aGender*" corpus containing 65,364 single utterances of 954 speakers with ages ranging from 7 to 80 years old [19]. For the age sub-challenge, four classes were defined: 7-14 (child), 15-24 (youth), 25-54 (adult) and (55-80) senior. The gender sub-challenge considered three categories: f (female), m (male) and x for children. This work found the discrimination of children's gender as notable difficult. [9], presented the best accuracy obtained for age classification. The authors used a fusion of different subsystems and employed Gaussian Mixture Models (GMM) and Support Vector Machines to model the tasks. In the gender sub-challenge, the best accuracy was obtained by a work that also used the fusion of different sub-systems and the classification models were constructed using SVM, Multi-Layer Perceptrons (MLP) and GMM [11]. A deeper review of the research efforts in the automatic classification of age and gender from speech, as well as the works and results obtained in the INTERSPEECH challenge, can be found in [20].

Much of the effort in the automatic recognition of age and gender using paralinguistic information has included children speech data as a single class to be differentiated from adults and elderly. Not many studies have been developed to find relevant information for age and gender classification between children at different ages. The development of speech-based interactive systems for children is equally important than those addressed for adults. Some applications of these methods include the study of child development [10], child education [23], or facilitators for autism therapy [8], to name a few.

The main contribution of this article is the classification of age and gender in children from their voices in a range from 6 to 11 years old, which is a challenging and little-explored task. Furthermore, to solve this task, we use a convolutional neural network which is a promising technique for the processing of audio signals.

2 Related Work

Only a small group of research initiatives have studied the acoustic features of children speech for the construction of automatic classifiers. One of these studies is the presented in [18], where the authors show the results of experiments in speaker recognition to identify a child in a class (30 children, similar age) and

the school (288 children, varying ages). Using the GMM and SVM approaches, the authors reported accuracy from 90%, for young children to 99%, for older children in the speaker recognition in one class, and 81% of identification rate achieved for a child in the school.

Regarding age identification in children from paralinguistic information, the work presented in [17], describes the experiments on age-group identification using the OGI Kids speech corpus [22], that contains recordings of words and sentences from 1,100 speakers. For this study, 766 speakers were chosen randomly for training and the remaining 334 for testing. The authors used GMM-UBM, GMM-SVM and i-vector systems for the construction of the classifiers. They used three age-group classes for age identification: 5-9, 9-13 and 13-16 years old. The reported results show that the GMM-UBM and i-vector systems considerably outperform the GMM-SVM system. The i-vector system applied to band-limited speech to 5.5 kHz obtained the best Age-ID performance 85.77%.

A similar study presents the work developed for the automatic recognition of gender in children using the same OGI Kids speech corpus [17]. In this experiment, the authors used GMM-UBM and GMM-SVM approaches for the identification of gender employing age-independent and age-dependent models. The authors obtained the best results when they used age-dependent models (GMM-SVM: 79.18%, GMM-UBM: 71.76%), in comparison with the age-independent models (GMM-SVM: 77.44%, GMM-UBM: 67.39%). Another relevant reported result is that the frequencies below 1.8 kHz and above 3.8 kHz are most useful for gender identification for older children, while the frequencies above 1.4 kHz are most useful for the youngest children.

In this paper, we report experiments focused on the identification of age and gender in children's speech. A main difference with the above-presented studies is that we focused on children attending to the elementary school (ages 6-11). Using a speech corpus from 174 children that we collected through an interactive scenario with a robot, our primary aim is to investigate to what extent is possible a relevant discrimination of age and gender in children aging in this age range.

3 Methodology

3.1 Speech Data Collection

Given our interest in the study of children, we decided to create a corpus of children's emotional speech. We collected children's speech during the interaction with two Lego robots. We induced different reactions in children depending on the robot's behavior. The activity that children performed consisted in giving verbal directions to the robots to go from one place to another, picking up candies and avoiding obstacles. We used two Lego Mindstorms EV3 robots, and we built a scenario in a 2x3 meters floor mat containing on its surface a route that the robots must follow, from the start to the finish position, see Figure **1**. The route has a curved trajectory comprising six depots represented as rectangles with three different colors.

There is a small bowl containing a pre-defined number of candies on each depot. The color of each depot represents the number of candies contained inside the bowl: blue depots contain bowls with two candies, orange depots have three candies in their bowls, and yellow depots hold a bowl with four candies.

Additionally, there are fourteen obstacles distributed along the route. The obstacles have the same colors as the depots to represent the number of candies lost if the robot knocks over the obstacle: blue obstacles represent the loss of two candies, orange obstacles for three candies and yellow obstacles represent four lost candies.

Fig. 1. From left to right: Child interacting with the robot, the technician operating the Text to Speech system, the technician controlling the robot's movements, the facilitator, and a friend of the child.

At the beginning of the interactive session, a facilitator explains to each child his/her mission: the primary goal is to guide the robot (using verbal commands), through the route, from the start position to the exit of the route. A second objective is to collect as many candies as possible from the bowls located in the depots. The robot needs to enter completely into the depot to collect the candies from the bowl. If the robot passes a depot without entering, the child loses the candies of that depot, given that the robot cannot go back. Additionally, the robot should avoid the obstacles located through the route. If the robot knocks over an obstacle, the candies collected so far are lost depending on the color of the obstacle.

A Wizard of Oz method has been used to generate the movements and the dialogues of the robots which are controlled by two technicians. One technician was responsible for simulating the movements of the robot using the mobile application Robot Commander App provided by Lego Mindstorms. The other technician was in charge of generating the robot's speech using a MacBook Air laptop. We used the Text To Speech engine of the OS X to synthesize the prompts spoken by the robots.

The robots used a vocabulary of 162 pre-defined sentences.

With the objective of generating speech with paralinguistic variations, the children interacted with two robots which are physically identical but have different personalities. One of the robots acts in a non-collaborative manner, ignoring some of the commands given by the child and playing some prompts to blame the child for the errors. This robot also shows a selfish behavior, keeping all the credit when it entered into stations. The other robot has a collaborative behavior and a happy mood. The utterances spoken by this robot encourage the child to get more candies. This robot is obedient and gives the child credit for his or her achievements. Children engaged in the activity and reacted emotionally to positive and negative events such as winning or losing candies, avoiding or knocking over obstacles, and being congratulated or reproached by the robot. They showed negative emotions like frustration during the session with the non-collaborative robot and showed positive attitudes and emotions like enthusiasm and joy during the interaction with the collaborative robot.

The children speech was collected using a wireless headset Logitech h600. This headset uses a USB receiver antenna with a range up to 10 m. We connected the antenna to a Dell computer with Windows 8.1 operating system that records the voice using the software Audacity v2.1.1.

A total of 174 children (78 female, 96 male), aged 6-11 (8.62 mean, 1.73 standard deviation), participated in the experiment. After finishing the recording of the interactive sessions, we manually segmented the children speech at speaker turn level. We define turn level as continuous segments in which the child is speaking uninterruptedly. It means that we take every intervention of the children (each instruction or answer to the questions made by the robot), as one speech segment. We use the software Audacity to analyze the audio recordings and cut out the parts of interest. After the segmentation, we obtained 18,674 segments. Table **1**, shows some details about the collected data.

Table 1. Minimum, maximum, average and standard deviation minutes that were used to accomplish the mission according to age of the participant.

Age	*Min*	*Max*	*Avg.Dur.*	*Std.Dev.*	*Participants*	*Female*	*Male*	*Segments*
6	05:51	22:46	13:59	03:36	32	20	12	3,739
7	08:09	20:05	13:53	03:03	21	8	13	2,714
8	07:45	21:47	12:21	02:47	23	9	14	2,473
9	08:52	17:31	11:58	02:55	30	12	18	3,111
10	05:54	19:03	11:00	02:27	41	19	22	3,697
11	06:39	16:42	10:59	02:02	27	12	15	2,940

3.2 Audio Characterization

We characterized the audio data acoustically using the software openSMILE [4]. We used the set of features proposed in [21], that is designed to reflect a broad coverage of paralinguistic information assessment.

This feature set includes the Low-Level Descriptors (LLD), listed in Table **2**. We computed these acoustic features using a frame size of 25 ms and a frame step of 10 ms. We applied a moving average filter for smoothing data contours.

We are using static acoustic features; this means that they do not model temporal phenomena, which might be discriminative for age and gender recognition[9]. For this purpose, we compute LLDs and then generate delta and double-delta regression coefficients. The result of this computation is three numbers per each of the speech sample frames. Delta and double-delta coefficients are calculated using the following equation:

$$d^t = \frac{\sum_{i=1}^{w} i * (x^{t+1} - x^{t-i})}{2\sum_{i=1}^{w} i^2,},$$

where w is the length of the frame and x^t is the signal data.

Then we calculated 39 statistical functions over the values of the LLD, its delta, and its double deltas coefficients in each frame. The 39 statistical functions that we use are: range, maximum position, minimum position, maximum mean distance, minimum mean distance, linear regression coefficient 1, linear regression coefficient 2, linear regression error A, linear regression error Q, quadratic regression, coefficient 1, quadratic regression coefficient 2, quadratic regression coefficient 3, quadratic regression error A, quadratic regression error Q, centroid, variance, standard deviation, skewness, kurtosis, quartile 1, quartile 2, quartile 3, inter-quartile range: quartile 2-quartile 1, inter-quartile range: quartile 3-quartile 2, inter-quartile range: quartile 3-quartile 1, percentile 95.0, percentile 98.0, zero crossing rate, number of peaks, mean peak distribution, peak mean, peak mean mean distribution, arithmetic mean of the contour, absolute mean, quadratic mean, nz absolute mean, nz quadratic mean, nz geometric mean, and nnz.

The result of this procedure is a set of 6,552 attributes for each single audio sample. After an experimentation stage comparing several feature selection methods, the method with the best results was *Relief Attribute*. We used this evaluation method as implemented in Weka [6]. The method showed the best accuracy rates when we took the 350 best-ranked attributes. We selected these features from the original feature set of 6,552 attributes to obtain the best descriptors and reduce the dimensionality of the attributes vector.

3.3 Convolutional Artificial Neural Networks

The convolutional artificial neural Networks (ConvNets), are biologically inspired by the Human Visual System "HVS" and its hierarchal architecture [7]. In these networks, the weights are shared across time or space. Neurons with the same weights are applied on input patches of the previous layer at different segments of the input data. In this way, it is achieved a degree of *translational invariance*, by computing feature maps [5], that allows the networks to learn patterns and reuse them in different space or time context.

Table 2. Set of acoustic features.

LLD	***Number of features***
F0 via autocorrelation function	117
F0 envelope	117
MFCC [0-12]	1,521
Spectral band energy (0-250 Hertz)	117
Spectral band energy (0-650 Hertz)	117
Spectral band energy (250-650 Hertz)	117
Spectral band energy (1000-4000 Hertz)	117
Mel-spectrum [0-25]	3,042
Spectral centroid	117
Spectral Flux	117
Spectral maximum position	117
Spectral minimum position	117
Spectral roll-off point (25%)	117
Spectral roll-off point (50%)	117
Spectral roll-off point (75%)	117
Spectral roll-off point (90%)	117
Logaritmic energy	117
Zero-crossing rate	117
Voicing probability	117
Total	6552

Therefore, ConvNets are useful when the inputs samples are statistical invariants, that is, the input samples contain the same kind of information, and it does not change on average across time or space. In a ConvNet, instead of having stacks of matrix multiply layers, there are stacks of convolutions. Currently, pattern recognition systems based on convolutional networks are among the best performing systems. This type of network architecture typically has five, six or seven layers [3]. ConvNets are structured by stacking up convolution layers. Usually, at the top of the structure there are fully connected layers, and at the end of the structure, there is a classifier.

Some critical parameters need to be configured in a ConvNet. For instance, *stride*, is the number of shifted features each time the filter moves. The strides between layers are used to reduce the dimensionality and to increase the depth of the neural network. Each layer in the structure is called a *feature map*. There could be more than one feature map, for instance, if an image has the channels R, G, and B, these three channels (K-size = 3), can be handled as individual features maps. In our particular case, we are handling only one channel because the audio recording is mono-channel.

When the shifting filter does not go beyond the edge of the feature vector, it is called valid padding. If the filter goes off the edge of the feature vector and therefore the output map size is the same size than the input map, it is called *same padding*. The pooling layers take all the convolutions in a neighborhood and combine them instead of skipping one in every two convolutions. Max pooling takes a small neighborhood around every point in the feature map and computes the maximum of all the responses around it. Given that the convolutions are done on lower stride, the structure becomes more expensive to calculate, and there are more parameters to tune. The average pooling, instead of taking the max, takes an average over the window of features around a particular location.

We implemented a typical architecture for a ConvNet.

It has two alternated layers of convolution and pooling, followed by a fully connected layer and a classification layer at the end. Fig. **2**, shows the network structure.

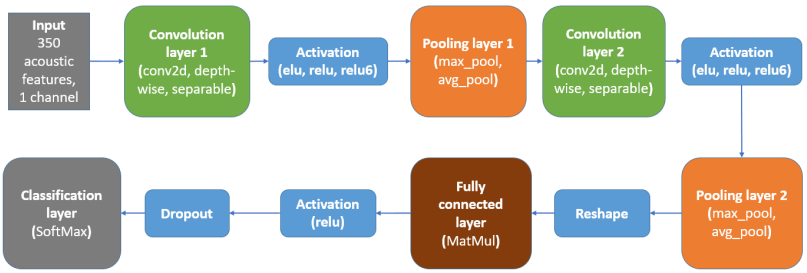


Fig. 2. Convolutional neural network architecture used for the experiments presented in this paper. The different options that we tested at each layer are shown in parentheses.

For the implementation of the convolutional neural network, we used the software TensorFlow [1]. An important characteristic of this tool is its flexible architecture which allows deploying computation to one or more CPUs or GPUs in a desktop, server, or mobile device using the same API. Google released TensorFlow under the Apache 2.0 open source license.

3.4 ConvNet Parametrization

Below we present the list of variables that needs to be tuned for the accurate training of the ConvNet and the corresponding values that we selected for testing. The $v\#$, stands for the identifier of the variable.

- **v1 - Num_hidden:** a positive integer that represents the depth of the convolution layers. It can have nine values, from 32 to 96 with steps of eight between values.
- **v2 - Num_epochs:** a positive integer that limits the number of steps that the validation set is evaluated. It can have seven values, from 10 to 70 with steps of 10 between values.
- **v3 - Learning_rate:** a float that represents the magnitude of the update per each training step. Decay once per epoch. It can have seven values, from 0.0007 to 0.003 with steps of 0.000383 between values.
- **v4 - Batch_size:** a positive integer that indicates the percentage of the training data that are feed at each iteration. It can have seven values, from 5 to 11 with steps of one between values.
- **v5 - Eval_batch_size:** a positive integer that indicates the percentage of the validation data that is feed at each iteration. It can have seven values, from 8 to 32 with steps of four between values.

v6 - **Patch_size** a positive integer that indicates the size of the window that slides across the feature vector. It could also be bidimensional, in our case it is unidimensional. It can have five values, from 3 to 7 with steps of one between values.

v7 - **Depth** a positive integer that represents the depth of the second convolution layer. It can have nine values, from 16 to 32 with steps of four between values.

v8 - **Strides** a list of integers that represents the number of features that are shifted by the sliding window for each dimension of the input vector each time the filter moves. It can have three values, from 1 to 3 with steps of one between values.

v9 - **K-size** an integer that represents the size of the dimension of the input. It can have three values, from 1 to 3 with steps of one between values.

v10 - **Partition** an integer that represents the distribution of the data for the train, validation, and test sets. It can have three values, from 1 to 3 with steps of one between values. The tested partition were 1 = 33/33/33, 2 = 40/40/20 and 3 = 70/20/10.

v11 - **Padding** If the filter goes off the edge of the feature map or not. The two possible values for this parameter are same and valid. It can have two values, from 1 to 2 with steps of one between values.

v12 - **Optimizer:** functions to compute gradients for a loss measure and apply gradients to variables. It can have two values, from 1 to 2 with steps of one between values. The possible values of this parameter are 1 = *GradientDescentOptimizer*, 2= *AdamOptimizer*.

v13 - **Activation:** functions that provide nonlinearities. It can have two values, from 1 to 2 with steps of one between values. 1 = continuous but not everywhere differentiable functions (*relu6*) 2= smooth nonlinearities functions (*elu*).

The methodology for setting the values of the parameters of an ANN is based on the study of the effect on the quality of the solution, which is caused by the interaction between the variables of the experiment.

The tuning process of the parameters of the ANN was done using a mixed-level covering array denoted by $MCA(N; t, k, (v_1, \ldots, v_k))$. An MCA is a $N \times k$ matrix in which the entries of the ith column arise from an alphabet of size v_i; additionally, each column $i(1 \leq i \leq k)$ contains only elements from a set S_i with $|S_i| = v_i$, and the rows of each $N \times t$ subarray cover all t-tuples of values from the t columns at least once.

We use a $MCA(64; 2, 9^1 7^4 5^2 3^3 2^3)$ to tune the parameters of the ANN, this MCA represents the experimental plan. The MCA was constructed using the simulated annealing algorithm reported in [2]. The experimental plan is composed of sixty-four experiments for each of the ages groupings tested.

4 Results

For the experiment implementation, we used the TensorFlow framework installed on a Linux server equipped with 72 Intel Xeon processors, 64GB RAM, and a Tesla K20 GPU accelerator. The Linux distribution for this server is CentOS.

The Table **3**, shows the results of the classification experiments including precision, recall, and F1-score. These metrics reach its best value at 1; this is, the closest to 1, the most accurate is the classification. To measure the classification performance we used three data subsets: training, validation, and testing. The results showed in the tables are the ones obtained with the test set. The speech segments of each child belong only to one of the three data subsets to ensure the training of speaker-independent models.

As we can see in table **3**, we obtained the best results when we classified two classes of age and two classes of gender. For age and gender classification we got comparable results to the ones reported in the literature. However, our results do not surpass the obtained by [17], using the OGI Kids speech corpus. It is important to say that we are working with a narrower age group where it is expected to have less variability in the acoustic properties of the children speech. Another important difference is that our data were recorded in a noisier environment. We also built classification models for three groups of ages and all the six included ages without grouping. As reported before [19], we found these task tough, obtaining a poor classification performance.

The Table **4**, shows the best combinations of parameters for each of the classification tasks. We can see that from the 64 tested combinations, experiment 60 had the best performance for two tasks (2 age classes, and two gender classes). On the other side, the combinations 30 and 7 had the worst performance of the two task each one. From these results, we can have a better idea about what are the most recommendable parameters for the ConvNet for these tasks. We can see for example that to obtain a good classification the parameter v8 (strides), should not be set to 1, or that the ConvNet has a better performance when parameter v13 (activation), is set to 2.

5 Conclusions

Children's age classification is a challenging task; we were able to obtain a good classification performance grouping the data in two classes. However, we got poor results when we tried to classify the children per years old. We consider that the results obtained classifying by gender were good in comparison with results previously reported by other authors. We found that covering arrays are a useful tool to parameterize a ConvNet, given that we were able to select an adequate combination of parameters for our classification tasks. The obtained results encourage us to keep researching on the combination of deep neural networks structures and covering arrays to parameterize and solve classification tasks related to paralinguistic phenomena.

Table 3. Gender and age classification performance for different ranges. 2 - Classes (6,7,8 vs 9,10,11), 3 - Classes (6,7 vs 8,9 vs 10,11), 6 - Classes (6 vs 7 vs 8 vs 9 vs 10 vs 11). The table shows the maximum, minimum, mean and standard deviation of the evaluation metrics for the 64 experiments' results.

Number of classes	*Max*	*Min*	*Mean*	*StdDev*
	F1-score			
2 Classes Age	**0.71**	0.36	0.56	0.11
3 Classes Age	**0.50**	0.18	0.38	0.11
6 Classes Age	**0.34**	0.04	0.16	0.09
2 Classes Gender	**0.71**	0.37	0.52	0.10
	Precision			
2 Classes Age	**0.72**	0.27	0.58	0.10
3 Classes Age	**0.50**	0.12	0.38	0.11
6 Classes Age	**0.33**	0.04	0.17	0.10
2 Classes Gender	**0.71**	0.28	0.54	0.10
	Recall			
2 Classes Age	**0.72**	0.49	0.59	0.07
3 Classes Age	**0.53**	0.31	0.43	0.07
6 Classes Age	**0.35**	0.12	0.22	0.09
2 Classes Gender	**0.71**	0.28	0.54	0.10

Table 4. Best and worst experiments parametrization for ConvNet

Classes	*Exp*	*v1*	*v2*	*v3*	*v4*	*v5*	*v6*	*v7*	*v8*	*v9*	*v10*	*v11*	*v12*	*v13*
					Best result									
2 Classes age	60	48	30	0.0022	11	24	6	32	3	2	3	1	2	2
3 Classes age	41	72	30	0.0019	9	8	5	28	2	2	2	2	2	2
6 Classes age	37	48	70	0.0026	7	28	5	24	3	3	2	1	2	2
2 Classes gender	60	48	30	0.0022	11	24	6	32	3	2	3	1	2	2
					Worst result									
2 Classes age	30	40	50	0.0026	6	28	7	20	1	1	1	1	2	1
3 Classes age	30	40	50	0.0026	6	28	7	20	1	1	1	1	2	1
6 Classes age	7	64	60	0.0030	8	12	4	28	1	2	2	1	2	1
2 Classes gender	7	64	60	0.0030	8	12	4	28	1	2	2	1	2	1

Acknowledgements. This research work has been carried out in the context of the "Cátedras CONACyT" program funded by the Mexican National Research Council (CONACyT). This work was financed by CONACYT under the Thematic Networks of Research program (Thematic Network on Language Technologies Ref. 260178, 271622).

References

1. Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., Corrado, G., Davis, A., Dean, J., Devin, M., et al: TensorFlow: Large-scale machine learning on heterogeneous systems. Software available from tensorflow.org 1 (2015)
2. Avila-George, H., Torres-Jimenez, J., Gonzalez-Hernandez, L., Hernández, V.: Metaheuristic approach for constructing functional test-suites. IET Software 7(2), 104–117 (2013)
3. Bengio, Y.: Learning deep architectures for AI. Foundations and trends® in Machine Learning 2((1)), 1–127 (2009)

4. Eyben, F., Wöllmer, M., Schuller, B.: Opensmile: the munich versatile and fast open-source audio feature extractor. In: Proceedings of the international conference on Multimedia. pp. 1459–1462. ACM (2010)
5. Gens, R., Domingos, P.: Deep symmetry networks. In: Advances in neural information processing systems. pp. 2537–2545 (2014)
6. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.: The WEKA data mining software: an update. SIGKDD explorations newsletter 11((1)), 10–18 (2009)
7. Hassan, M., Khalifa, O., Talib, A., Abdulla, A.: Unconstrained facial recognition systems: a review. Asian Journal of Applied Sciences 3((02)) (2015)
8. Ketterl, M., Knipping, L., Ludwig, N., Mertens, R., Rahman, M., Ferdous, S., Ishtiaque Ahmed, S., Anwar, A.: Speech development of autistic children by interactive computer games. Interactive Technology and Smart Education 8(4), 208–223 (2011)
9. Kockmann, M., Burget, L., Černocký, J.: Brno university of technology system for interspeech 2010 paralinguistic challenge. In: Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010). vol. 2010, pp. 2822–2825. International Speech Communication Association (2010)
10. Laplante, J., Michaud, F., Larouche, H., Duquette, A., Caron, S., Letourneau, D., Masson, P.: Autonomous spherical mobile robotic to study child development. IEEE Trans. on Systems, Man, and Cybernetics (2005)
11. Meinedo, H., Trancoso, I.: Age and gender classification using fusion of acoustic and prosodic features. In: INTERSPEECH. pp. 2818–2821. Citeseer (2010)
12. Minematsu, N., Sekiguchi, M., Hirose, K.: Automatic estimation of one's age with his/her speech based upon acoustic modeling techniques of speakers. In: Acoustics, Speech, and Signal Processing (ICASSP), IEEE International Conference on. vol. 1, pp. I–137. IEEE (2002)
13. Muller, C.: Automatic recognition of speakers' age and gender on the basis of empirical studies. In: Proceedings of the Interspeech,Pittsburgh, Pennsylvania. pp. 1–4 (2006)
14. Muller, C., Wittig, F., Baus, J.: Exploiting speech for recognizing elderly users to respond to their special needs. In: Proceedings of the Eurospeech, Geneva, Switzerland. pp. 1305–1308 (2003)
15. Mysak, E.: Pitch and duration characteristics of older males. Journal of Speech & Hearing Research (1959)
16. Pérez-Espinosa, H., Martínez-Miranda, J., Espinosa-Curiel, I., Rodríguez-Jacobo, J., Avila-George, H.: Using acoustic paralinguistic information to assess the interaction quality in speech-based systems for elderly users. International Journal of Human-Computer Studies 98, 1–13 (2017)
17. Safavi, S., Russell, M., Jancovic, P.: Identification of age-group from children's speech by computers and humans. In: INTERSPEECH. pp. 243–247 (2014)
18. Safavi, S., Najafian, M., Hanani, A., Russell, M.J., Jancovic, P., Carey, M.J.: Speaker recognition for children's speech. In: INTERSPEECH. pp. 1836–1839 (2012)
19. Schuller, B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Muller, C.A., Narayanan, S., et al: The interspeech 2010 paralinguistic challenge. In: InterSpeech. vol. 2010, pp. 2795–2798 (2010)
20. Schuller, B., Steidl, S., Batliner, A., Burkhardt, F., Devillers, L., Müller, C., Narayanan, S.: Paralinguistics in speech and language—state-of-the-art and the challenge. Computer Speech & Language 27(1), 4–39 (2013)

21. Schuller, B., Steidl, S., Batliner, A., et al: The interspeech 2009 emotion challenge. In: Interspeech. pp. 312–315 (2009)
22. Shobaki, K., Hosom, J., Cole, R.: The OGI kids' speech corpus and recognizers. pp. 564–567. ICSLP (2000)
23. Ward, W., Cole, R., Bolaños, D., Buchenroth-Martin, C., Svirsky, E., Vuuren, S.V., Weston, T., Zheng, J., Becker, L.: My science tutor: A conversational multimedia virtual tutor for elementary school science. ACM Transactions on Speech and Language Processing (TSLP) 7(4), 18 (2011)

ISSN 1870-4069

Modeling the Student Success or Failure in Engineering at VUT Using the Date Band Algorithm

Langa Hendrick Musawenkosi[1], Twala Bhekisipho[2]

[1] Vaal University of Technology, Department of Electrical Engineering, Vanderbijlpark, South Africa

[2] University of South Africa , Department of Electrical & Mining Engineering, Johannesburg, South Africa

hendrickl@vut.ac.za, twalab@unisa.ac.za

Abstract. The success or failure of students is a concern for every academic institution, college, university, governments and students themselves. This paper presents a model to determine the propensity of a student to succeed in the Electrical Engineering Department at Vaal University of Technology. Firstly, various machine learning algorithms which can be used in modelling and in predicting student success or failure are discussed as well a new algorithm called the date band algorithm. Secondly, the concept of an academic model is also discussed. This model defines the domain and focus of data used to make preditcions. The academic model consists of the subject, the lecturer and the sudent each of which has various attributes. One of the attributes discussed in this paper is the popularity index which is a measure of cohesiveness of the model.The Date Band Algorithm is presented among others in the development of the model. In this algorithm, predictions are made to optimize the performance of academic environment, thereby impacting on the choices of funders when they support students.

Keywords: academic environment model, date band algorithm, decision trees, k-nearest neighbor, machine learning.

1 Introduction

The success rate in academic environments is not only a concern for those institutions but also governments, sponsors such as the public and the private sectors, parents, students themselves and other stakeholders. It therefore behooves us to investigate the propensity of those students to succeed using scientific methods such as machine learning. Machine Learning (ML), has a variety of algorithms that can be applied in addressing this problem.

The South African government and funders can save a lot of resources when funding these institutions. Therefore, the application of rigorous methods of machine learning can improve the efficiency in the academic sector.

For the most part in South Africa, the largest contributor of funding in public education is government, that is, the ministry of education. Although the ministry of education takes no account of income that is raised from student fees and other private sources, these public institutions have to account by submitting annual financial statements which reflect all income and all expenditure from all public and private sources [8].

The need to attract and retain students in engineering programs remains by necessity a focal point of interest and effort in engineering education, [12]. All universities and colleges have marketing departments to make sure that they attract the best of the best. They run various marketing programs for this purpose. Paul and Cowe Falls, [3] highlighted the three aspects for engineering careers success based on the availability of the resources. Firstly, lifelong learning is fundamental for success in the 21st, century engineering career. Staying abreast with the most recent technological advancement is essential for being innovative and creative. Secondly a study in the engineering construction industry is the most critical aspect of fostering a successful career path was in developing a career network. This includes networking, mentorship training and constructive feedback. Thirdly, the aspect of engineering career success relates to the models "proactive personality" variable.

2 Machine Learning Algorithms Overview

2.1 Decision Trees, DTs

Decision Trees, DTs [12], are simple yet successful techniques for supervised classification learning. This classification method consists of decision nodes, connected by branches, extending from the root node until the terminating leaf nodes, [1]. Starting at the root node attributes are tested at the decision node, with each possible outcome resulting in a branch. A decision tree algorithm aims to recursively split the observations into mutually exclusive subgroups until there is not further split that makes a difference in terms of statistical or impurity measures, [4]. A path is traced from the root to a leaf node which holds the class predication for that sample. Decision trees can easily be converted into IF-THEN rules and used for decision making, [9].

2.2 The K-Nearest Neighbor Classifier, KNN

The K-Nearest Neighbor is an example of instance based learning, in which the data set is stored, so that the classification for a new unclassified record may be found by simply comparing it to the most similar records in the training set [5].

The KNN algorithm is the earliest researched algorithm used for classification and is proved as one of the algorithms which have good classification results, but there are still some problems that need to be attended to. For example, it is not yet settled, how to select the value of k and how to select feature sets to make the classification better and their impact on each other [11].

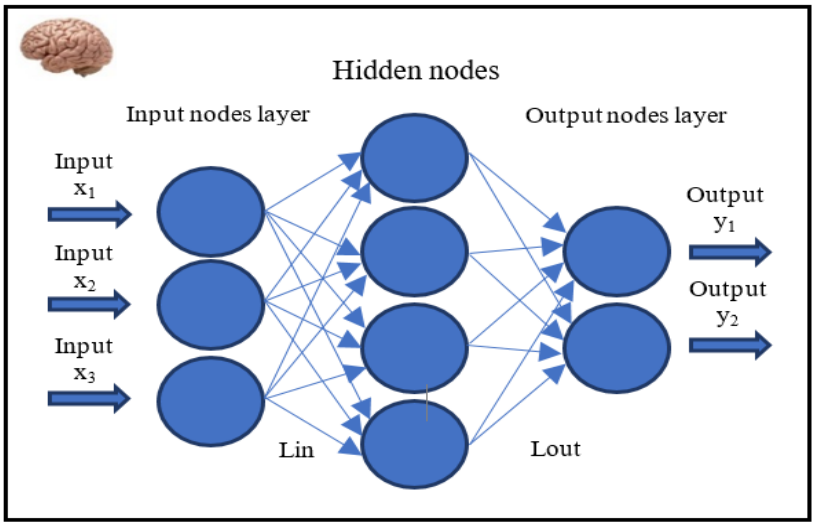


Fig. 1. Artificial Neural Network.

2.3 Support Vector Machine, SVM

Support Vector Machines (SVM), is an algorithm that uses nonlinear mapping to transform the original data into a higher dimension [7]. SVM's are pattern classifiers that can be expressed in the form of hyper planes to discriminate between positive instances and negative instances pioneered by Vapkin [10].

2.4 Artificial Neural Network, ANN

The artificial neural network (ANN), is an information processing paradigm that is inspired by the way biological nervous systems, like the brain process information [6]. ANN's are powerful tools that can be used to learn patterns from data.

According to [5], a neural network is a collection of nodes that are connected in some pattern to allow communication between the nodes. These nodes, also referred to as neurons or units, are simple processors whose computing ability is typically restricted to a rule for combining input signals and an activation rule that takes the combined input to calculate the output signal. Output signals may be sent to other nodes along connections known as weights. The weights usually excite or inhibit the signal that is being communicated.

An example of neural network is shown in Fig. 1. Since the artificial neuron mimics the natural neuron, neural networks therefore mimics the operation of the brain and can be quite useful in making classifications and predictions.

3 Modeling an Academic Environment

3.1 The Academic Environment Model

The environment is the sum total of surroundings of a living organism including natural forces and other things, which provide conditions for development and growth as well as danger and damage. An academic environment, in figure 2, is where a student exists. In order to model this, it is necessary to gather information about the student, the lecturer and the module which will form part of the environment.

Tinto's (1975), Student Integration Model (SIM), postulated that students who persist and succeed in college are those who are able to integrate successfully into an institution's social an academic environment. Alternatively, the students who are more likely to struggle and fail to persist are those who do not attempt or achieve social and academic integration.

3.2 The Lecturer's Popularity Index

One of the common problems in higher education is the evaluation of the instructor's performances in a course.

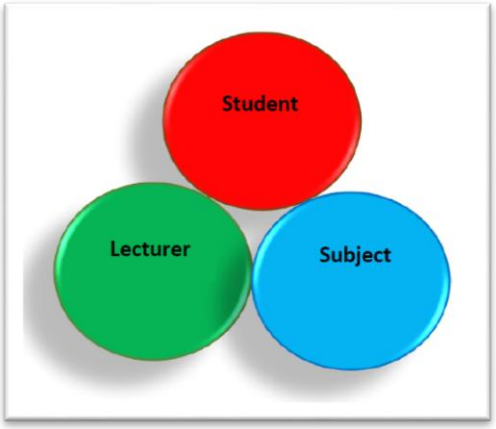


Fig. 2 Academic Environment Model.

The most widely used tool to evaluate instructors' performance in a course is through surveying students' responses about the course and its instructor through a questionnaire [2]. The percentage score of likes for a given the subject or the lecturer is given by the following equation:

$$L(x) = \frac{\sum_{i=1}^{n} x_i}{nk}, \tag{1}$$

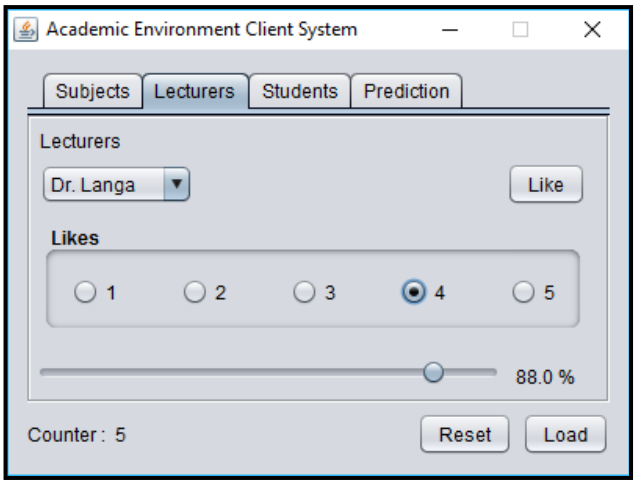


Fig. 3. Academic Environment Client System.

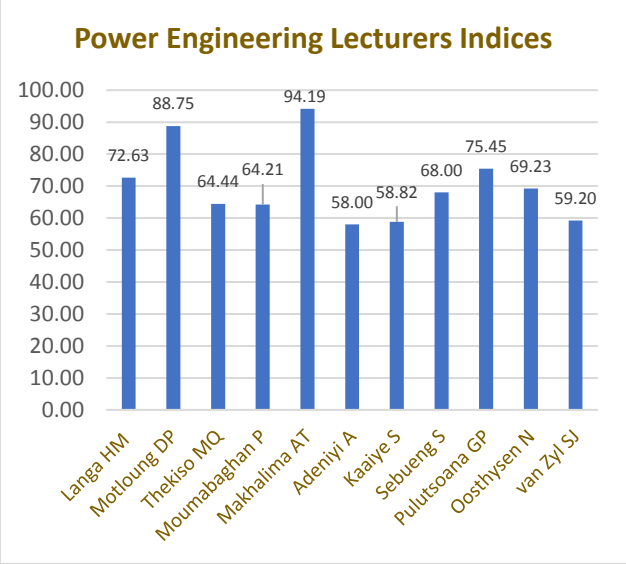


Fig. 4. Popularity Indices for Power Engineering Lecturers.

where:

n = number of instances of likes for a lecturer or subject, and

k = the highest number of likes in an instance.

Clearly, some lecturers are more popular than others. There are lecturers whom students really detest and there are lecturers whom they adore. These can be due to several reasons, such as the appearance, teaching style, level of education, leadership

and so on. The popularity of the lecturer has a correlation with the performance of the student. Similarly, the popularity of the subject can be measured.

3.3 The Academic Environment Client System

The panel below allows the student to rate the lecturer using choices of numbers between 1 and 5 as shown in Fig 3. If the lecturer is least popular then the choice would be a 1 and if the lecturer is a student's favorite then the choice would be a 5. This information is then captured in a database for future references. The popularity of the lecturer can increase or decrease with time depending on the performance of the lecturer. This is an important feature to have in the design.

A survey of approximately 600 students was completed where eleven lecturers were evaluated and eight subjects were also evaluated using the system as shown in the figure below. Upon analyzing results, interesting observations were noted. A lecturer teaches an average of 120 students per semester.

3.4 Popularity Indices for Power Engineering Lecturers

It is clear from the graph that some lecturers are more popular than others and that no lecturer obtained less than 50%, which is fair enough in terms of the quality of lecturers employed in the department of Power Engineering.

There are various reasons why a lecturer would be unpopular. It could just be shear laziness on his part, lack of understanding of the subject he teaches, the attitude, very strict, to name a few but a few. And there are various reasons why a lecturer could be popular. It could be that they are good in the subject matter, they have good qualifications, they are lenient, their attitude, again to name a few.

Evidently, as seen in Fig. 4, some lecturers are more popular than others in this department as would be the case with other departments.

4 Throughputs for 2016 Semester 1 & 2

There are many ML algorithms used for making predictions. This paper focuses on the algorithm called the date band algorithm with special reference to making predictions of student success or failure. Some algorithms will be more accurate than others and some will be more appropriate than others also. Evidently, as shown in Fig. 5, some subjects are more popular than others in this department as would be the case with other departments.

The x axis is the representation of subject codes for the department of Power Engineering. T Put% S1 and T Put% S1 are percentage throughputs for semester 1 and 2 respectively. An interesting observation in the graph is that there is a general improvement in the throughput for all subjects except for EPPRT3C and a slight decrease in EPDES3B.

While it is plausible to observe an increase in the throughput it is conversely concerning to see a decline in the throughput. This decline in the throughput, however,

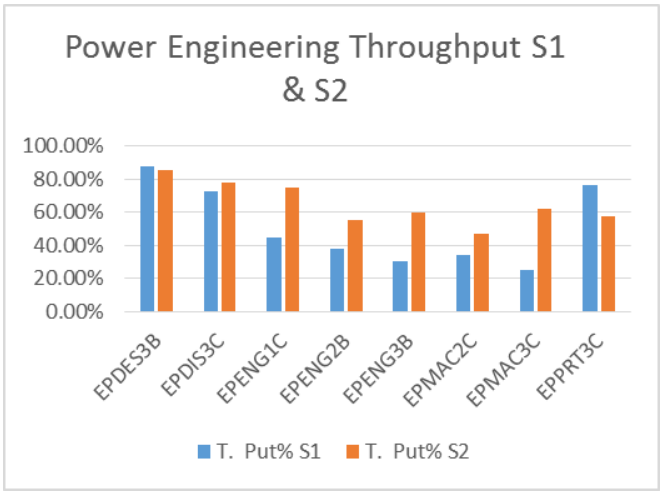


Fig. 5. Throughput for Semester 1 & 2 in 2016

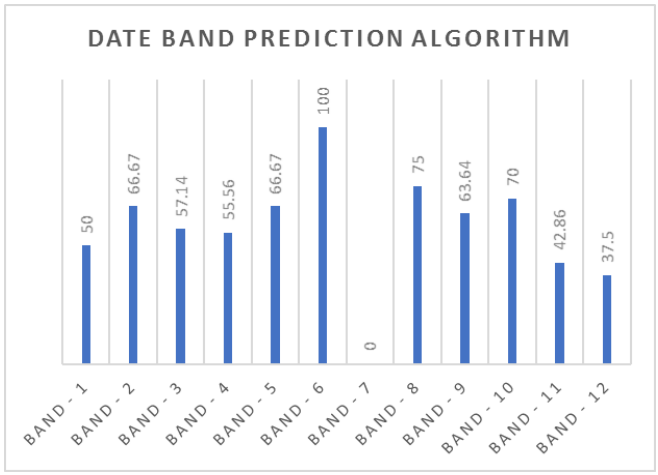


Fig. 6. Date Band AlgorithmProbabilities

can be addressed with the lecturers who are responsible for teaching those subjects. On the other hand, for those subjects that have seen improvement, either maintain the position or strive for more improvement.

5 The Date Band Algorithm

The date band algorithm, shown in Algorithm 1.1, is derived from an assumption that there exists a relationship between the date of birth of the student attribute and the propensity of that student to pass or fail.

Algorithm 1.1 Date Band Algorithm

```
Start
{
int Count = 0, Passed = 0, n = 0;
Enter date of birth
Read date of birth,
for ( n = 1, to number of entries, n++)
{
Scan the list of students for students within this date band
If (Found) then
{
Count++;
Check if Successful
If (Successful) then
{
Passed++
}
End if
}
End if
}
End for
Predict Probability for Success
}
End Start
```

The algorithm scans the date band to determine the number of students that succeed in that band and those that fail in that band. To predict the probability of the student instance to succeed, it is a question of classifying the student according to that probability.

So, this algorithm, as it has been stated, works on the premise that students can first be classified in date bands as far as their date of birth are concerned and then a prediction of the probability of that student to succeed or fail in the academic environment can be made. The following graph reveals some interesting results.

It is clear from this bar graph, in Fig. 6, that Date band – 6 scores the highest, and this means that the students in this band have 100 % probability that they will succeed in when enrolled for this subject. The lowest in this test is Band – 12 with 37.5 % probability. Band – 7 had a 0 % probability, but the reason it is zero is that there were no students in that band in fact.

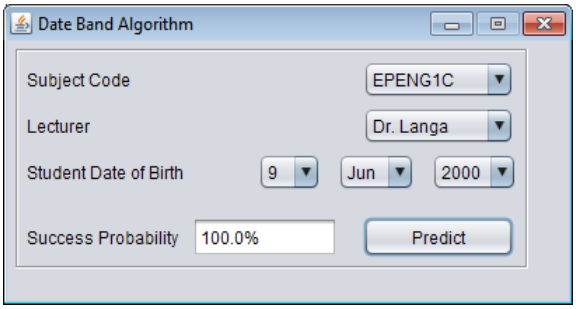


Fig. 7. Date Band Algorithm GUI.

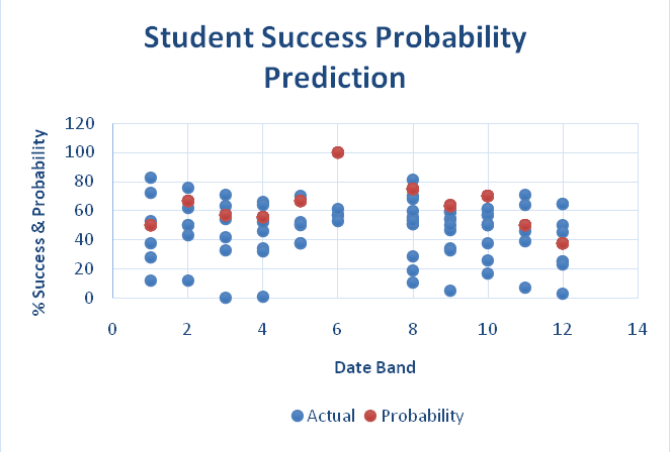


Fig. 8. Student Success Probalitiy Prediction Model Rerults.

6 Results of the Date Band Algorithm

The academic environment consists of the student, the lecturer and the course or subject that the student enrolls for. It has been shown that the condition of the environment can be measured by making use of the popularity indices of lecturers and the courses themselves. So, when a student enters an environment, he or she will be subjected to these conditions. On the other hand, the student has certain features which play a part in determining his or her success. These features can include the APS score. The target variable could pass or fail, which implies that we want to predict whether the student will pass or not given certain set of variables such as:

X_1 = English,

X_2 = Mathematics,

X_3 = Physical Science,

X_4 = Students Birth Date Band.

These variables were used also in the multiple regression model and the decision tree model to realize similar results as shown in Fig. 7.

Other variables or features can be considered such a as age, place of birth, household income etc. and they would have varying degrees of influence in the prediction outcome. For example, in the KNN algorithm, it has been found that the most influential variables are Mathematics, Physics and Chemistry and English is least influential.

In our algorithm, we use the Date Band as an input variable although a combination with other features can be performed as in a neural network for instance. A common standard of entry requirements for engineering schools in South Africa are Mathematics, Science and English.

It is interesting in Fig. 8, to note that there is no data available for Band 7, therefore a prediction cannot be made. This is because of the lack of historical data with which to make predictions. It is also that for Band 6 there is a 100% chance for success but it does not mean that, that position will necessarily remain the case because input data sets change with time.

When using the Date Band Algorithm, it has been found that a prediction can be made on the basis of the environmental conditions that, it is known how the class of students are doing given a lecturer and a subject and the most recent performance of that class. Consequently, it becomes a very dynamic model.

7 Conclusion

This paper has presented the Date Band Algorithm as one of the solutions to predict the probability of a student who wishes to enroll for a subject to succeed. So, the basic question that is asked by a student who wishes to take a subject is this, "What are my chances of succeeding should I take a particular subject with lecturer so and so?"

He then enters that information in the system and the algorithm will generate a probability which can be essentially regarded as an advice, a warning or information. It is in this paper, finally that the Date Band algorithm was used to predict the propensity of the student who enters the academic environment in the department of Power Engineering at the Vaal University of Technology.

References

1. Larose, T. D., Larose, D.C.: Data Mining and Predictive Analytics. Wiley (2015)
2. Agaolgu, M.: Predicting Instructor Performance Using Data Mining Techniques in Higher Education. IEEE, pp. 2379–2387 (2016)
3. Paul, R., Cowe-Falls, L.: Mapping Career Success Competencies to Engieering Leadership Capabilities. IEEE, pp. 1–6 (2015)
4. Agaolgu, M.: Predicting Instructor Performance Using Data Mining Techniques in Higher Education. IEEE (2016)
5. Callan, R.: Artificial Intelligence. Palgrave Macmillan (2003)
6. Abdella, M., Marwala, T.: Treatment of missing data using neural networks and genetic algorithms. Proceedings of the International Joint Conference on Neural Networks, pp. 598–603 (2005)
7. Han, J., Kamber, M., Pei, J.: Data Mining 3rd edn. Morgan Kaufmann Publications (2012)

8. Ministry of Education: A New Funding Framework: How Government grants are allocated to Higher Education. Public Institutions (2004)
9. Kamber, M., Winstone, L., Gong, W., Cheng, S., Han, J.: Generalization and Decision Tree Induction: Efficient Classification in Data Mining. IEEE, pp. 111–120 (1997)
10. Twala, B.: Robot Execution Failure Prediction Using Incomplete Data. Proceedings of the 2009 IEEE International Conference on Robotics and Biometrics, pp. 1518–1523 (2009)
11. Shang, W., Zhu, H.: The Improved Ontology kNN Algorithm and its Application. pp. 198–203, IEEE (2006)
12. Imbrie, P. K., Lin, J.: Work in Progress Engineering Students Change in Profile Over the Freshman Year across Male and Female Samples: A Neural Network Approach. 36th (ASEE), IEEE Frontiers in Education Conference, pp. 27–28 (2006)

ISSN 1870-4069

A Legislation-Oriented VLE-MAS System Applied to MOODLE

Maria Isabel Giusti Moreira[1], Marilton Sanchotene de Aguiar[2], Antônio Carlos Da Rocha Costa[1], Verlani Tim Hinz[1]

[1] Universidade Federal do Rio Grande do Sul (UFRGS), Brasil

[2] Universidade Federal de Pelotas (UFPel), Brasil

{isabelmoreira, vertimm, ac.rocha.costa}@gmail.com, marilton@inf.ufpel.edu.br

Abstract. The Virtual Learning Environments (VLE) are considered virtual sites that have a vast amount of resources that allow it to host courses that occur as Distance Education (EaD in Portuguese) or blended mode, favoring communication between the actors involved in these arrangements. Assign Artificial Intelligence to VLEs using Multi-Agent Systems (MAS) is a way to facilitate the learning process. By analyzing the state of the art of existing VLEs today, it can be seen that they all work as tools to aid students, but none works the management aspects of distance education supporting legislation. Therefore, this article presents a model and VLE-MAS integration system that can make the VLE able to assist managers of distance education in their tasks incorporating legislation representation. In addition, this system will work in an evolutionary way where the machine-learning method called the nearest neighbors algorithm will be applied to help the dead-time warning mechanism.

Keywords: multi-agent systems, organizational models, virtual learning environments, distance education, legal systems.

1 Introduction

In many countries, such as Brazil, Distance Education (EAD), has become a chance to democratize education, allowing social inclusion, as it can reach places where there are no educational institutions (EIs).

In Brazil, distance education is governed by legal bases that were established by the Law of Guidelines and Bases of National Education (Law no. 9,334). The legislation that supports Distance Education also regulates and determines its actions and functions, while one of the great challenges is to make the EIs meet the educational requirements, ensuring safety and quality throughout the educational process developed in this mode.

The technology that enabled the advancement of distance education was the development of focused virtual spaces to the mode called Virtual Learning Environments (VLE).

In view of the process that involves administrative nature issues, existing VLEs today, do not have any support, as well as work in a distanced way from laws, rules and regulations that the government and EIs determine for the proper conduct of distance education. Thus, VLEs are unable to enforce their players the rules that are in force, affecting the actions carried out in the environment to those standards.

This paper proposes an alternative to that deficiency in the support to legislation in the existing VLE to enable the incorporation of laws and regulations governing distance education, aiming to help on tasks related to institutional management. Based on this, the objective of this article is to present the creation of a system based on the model that allows the integration to VLE MOODLE of the main aspects of distance education legislation, intermediated by an Multi-Agent System (MAS).

One way to manipulate the interaction of agents in an MAS is the use of an organizational structure and we used an adaptation of the Moise+ model that will define the behavior of the actors in distance education process, regulating the restrictions and permissions that are set by laws and regulations [2]. In order for the model to be able to make some decisions based on event and/or actions previously performed by agents that make up society, the MAS will be applied alongside data mining techniques, making the model able to recognize some patterns displayed by the VLE user, and from that make decisions.

For the formalization of this model, the modality of distance education in a Federal Institute (FI), will be used as a case that currently offers higher education, technical and continuing education programs in the distance mode, focusing on a government program called Network e-Tec, which focuses on the provision of technical courses and the initial and continuing training of workers.

2 Multiagent Systems and their Organizational Models

The MASs allow for intelligent global behavior that can be achieved through the individual behavior of agents that compose it, based on the social behavior of a community of independent agents.

To [1], an organization is defined as a pattern that describes how the members belonging to a society are in relation to others and how they interact to achieve a common goal. To organize an MAS, it is necessary to work with independent agents, as achieving a global goal requires that autonomy offers limits. The MAS organization is a set of restrictions on the behavior of agents in order to lead to a common goal. There are several types of organizational models, but the model chosen for the development of the work was the Model of Organization for Multi-Agent+ (Moise +), developed by [2] because of its ease of incorporation into the Jason platform [3].

Moise+ considers that the goal of an organization is to restrict the agent's autonomy.

The restrictions expressed in Moise+ are indicated by the requirement of permission and prohibition. It also works with the organization of an MAS based on three dimensions, shown below.

The structural specification (SS), is based on three main concepts [2]: 1) roles: it is a set of behavioral constraints that an agent agrees to when joining a group; 2) bonds and compatibilities: the roles are related to others representing restrictions on interactions; 3) groups: represent a set of agents with higher affinity and close goals.

The functional specification (FS), consists of a set of social schemes, and a preferred ratio between missions. A social scheme is based on the idea of global targets; whose function is to represent a world state that is desired by the MAS. The main concepts related to the global targets are: missions: is a set of global targets that can be assigned to an agent through their roles; social scheme: a decomposition tree of global targets where the root is the goal of a social scheme; preferences between missions: in some cases, it is necessary to establish an order of preference between missions. The deontic specification (DE), aims at relating the SS with FS, establishing which tasks a role is allowed or required to commit to.

3 Kelsen's Legal System Concept

Incorporating the MAS to VLE as well as working with the normativity within social systems is still an existing challenge. But using these normative notions such as duty, obligation and ban facilitate the design and the computer systems specification, making them more autonomous. Based on this question, it is essential to present the issues related to normativity proposed by Hans Kelsen.

The starting point of the theory proposed by Kelsen is the idea of norm as something that determines an objective sense of must-do, highlighting the imperative character, both as imposition and ban of this must-do [5].

A rule determines how to behave in certain conditions, i.e., if a behavior and social relationships may be prescribed, prohibited or permitted. A rule indicates either conduct or omission of conduct that should be avoided by individual agents of the legal system.

According to [5], the functions of the rule of law are divided into standards: imposing, permission, authorization and derogation. The legal standards form a regulatory system that requires that those involved are in accordance with the modes of behavior stated in the rules. Each standard is derived from a higher standard. The base system is a fundamental rule of law. And a legal system that consists of a hierarchy of standards and a set of agents who supervise compliance with the rules.

4 k-Nearest Neighbors Algorithm

According to [6], the k-Nearest Neighbors algorithm (k-NN) is a powerful non-parametric algorithm used in data mining for classification and pattern recognition.

The k-NN requires a set of pre-classified examples. Based on this set of examples, a new pattern K of unknown class is compared to the set of pre-classified examples

through the relationships between attributes, that is, the K classifier seeks K elements from a set of examples nearest this unknown element for the shortest distance.

Thus, it is necessary to classify each new instance and seek K instances with the shortest distance in relation to this new instance.

There are different ways to calculate this distance, the simplest of which is the Euclidean distance (1), where pi is the value of the current line i that one wants to discover the nearest neighbor; qi is the value of each row in column i; and, maxi and mini are respectively the maximum and minimum value of all column i:

$$d(p,q) = \sqrt{\sum_{i=1}^{n} \left(\frac{(p_i - q_i)}{(max_i - min_{i)}} \right)^2}. \quad (1)$$

5 Organizational Model for a Legislation-Oriented VLE-MAS

This section aims at presenting an organizational model for VLE-MAS, based on the FI case study, to be able to integrate legal aspects, making the Legislation-Oriented VLE-MAS. To formalize this model, we will use a variation of the organizational model Moise+ by the fact that this model is able to restrict the behavior of agents.

5.1 Actors of Distance Education Mode and Structural Specification (SS)

In the FI concerned, connected to the Vice President's Office of Education, there are the actions involving distance education, which currently perform the planning and regulation of the activities of this type. Connected to the Vice President's Office, we have the campus of the FI, with its distance education departments, composed of various actors, such: Course Coordinator who is a teacher with training in related field to the course and responsible for its management; Research Professor: it is one whose role is to plan and manage all the learning development process in the subject and, Distance Tutor: will monitor the activities carried out online by students.

Based on these actors, we propose the creation of the SS of the VLE-MAS, where the roles, groups and relations between them will be presented. In distance education, we have several agents committed to the abstract role of coordinator. These are: a center, course, tutoring, general or deputy general coordinator. Agents committed to the teacher abstract role can be a content developer or research professor. The agents who assume the abstract role of the tutor can be divided into: classroom tutor and distance tutor.

In addition to these roles, there are a set of roles that take a collective level thus forming groups. Finalizing the organizational modeling of SS, in relation to the campus we have the following groups: Coordination of distance education, Center, IT Team.

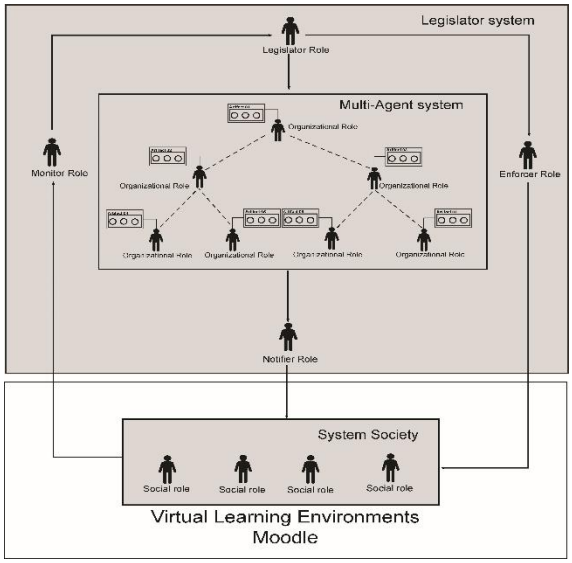


Fig. 1. Legislation-Oriented VLE-MAS System Model.

5.2 Functions and Functional Specification (FS)

Each role has a set of functions that are legislated by Law. 9.394/1996. Because of the great diversity of function, this article will present the definitions only of the role of the Course Coordinator, whose responsibilities are, for example: follow the course; carry out the academic management of the classes; monitor the activities of tutors, teachers; in-loco verification of the progress of the courses.

With these defined functions, we can determine the FS, with a set of global goals, structured in a goal decomposition tree, which is made by plans showing a way to meet this goal. In this case the tree root is the goal to be achieved and the stipulated missions determine the order in which the plans must occur so that they can achieve the specific objective.

5.3 Contractual Rules and Deontic Specification (DS)

Based on the internal rules of the FI and the contractual rules, the program can define the duties, rights, obligations to be performed by the Course Coordinator.

The rights of a course coordinator are: access to VLE-MAS; participate in training and updating activities. While the duties of this player include holding the record of the assessments undertaken by students; coordinating the preparation of the course of the project; follow the academic activities of the course. With the contractual standard, we can define the DS where we have the definition of the role will be responsible for carrying out the actions, the name of the actions, the description of the goals and legislation where it is found. While the deontic relation of this model is stipulated by

three concepts: duty, law and the prohibition of the mission. Finally, we have time restriction, because the deontic relations must have a shelf life. In addition to that, two columns were inserted, which determine which roles will monitor the action and apply punishments and who should help so the action can reach its goal.

6 Legislation-Oriented VLE-MAS System Model

To model and develop the system of the legislation-oriented VLE-MAS some fundamental aspects of systematic, hierarchical model of distance education of the FI were taken as a base, and the instrumental approaches determined by the Ministry of Education. The proposed legislation-oriented VLE-MAS system is composed of two subsystems, as shown in Figure 1, where the components, its organization and its functions will be detailed below.

6.1 Multi-Agent System

The multi-agent system is integrated within the VLE, and therefore uses its database with users and profiles that are registered.

The multi-agent system allows users to make their actions usually so the legislature system can monitor and verify the legality of such activities. It can be said that part of the system is the VLE itself. Within the multi-agent system, we have various roles called Social Agents without hierarchy. These social agents that make up the system take on roles or group roles. In this case, the system is divided into various categories, and a user can assume different roles, where, for example, a teacher may be a member of the multidisciplinary team or the teacher who will teach a course.

6.2 Legislator System

The legislator system will aim at monitoring all actions performed in the system of society in order to ensure that agents are acting within the law. If social roles are not fulfilling their goals, the system is responsible for sending notifications and applying penalties.

Still within the legislative system, a multi-agent system will be included based on an organizational model previously introduced that will work with the hierarchy of roles or role groups.

The legislator system will consist of four main roles that are to be regarded as software agents, which will interact with each other in order to monitor and evaluate the progress of actions and goals that are taking place within the company system.

The first part is called the legislator role, which is responsible for carrying out the actions to create, modify and remove rules and plans of the MAS, and is considered an issuer of standards. In this case, the legislative role will be sovereign over the rules, therefore, all tasks to create, modify and remove rules is the responsibility of this role.

The second part is called the monitor role, whose purpose is to assist in the extraction and collection of information of what is happening in the company system. This agent

is watching real-time system actions to see if the other agents are complying with the rules and plans.

The monitor roles can be classified into two types: 1) Standard Monitors: observe the standards and behavior of social roles, capturing information regarding compliance or not with those standards; 2) Plan Monitors: capture information about the current state of social roles, ensuring the implementation of plans to be undertaken to achieve the goals. In this case, the legislature system may have at least two monitoring agents.

The third role is a notified role, which aims at presenting reminders (via VLE block and email), of the outstanding actions or deadlines to be met before being penalized. In addition to that, this role will also have the purpose of showing where the rules were based upon, that is, which law, rules, regulations or contracts the rules were based upon and consequently extracted. In this case, the model will be based on deontic specification of the MAS in order to get this information. The notified also based on registered email in VLE, will trigger messages to users about pending or penalties applied to agents. Finally, the last agent is called Enforcer role, and its function is to assist in the implementation of standards and plans and will be responsible for applying the sanctions and penalties when the rules are not followed correctly.

The Enforcer role will have two subdivisions: Penalties Enforcer performing the effective implementation of sanctions and penalties prescribed in the rules of the social roles that violate it and the plan Enforcer, performing the action plans and altering the structural properties of the set of social roles. While the MAS roles, which is located within the legislative system will be the same roles used in VLE, which in this case work with the notion of hierarchy. This hierarchy was determined according to the SS of the organizational model.

This hierarchy shows the roles according to their order of autonomy and is based on certain rules that are important to that society agents where the role with lower position is always subordinate to roles with top positions. In this case, the roles are no longer called social roles to be called organizational roles that become proactive components of the legislative system, because they are linked to a set of artifacts and its main feature is autonomy.

These devices are based on the metamodel of Agents and Artifacts (A&A), [4] which incorporates a set of goals and plans. Thus, it can be said that these artifacts are the laws, rules, regulations, decrees and agreements that will determine the actions to be performed by the agents and other roles.

7 Development of the Legislation-Oriented VLE-MAS System Applied to Moodle

Initially it was necessary to conduct a research on the MOODLE main structure as well as the development of new blocks of rules. It can be seen that MOODLE is divided into two structural parts called standard and development structures.

The first structure is composed of the basic features that the environment offers, which include course management tools, user authentication and commonly used

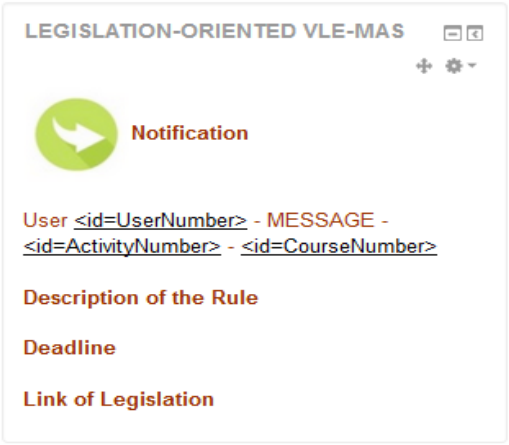


Fig. 2. Components of the Message.

functions. While the development structure consists of modules and auxiliary blocks, which have the ability to be turned on or off as needed.

Some of these blocks come together with MOODLE, but others can be developed using the programming language PHP 5 (Hypertext Preprocessor). Thus, we can say that the legislation-oriented VLE-MAS system is an example of blocks that can be developed by users.

From version 2.7, MOODLE provides a module called "Local" that enables the transformation of MOODLE in a complete development framework. Thus the legislation-oriented VLE-MAS system is being developed within the MOODLE API itself, in version 3.1, also making necessary the use of PostgreSQL.

In order to facilitate the implementation of MOODLE is taking as a basis the use of an existing MOODLE plug-in called blog, which has access to the VLE database.

8 Partial Prototype of the Legislation Oriented VLE-MAS System Applied to Moodle

The legislation-oriented VLE-MAS system when applied to MOODLE [7] notifies users of the company system through a box built into the environment called box that can be considered a MOODLE plug-in. When you activate this plug-in, the Monitor Role will review all user actions based on rules previously established by the Legislator Role and in the actions that must be performed by the system in case of compliance or non-compliance with these rules. By identifying a rule pending or that has been met, the Monitor Role in conjunction with the Notified Role will display a message box formed by the components shown in Figure 2.

In addition to that, the Monitor Role will review all actions that are not held and the Enforcer Role will conduct sanction actions on users of the system of society that fail to comply with the rule, within the stipulated limit.

The Enforcer Role always shows a message to the box on the action that was taken back to higher users as deontic specification of the organizational model MAS, as well as a message via email. The Enforcer message is composed of: User name and default message for each action performed by the Enforcer role.

To perform these simulations, the interface implementation was used, using the PHP language, built into MOODLE version 3.1. The choice of VLE MOODLE 3.1 was due to the fact that all of the FI`s distance education programs in question already use it. Based on this interface simulation, the following will be presented as an example of how MOODLE behaves with regard to compliance or non-compliance with certain rules involving the course coordinator.

A course coordinator to be selected to perform the role should make a previous registration in MOODLE. After that, he/she should post on MOODLE supporting documents to work in the role.

Figure 2 shows a diagram representing the exchange of messages held by users as well as the actions taken by the Enforcer role for the CompExp Rule.

By posting, the General Coordination will get through the Notifier role a warning that all documents have been posted. In addition to that, the manager who is hierarchically above the Course Coordinator, in case of Department Coordinator also receives a notification.

Thus, General Coordination makes manual conference documents. If the documents are all delivered with no pending issues, the Notifier role will send a return message to that user while the Enforcer Role releases access in all areas of MOODLE, with the visitor user being promoted to Course Coordinator. In the case of the documents presented including dependencies, the multidisciplinary team will mark which documents have problems choosing from a list what is pending according to the legislation. Also, it is given a period for resending the document.

An example of pending documents can be when the course coordinator does not have the certificate attesting teaching experience required by law. In this case, the multidisciplinary team, led by general coordinator and assistant, chooses in the list that the rule is pending. After this Notified role will send both the course coordinator and the department a message with the user's name, pending legislation as well as a link to the law governing this rule and the deadline for the new posting of the document.

If the document is returned on time and is in agreement, the Notified role will send a return message to that user while the Enforcer Role releases access in all areas of MOODLE. If the outstanding documents are not submitted on time and/or not in accordance with the law, the Notified role will cut user access and notify the roles that are hierarchically above the role of this action.

9 Application of the k-nn Algorithm to the MAS-VLE Model

In order for the proposed model to be capable of autonomously deciding when to show delay messages to each of the users, the k-NN algorithm was implemented through a virtual learning environment (VLE), multiagent system (MAS).

The purpose of applying the algorithm to the VLE-MAS is to prompt the system to analyze similar actions previously performed by users and, based on this analysis, define the time that current users need to perform the same action. As such, the notified role will only display messages if the user exceeds the time determined by the k-NN algorithm.

For example, the VLE-MAS system along with the k-NN algorithm analyze all the courses in the VLE in order to determine the time each professor takes to mark an activity. Based on the number of posts, number of objective and subjective questions, class hours and type of activity, the K-NN algorithm determines the nearest neighbors, that is, the time taken by other professors to mark similar activities. Should the professor exceed this time, the notified role will advise that there are X days left to mark it or that marking is X days late.

9.1 Stage 1 – Preparing the Training Data Set

In order to test the operation of the algorithm developed in PHP and applied to the VLE-MAS, a data set important to the functioning of the k-NN was extracted from the database of the Federal Institute`s (FI) MOODLE VLE. The data extracted were:

- Professor's name;
- Class hours for the course (HRS);
- Number of activities per course (NA);
- End date for posting, that is, the latest date on which students can hand in their activities, since the professor can only begin marking after this date;
- Date of marking completion, that is, the last day on which the professor marked the activities;
- Type of activity, namely 1 for forum, 2 for questionnaire, 3 for sending a file and 4 for glossary (TA);
- Number of objective questions (NOQ);
- Number of descriptive questions, which take longer to be marked (NDQ);
- Number of attempts, that is, the number of students who sent activities to be marked (NATT);
- Whether marking occurred within the deadline or not, using 1 when the deadline was exceeded and 0 when it was not. To that end, teaching guidelines stipulating that professors have 15 to 20 days to mark activities was applied (Deadline);
- Number of days the professor took to mark the activity based on the end date for posting and date of marking completion.

9.2 Stage 2 – Example 01 - Obtaining the Nearest Neighbors

Using the K-NN algorithm applied to the VLE-MAS, testing was conducted with the following data set (Table 1).

The aim is to discover how many days' professor B would take to mark activity 6. To that end, the following formula was applied (2):

Table 1. Data set – Example 01.

Teacher	Hrs	NA	Posting End Date	Marking Completion Date	TA	NOQ	NDQ	NATT	Deadline	Days
A	30	1	16/mar/15	23/mar/15	3	0	1	340	0	7
A	30	2	16/mar/15	21/mar/15	3	0	1	338	0	5
A	30	3	16/mar/15	06/apr/15	3	0	3	284	1	21
A	30	4	16/mar/15	27/mar/15	2	10	0	306	0	11
B	60	1	22/apr/15	11/may/15	4	0	1	306	1	19
B	60	2	16/mar/15	29/apr/15	3	0	10	300	1	44
B	60	3	23/mar/15	31/mar/15	1	0	1	338	0	8
B	60	4	25/mar/15	06/apr/15	2	0	5	217	0	12
B	60	5	06/apr/15	18/apr/15	3	0	1	242	0	12
B	60	6	06/apr/15	24/may/15	3	0	1	338	1	**??????**

Table 2. Score – Example 01.

Teacher	Score	Days
A	1.732127	7
A	1.624808	5
A	1.262039	21
A	1.829971	11
B	1.085724	19
B	1.24316	44
B	1.343296	8
B	1.548824	12
B	1.284197	12
B	0	**??????**

$$\sqrt{\left(\frac{(60-HRS)}{(60-30)}\right)^2 + \left(\frac{(6-NA)}{(6-1)}\right)^2 + \left(\frac{(3-TA)}{(4-1)}\right)^2 + \left(\frac{(0-NOQ)}{(10-0)}\right)^2 + \left(\frac{(1-NDQ)}{(10-0)}\right)^2 + \left(\frac{(338-NATT)}{(340-217)}\right)^2 + \left(\frac{(1-Deadline)}{(1-0)}\right)^2} \quad (1). \qquad (2)$$

At the end, a score was generated for each line of the data set (Table 2).

The scores were used to determine the shortest distance, that is, the lowest score was 1.085724, meaning that the average time Professor B took to mark the activities was 19 days. Should he exceed this number, the notified role should send a message to that effect.

At each execution, the data set is applied to the KNN algorithm and thus always updated at the end of each period, making its execution based on an evolutionary

prediction. Throughout each run, new samples are incorporated into the data set making the algorithm more accurate, causing SCORE to change with each run.

10 Final Considerations

The legislation-oriented VLE-MAS system was developed in order to assist the manual work done today by distance education managers when working with issues related to corporate or institutional management process. With the proposed model and the system, it is being possible to trace directions for automation of control roles in relation to legislation, since it became possible to carry out the systematization of knowledge about actions and plans that agents must perform.

The advantage of the model, to be incorporated into a VLE interface, allows any user who has already received training to use the model without major difficulties, since the design and simplicity of the environment allows easy and enjoyable use that leads to understanding and carrying out of actions whose legal constraints are often considered difficult to understand. In future projects, you can implement a greater number of actions as well and incorporate part of reports available to managers, with activities involving administrative matters carried out or not by users of the system of the society.

References

1. Fox, M.: The Tove project towards a common-sense model of the enterprise. Proceedings of the 5th International Conference on Industrial and Engineering Applications of Artificial Intelligence and Expert Systems, Springer-Verlag, pp. 25–34 (1992)
2. Hübner, J. F., Sichman, J. S., Boissier, O.: Moise+: Towards a structural, functional, and deontic model for mas organization. Proceedings of the First International Joint Conference on Autonomous Agents and Multiagent Systems, (ACM), pp. 501–502 (2002)
3. Boissier, O., Bordini, R., Hübner, J. F., Ricci, A., Santi, A.: Multi-agent oriented programming with JaCaMo. Science of Computer Programming, pp. 747–761 (2013)
4. Hubner, J. F., et al.: Instrumenting multi-agent organisations with orgnisational artifacts and agents. Autonomous Agents and Multi-Agent Systems, Kluwer Academic Publishers, Hingham, 20(3), pp. 369–400 (2010)
5. Kelsen, H.: Teoria Pura do Direito (2009)
6. Mitchell, T.: Machine Learning. McGraw-Hill (1997)
7. Dougiamas, M., Taylor, P. C.: MOODLE: Using learning communities to create an open source course management system. Proceedings (EDMEDIA) (2003)

ISSN 1870-4069

Collaborative Learning Team Formation Considering Team Roles: An Evolutionary Approach based on Adaptive Crossover, Mutation and Simulated Annealing

Virginia Yannibelli[1,2], Analía Amandi[1,2]

[1] ISISTAN Research Institute,
UNCPBA University Campus Universitario,
Argentina

[2] CONICET, National Council of Scientific and Technological Research,
Argentina

{virginia.yannibelli, amandi}@isistan.unicen.edu.ar

Abstract. In this paper, a hybrid evolutionary algorithm is proposed to solve a collaborative learning team formation problem in higher education contexts. This problem involves a grouping criterion evaluated satisfactorily in a great variety of higher education courses as well as training programs. This criterion is based on the team roles of students, and implies forming well-balanced teams respecting the team roles of their members. The hybrid evolutionary algorithm uses adaptive crossover, mutation and simulated annealing processes, in order to improve the performance of the evolutionary search. These processes adapt their behavior regarding the state of the evolutionary search. The performance of the hybrid evolutionary algorithm is exhaustively evaluated on data sets with very different complexity levels, and after that, is compared with those of the algorithms previously reported in the literature to solve the addressed problem. The results obtained from the performance comparison indicate that the hybrid evolutionary algorithm significantly outperforms the algorithms previously reported, in both effectiveness and efficiency.

Keywords: collaborative learning, collaborative learning team formation, team roles, evolutionary algorithms, hybrid evolutionary algorithms, adaptive evolutionary algorithms, simulated annealing algorithms.

1 Introduction

Collaborative learning is a pedagogical approach used frequently in higher education contexts, with the aim of enriching the individual learning of students. Such approach implies organizing students into collaborative learning teams to develop collaborative learning tasks.

These collaborative learning teams must be formed considering that students can gain new knowledge and also develop new skills by interacting with their peers, to improve their individual learning.

In this collaborative learning context, the grouping criterion is extremely important since the composition of the collaborative learning teams influences significantly the learning process and the social behavior of their members as well as the performance of the teams [1, 2]. Besides, the procedure used to apply the grouping criterion (i.e., either a manual or automated procedure), is important since many grouping criteria reported in the literature need a significant amount of knowledge, time and effort to be applied manually [10]. In these cases, an automated procedure could reduce the workload of professors as well as optimize the collaborative learning team formation.

In the literature, many different works have addressed the problem of automatically forming collaborative learning teams from the students [10, 4]. These works have significant differences in many aspects, particularly the grouping criteria applied, and the algorithms used. In this respect, to the best of the authors' knowledge, only few works apply grouping criteria that have been both satisfactorily and widely evaluated in higher education contexts.

The work reported in [5], formally describes the problem of automatically forming collaborative learning teams from the students enrolled in a given course, considering a grouping criterion satisfactorily evaluated in a very great variety of higher education courses as well as training programs. This grouping criterion refers to that defined by Belbin's team role model [3]. Such criterion based on the team roles of students, and implies forming well-balanced teams with respect of the team roles of their members. In this respect, a team role is defined as the way in which a person tends to behave, contribute and interrelate with others throughout a collaborative task. In the literature, many different studies indicate that collaborative learning team formation in higher education contexts considering the Belbin's criterion generates good interactions and discussions during the learning process, improves the social behavior of the students, enhances the learning process of the students, and influences very positively on the learning level of the students and also the performance of the teams [4]. Therefore, the collaborative learning team formation problem described in [5], is very valuable in higher education contexts.

The collaborative learning team formation problem described in [5], is an NP-Hard optimization problem. Because of this reason, as reported in [5], exhaustive search and optimization algorithms only can solve very small instances of the problem in a reasonable period of time. Thus, heuristic search and optimization algorithms have been proposed in the literature to solve the problem: an evolutionary algorithm was proposed in [5], a memetic algorithm was proposed in [9], which incorporates a hill-climbing algorithm into the framework of an evolutionary algorithm, and a hybrid evolutionary algorithm was proposed in [8] that integrates an adaptive simulated annealing algorithm into the framework of an evolutionary algorithm. These three algorithms utilize non-adaptive crossover and mutation processes to develop the evolutionary search.

In this paper, the collaborative learning team formation problem described in [5], is addressed with the aim of proposing a better heuristic search and optimization

algorithm to solve it. In this regards, a hybrid evolutionary algorithm is proposed that uses adaptive crossover, mutation and simulated annealing processes.

The behavior of these processes is adaptive in accordance with the state of the evolutionary search. Such adaptive crossover, mutation and simulated annealing processes are utilized in order to enhance the performance of the evolutionary search [6, 12, 13].

The above-mentioned hybrid evolutionary algorithm is proposed mainly because of the following reason. Evolutionary algorithms with adaptive crossover and mutation processes have been proven to be much more effective than evolutionary algorithms with non-adaptive crossover and mutation processes in respect of the resolution of a great variety of NP-Hard optimization problems [6, 12, 13]. Thus, the proposed hybrid evolutionary algorithm could outperform the heuristic search and optimization algorithms previously proposed to solve the problem.

The remainder of the paper is organized as follows. Section 2, describes in detail the problem addressed in this paper. Section 3, presents the hybrid evolutionary algorithm proposed. Section 4, presents the computational experiments developed to evaluate the performance of the hybrid evolutionary algorithm, and an analysis of the obtained results. Section 5, presents related works. Finally, Section 6, presents the conclusions of the present work.

2 Description of the Collaborative Learning Team Formation Problem

In this paper, the collaborative learning team formation problem described in [5] is addressed. A description of this problem is presented below.

Assume that a course S has a number of n students enrolled, $S = \{s_1, s_2, \ldots, s_n\}$, and the professor must organize these n students into g teams, $G = \{G_1, G_2, \ldots, G_g\}$. Each team G_i is composed of a number z_i of students, and each student can only belong to one team. In relation to team size, students must be organized considering the g teams have a similar number of students each. In particular, the difference among the sizes of the teams must not exceed one. The values of the terms S, n and g are known.

Regarding the students, it is assumed that they naturally play different team roles when participating in a collaborative task. A team role is the way in which a person tends to behave, contribute and interrelate with others throughout a collaborative task. In relation to the team roles which can be played by the students, the nine team roles defined in the Belbin's model [3], are considered. Table 1 shows these nine roles, with a brief description of the features of each one.

Based on the Belbin's model [3], it is assumed that each student naturally plays one or several of the nine roles presented in Table 1. In this respect, the roles naturally played by each student are known data. Such roles may be obtained by using the Belbin Team-Role Self-Perception Inventory (BTRSPI) developed by Belbin [3]. The BTRSPI determines the team roles of the persons by giving them self-evaluation tests [3].

As part of the problem, teams must be formed considering that the balance among the team roles of their members is maximized.

Table 1. Belbin's team role characteristics.

Role	Characteristics
Plant (PL)	Creative, imaginative, unorthodox. Solves difficult problems.
Resource Investigator (RI)	Extrovert, enthusiastic, communicative. Explore opportunities. Develops contacts.
Co-ordinator (CO)	Mature, confident, a good chairperson. Clarifies goals, promotes decision-making, delegates well.
Shaper (SH)	Challenging, dynamic, thrives on pressure. Has the drive and courage to overcome obstacles.
Monitor Evaluator (ME)	Sober, strategic and discerning. Sees all options. Judges accurately.
Teamworker (TW)	Co-operative, mild, perceptive and diplomatic. Listens, builds, averts friction.
Implementer (IM)	Disciplined, reliable, conservative and efficient. Turns ideas into practical actions.
Completer/Finisher (CF)	Painstaking, conscientious, anxious. Searches out errors and omissions. Polishes and perfects.
Specialist (SP)	Single-minded, self-starting, dedicated. Provides knowledge and skills in key areas.

This grouping criterion requires analyzing the balance level of the formed teams. In order to analyze such balance level, the balance conditions defined by Belbin are considered [3]. In relation to these conditions, Belbin [3], states that a team is balanced if each role included in his model is played naturally by at least one team member. Thus, in a balanced team, all team roles are naturally played. Besides, Belbin states that each role should be naturally played by only one team member [3]. Belbin states that a team is unbalanced if some roles are not played naturally, or if several of its members play the same role naturally (i.e., duplicate role) [3].

The grouping criterion above-mentioned is modeled by Formulas (1)-(3). In this respect, Formulas (1)-(2), model the balance conditions defined by Belbin [3].

Formula (1), analyzes the way in which a given role r is played within a given team G_i, and after that gives a score accordingly. If role r is naturally played by only one member of team G_i, then 1 point is awarded to G_i. Otherwise, if role r is not naturally played by any member of G_i, or role r is naturally played by several members of G_i, then 2 points and p points are taken off respectively.

Formula (2), defines the balance level of a given team G_i. This balance level is defined based on the scores obtained by G_i, through Formula (1), regarding the nine roles. Thus, the greater the number of non-duplicate roles (i.e., roles played naturally by only one member of G_i), the greater the balance level assigned to G_i. On contrary, the fewer the number of roles played naturally, or the more duplicate roles, the lower the balance level assigned to G_i. Note that the balance conditions defined by Belbin [3] can be seen in Formula (2).

By this formula, a perfectly balanced team (i.e., a team in which each one of the nine roles is played naturally by only one team member), will obtain a level equal to 9.

Formula (3) maximizes the average balance level of *g* teams defined from the *n* students of the course. Specifically, this formula aims to find a solution (i.e., set of *g* teams), that maximizes the average balance level of *g* teams. Such solution is the optimal solution to the addressed problem. In Formula (3), set *C* contains all the sets of *g* teams that may be defined from the *n* students. The term *G* represents a set of *g* teams belonging to *C*. The term *b(G),* represents the average balance level of the *g* teams belonging to set *G*. Note that in the case of a set *G* of perfectly balanced *g* teams, the value of the term *b(G),* is equal to 9.

For a more detailed discussion of Formulas (1)-(3), readers are referred to [5].

$$nr(G_i, r) = \begin{cases} 1 & \text{if } r \text{ is naturally played by only one member of } G_i, \\ -2 & \text{if } r \text{ is not naturally played in } G_i, \\ -p & \text{if } r \text{ is naturally played by } p \text{ members of } G_i. \end{cases} \quad (1)$$

$$nb(G_i) = \sum_{r=1}^{9} nr(G_i, r), \quad (2)$$

$$\max_{\forall G \in C} \left(b(G) = \frac{\sum_{i=1}^{g} nb(G_i)}{g} \right). \quad (3)$$

3 Description of the Hybrid Evolutionary Algorithm

To solve the problem, a hybrid evolutionary algorithm is proposed. This algorithm uses adaptive crossover, mutation and simulated annealing processes. The behavior of these processes is adaptive regarding the state of the evolutionary search. The use of adaptive crossover, mutation and simulated annealing processes is meant to enhance the performance of the evolutionary search, in both exploration and exploitation of the search space [6, 12, 13].

The general behavior of this hybrid evolutionary algorithm is described as follows. Assuming a course with *n* students enrolled who should be organized into *g* teams, the algorithm creates a random initial population of feasible solutions. In such population, each one of the solutions encodes a feasible set of *g* teams that may be defined from the *n* students. Once the initial population is created, the algorithm both decodes and evaluates each solution of this population by a fitness function. Specifically, the set of *g* teams represented by each solution is built, and after that evaluated in relation to the optimization objective of the problem. As was mentioned previously in Section 2, this objective is to maximize the balance level of the *g* teams formed from the *n* students. Therefore, the fitness function analyzes the balance level of the *g* teams represented by

each solution, and then defines a fitness level for each solution. In order to develop such analysis, the fitness function based on knowledge of the students' team roles.

After each one of the solutions of the population is evaluated, a parent selection process is utilized to determine which solutions of the population will compose the mating pool. In this respect, the highest fitness solutions will have more likelihood of being selected. Once the mating pool is composed, the solutions within the mating pool are organized in pairs. After that, a crossover process is applied to each pair of solutions with an adaptive probability AP_c, in order to generate new feasible solutions. After that, a mutation process is applied to each one of the solutions obtained by the crossover process, with an adaptive probability AP_m. After that, a survival selection process is applied to determine which solutions from the solutions into the population and the solutions generated from the mating pool will compose the new population. Finally, an adaptive simulated annealing algorithm is applied to each solution within the new population, except to the highest fitness solution of this population which is maintained.

The described process is repeated until a given number of generations is achieved.

a. Encoding of Solutions

The representation proposed in [5], is used in order to encode the solutions. Thus, each solution is encoded as a list with as many positions as students enrolled in the course (i.e., n positions). Each position j ($j = 1,\ldots, n$), on this list contains a different student (i.e., repeated students are not admitted). Besides, each student s_k ($k = 1,\ldots, n$), may be in any position on the list. This list is a permutation of the n students.

To decode the set G of g teams from the list, the decoding process proposed in [5], is utilized. By using this process, the list is divided into g segments, considering that the difference among the sizes of the segments must not exceed one. Each segment represents to a different team.

b. Fitness Function

To evaluate the encoded solutions, a specially designed fitness function is used. Given an encoded solution, this function decodes the set G of g teams represented by the solution. The decoding is developed by the process mentioned in Section 3.1. Then, the function calculates the value of the term $b(G)$, corresponding to G (Formulas (1)-(3)). This value represents the average balance level of the g teams composing the set G, and thus, determines the fitness level of the encoded solution.

c. Parent Selection Process

To develop the parent selection, the well-known roulette wheel selection process [6], is utilized. In this process, a selection probability is defined for each solution of the population. The probability of each solution is proportional to its fitness level. Thus, the highest fitness solutions have more probability of being selected for the mating pool.

d. Adaptive Crossover and Adaptive Mutation Processes

In relation to the crossover and mutation processes, processes feasible for solutions encoded as permutations of n students are utilized.

To develop the crossover, the process named partially mapped crossover [6], is applied. This process generates two new feasible encoded solutions (i.e., two new permutations of the n students) from a given pair of encoded solutions. This crossover process is one of the most applied for permutations of n elements in the literature [6].

In order to develop the mutation, the process named insert mutation [6], is applied. This process generates a new feasible encoded solution (i.e., a new permutation of the n students), from a given encoded solution. This mutation process is one of the most used for permutations of n elements in the literature [6].

These crossover and mutation processes are applied with adaptive crossover and mutation probabilities, respectively. In this regards, an adaptive crossover probability named AP_c and an adaptive mutation probability named AP_m are defined by Formulas (4)-(7). In these formulas, the term S refers to the state of the evolutionary search, and the term S_W refers to the widest possible state of the evolutionary search. In Formula (4), C^L and C^H refer to the lower and upper bounds for the crossover probability, respectively. In Formula (5), M^L and M^H refer to the lower and upper bounds for the mutation probability, respectively. The term f_{max} refers to the maximal fitness within the population, f_{min} refers to the minimal fitness within the population, and f refers to the fitness of the solution to be mutated.

The term S is defined by Formula (6). In this formula, f_{max} refers to the maximal fitness within the population, f_{avg} refers to the average fitness of the population, and the term $(f_{max} - f_{avg})$ refers to a well-known measure of the state of the evolutionary search [7, 6].

The term S_W is defined by Formula (7). In this formula, the term f_{MAX} represents to the maximum fitness value possible (i.e., the upper bound of the fitness function), and the term f_{MIN} represents to the minimum fitness value possible (i.e., the lower bound of the fitness function).

Through Formulas (4)-(7), probabilities AP_c and AP_m are adaptive according to the state of the evolutionary search. In this respect, when the evolutionary search starts to converge, probabilities AP_c and AP_m are increased, to promote the exploration of new regions of the search space, and therefore, to avoid the premature convergence of the evolutionary search. In contrast, when the evolutionary search is well scattered in the search space, probabilities AP_c and AP_m are reduced, to promote the exploitation of known regions of the search space. Thus, probabilities AP_c and AP_m are adaptive to promote either the exploration or exploitation of the search space, according to the state of the evolutionary search.

Through Formula (5), probability AP_m is also adaptive regarding the fitness of the solution to be mutated. In this sense, lower values of AP_m are assigned to high-fitness solutions, whereas higher values of AP_m are assigned to low-fitness solutions. This adaptation has the aim of preserving high-fitness solutions, when the exploration of the search space is promoted.

$$AP_c = \left(\frac{S_W - S}{S_W} \right) * \left(C^H - C^L \right) + C^L, \tag{4}$$

$$AP_m = \left(\frac{f_{\max} - f}{f_{\max} - f_{\min}} \right) * \left(\frac{S_W - S}{S_W} \right) * \left(M^H - M^L \right) + M^L, \tag{5}$$

$$S = \left(f_{max} - f_{avg} \right), \tag{6}$$

$$S_W = \left(f_{MAX} - f_{MIN} \right). \tag{7}$$

e. Survival Selection Process

To develop the survival selection, the classical fitness-based steady-state selection process [6], is utilized. In this process, the worst λ solutions of the current population are replaced by the best λ solutions generated from the mating pool. This process preserves the highest fitness solutions reached by the hybrid evolutionary algorithm [6].

f. Adaptive Simulated Annealing Algorithm

The general behavior of the applied adaptive simulated annealing algorithm, which is a variation of the algorithm presented in [8], is described as follows.

This adaptive simulated annealing algorithm is mainly an iterative process. This process begins considering a given encoded solution *s*, and a given initial value T_0 for the parameter named temperature. In each of the iterations, a new encoded solution *s'* is created from the current encoded solution *s*, by applying a move operator. After that, the algorithm analyzes if solution *s* should be replaced or not by solution *s'*. When the fitness value of solution *s'* is better than that of solution *s*, the algorithm replaces to solution *s* by solution *s'*. In contrast, when the fitness value of solution *s'* is worse than or equal to that of solution *s*, the algorithm replaces to solution *s* by solution *s'* based on an acceptance probability which is $exp(-\Delta/T_c)$. In this probability, term T_c is the current value of the temperature parameter, and Δ is the difference between the fitness values of solutions *s* and *s'*. Thus, the acceptance probability is directly proportional to the current value of the temperature parameter.

The above-described process is repeated until a given number *I* of iterations is reached. It is necessary to mention that, at the end of each of the iterations, the value of the temperature parameter is reduced by a given cooling factor α.

In relation to the initial value T_0 of the temperature parameter, this value is defined based on the evolutionary search state S_P reached after the survival selection process, considering that such state is measured by calculating Formula (6), on the population obtained by the survival selection process. Specifically, the value T_0 is calculated by using the next formula: $T_0 = 1 / S_P$.

By this formula, when the evolutionary search is scattered in the search space, the value T_0 is low, and thus the acceptance probability of the algorithm is also low.

Consequently, the algorithm promotes the exploitation of known regions of the search space. When the evolutionary search starts to converge, the value T_0 increases, and therefore the acceptance probability of the algorithm also increases. Consequently, the algorithm promotes the exploration of new regions of the search space. Based on the mentioned, the algorithm is adaptive according to the state of the evolutionary search, in order to promote either the exploitation or exploration of the search space.

In relation to the move operator of the simulated annealing algorithm, an operator feasible for solutions encoded as permutations of *n* students is applied. Specifically, the operator named swap mutation [6], is applied. This operator creates a new encoded solution (i.e., new permutation of the *n* students), from a given encoded solution. This operator is one of the most applied for permutations of *n* elements in the literature [6].

4 Computational Experiments

To evaluate the performance of the hybrid evolutionary algorithm, the ten data sets introduced in [5], were used. Each data set contains a number *n* of students, and details a *g* number of teams to be built from the *n* students. In addition, each data set details the team roles of each of its *n* students, considering that these team roles belong to the Belbin's model [3]. The main characteristics of these ten data sets are shown in Table 2. For a description of the team roles of the students in each of the data sets, readers are referred to [5].

Each one of the ten data sets has a known optimal solution with a fitness level of 9. Note that a solution with a fitness level of 9 contains a set of perfectly balanced *g* teams, regarding the balance conditions defined by Belbin [3]. These known optimal solutions are considered here as references to evaluate the performance of the hybrid evolutionary algorithm.

The hybrid evolutionary algorithm was run 30 times on each of the data sets. After each of the runs, the algorithm provided the best solution found. In order to carry out

Table 2. Main characteristics of the ten data sets used.

Data set	Number of students enrolled (*n*)	Number of teams to be formed (*g*)
1	18	3
2	24	4
3	60	10
4	120	20
5	360	60
6	600	100
7	1200	200
8	1800	300
9	2400	400
10	3000	500

these runs, the algorithm parameters were set as follows: size of the population = 80; generations = 200; crossover process: C^L = 0.5 and C^H = 0.9; mutation process: M^L = 0.01 and M^H = 0.2; survival selection process: λ = 40; simulated annealing algorithm: I = 20 and α = 0.9. It is necessary to mention that the algorithm parameters were set with these values based on exhaustive preliminary experiments.

By these preliminary experiments, many different settings were considered for the algorithm parameters, and then the best of these settings was selected for the algorithm parameters.

Table 3, presents the results obtained for each data set. Column 1 presents the name of each data set. Column 2 presents the average fitness value of the solutions reached for each of the data sets. Column 3 presents the average computation time of the runs performed on each data set. The experiments were performed on a personal computer Intel Core 2 Duo at 3.00 GHz and 3 GB RAM under Windows XP Professional Version 2002. The algorithm was implemented in Java.

The results in Table 3, were analyzed considering that each data set has a known optimal solution with a fitness level of 9. For the first seven data sets (i.e., the seven less complex data sets), the algorithm reached an optimal average fitness value. This means that the algorithm found an optimal solution (i.e., a set of perfectly balanced g teams), in each run. For the last three data sets (i.e., the three more complex data sets), the algorithm reached an average fitness value higher than 8.75. This means that the algorithm found near-optimal solutions for each of the data sets.

The composition of such solutions was exhaustively analyzed, noting that these solutions contain a very high percentage of perfectly balanced teams. Based on these results, the algorithm reached very high-quality solutions for the problem instances represented by the data sets.

Table 3. Results obtained by the hybrid evolutionary algorithm for each data set.

Data Set	Fitness Value	Time (seconds)
1	9	0.18
2	9	0.46
3	9	3.78
4	9	6.01
5	9	14.9
6	9	19.03
7	9	72.4
8	8.87	133.07
9	8.81	211.18
10	8.76	303.84

Table 4. Results obtained by the algorithms previously proposed for the addressed problem.

Data set	Evolutionary algorithm [5]		Memetic algorithm [9]		Hybrid algorithm [8]	
	fitness value	**time (s)**	**fitness value**	**time (s)**	**fitness value**	**time (s)**
1	9	0.5537	9	0.42	9	0.29
2	9	1.3741	9	1.03	9	0.721
3	9	11.0669	9	8.30	9	5.81
4	9	17.5976	9	13.20	9	9.24
5	8.8	40.8722	8.92	30.65	9	21.46
6	8.76	55.7548	8.86	41.82	8.97	29.27
7	8.7	196.9964	8.78	147.75	8.86	103.43
8	8.64	362.0328	8.68	271.52	8.77	190.1
9	8.61	574.6589	8.65	430.994	8.74	301.69
10	8.592	771.6553	8.62	578.74	8.7	405.118

As regards the average computation time required by the algorithm, the following may be mentioned. For the first six data sets (i.e., the six less complex data sets), the average time required was lower than 20 seconds. For the last four data sets (i.e., the four more complex data sets), the average time required was higher than 72 seconds and lower than 304 seconds.

Taking into account the complexity level of the problem instances inherent to the ten data sets, particularly the complexity level of the problem instances inherent to the four more complex data sets, it is considered that the average times required by the algorithm are acceptable.

a. Comparative Analysis with Competing Algorithms

To the best of the authors' knowledge, only three heuristic search and optimization algorithms have been previously proposed in the literature for solving the addressed problem: a traditional evolutionary algorithm [5], a traditional memetic algorithm [9], which incorporates a hill-climbing algorithm into the framework of an evolutionary algorithm, and a hybrid evolutionary algorithm [8], which incorporates an adaptive simulated annealing algorithm within the framework of an evolutionary algorithm. These three algorithms use non-adaptive crossover and mutation processes to develop the evolutionary search.

Based on the computational experiments reported in [8, 5, 9], the three algorithms have been evaluated on the ten data sets presented in Table 2, and have obtained the results that are shown in Table 4.

These experiments have been performed on a personal computer Intel Core 2 Duo at 3.00 GHz and 3 GB RAM under Windows XP Professional Version 2002. The algorithms have been implemented in Java.

The results in Table 4 indicate that the algorithm proposed in [8] is the best of the three algorithms. Below, the performance of this algorithm is compared with that of the hybrid evolutionary algorithm proposed here. For sake of simplicity, the algorithm proposed in [8] will be referred as algorithm H.

The results in Tables 3-4 indicate that the hybrid evolutionary algorithm proposed here and the algorithm H reached the same average fitness value (i.e., an optimal average fitness value), for the first five data sets (i.e., the five less complex data sets). Nevertheless, the average fitness value reached by the hybrid evolutionary algorithm for each of the last five data sets (i.e., the five more complex data sets), is much higher than that reached by the algorithm H. In particular, the hybrid evolutionary algorithm reached optimal average fitness values for the data sets 6-7. These results mean that the quality of the solutions achieved by the hybrid evolutionary algorithm for the five more complex data sets is much better than that of the solutions achieved by the algorithm H. Furthermore, the average computation time of the hybrid evolutionary algorithm for each data set is significantly lower than that of the algorithm H.

Based on these results, the performance of the hybrid evolutionary algorithm on the five more complex data sets is much better than that of the algorithm H, regarding the quality of the solutions and also the computation time. This is mainly because of the following reasons.

The hybrid evolutionary algorithm proposed here integrates adaptive crossover and mutation processes. The behavior of these processes is adaptive regarding the state of the evolutionary search. This adaptation is meant with the aim of promoting either the exploitation or exploration of the search space, and thus, to enhance the evolutionary search. In contrast with the hybrid evolutionary algorithm, the algorithm H utilizes non-adaptive crossover and mutation processes. These processes disregard the state of the evolutionary search, and therefore, do not have the possibility of enhancing the evolutionary search.

5 Related Work

Many different studies in the literature indicate that collaborative learning in higher education environments significantly benefits from the application of the Belbin's model [10, 4]. However, to the best of our knowledge, only few works in the literature address the problem of automatically forming collaborative learning teams based on the Belbin's model. These works differ in several aspects, including the modelling of this collaborative learning team formation problem, and the algorithms used to form the collaborative learning teams. In this section, we review related works reported in the literature, focusing the attention on analysing the aspects above-mentioned

Ounnas et al. [11], proposed a framework which utilizes an ontology to describe students' characteristics including Belbin's team roles. This framework provides a list with grouping criteria that includes forming teams based on the Belbin's model.

In this framework, the team formation problem is modeled as a constraint satisfaction problem. The weak constraints of the problem refer to the grouping criteria selected by the professor from the provided list, and the optimization objective of the problem is to find the set of teams that minimizes the number of violated weak constraints. The problem is solved by a DLV constraint satisfaction solver (i.e., an exhaustive search algorithm).

Alberola et al. [4], proposed a tool based on the Belbin's team role model. This tool aims to build well-balanced teams regarding the team roles of their student members. In this case, the team formation problem is modeled as a coalition structure generation problem and is solved by means a linear programming method (i.e., an exhaustive search algorithm). In this tool, the team roles of students are estimated from the feedback given by the other students, by using Bayesian learning. Although this is meant to avoid the drawbacks of the Team Role Self-Perception Inventory, the estimation of the students' roles could be negatively affected by biased feedback.

The above-mentioned works utilize different exhaustive search algorithms to form the teams. However, this kind of algorithms only can solve very small instances of the problem in a reasonable period of time. Therefore, heuristic search and optimization algorithms are required to solve the problem.

Yannibelli and Amandi [5, 9, 8], proposed very different evolutionary algorithms in order to solve problem instances with different complexity levels. These algorithms, in particular the algorithm proposed in [8], reached promising results regarding both effectiveness and efficiency. Nevertheless, these algorithms based on non-adaptive crossover and mutation processes to develop the evolutionary search. These processes disregard the evolutionary search state, and therefore, do not have the possibility of enhancing the performance of the evolutionary search, regarding both effectiveness and efficiency.

Unlike the evolutionary algorithms proposed in [5, 9, 8], the hybrid evolutionary algorithm proposed here uses adaptive crossover and mutation processes. These processes adapt their behavior regarding the evolutionary search state, to enhance the performance of the evolutionary search.

6 Conclusions and Future Work

In this paper, the collaborative learning team formation problem described in [5], was addressed. This problem involves a grouping criterion that has been both satisfactorily and widely evaluated in higher education contexts. Such criterion corresponds to that defined by Belbin's team role model [3].

For solving the addressed problem, a hybrid evolutionary algorithm was proposed. This proposed hybrid evolutionary algorithm integrates adaptive crossover, mutation and simulated annealing processes. These processes adapt their behavior regarding the state of the evolutionary search. The integration of such adaptive processes is meant to enhance the performance of the evolutionary search [6, 12, 13].

The performance of the hybrid evolutionary algorithm was evaluated on ten data sets with very different complexity levels.

Subsequently, the performance of this algorithm on these data sets was compared with those of the algorithms previously reported in the literature for solving the addressed problem. Based on the obtained results, it may be stated that the proposed hybrid evolutionary algorithm considerably outperforms the previous algorithms.

In future works, the incorporation of other adaptive processes into the framework of the evolutionary algorithm will be evaluated. In particular, other adaptive crossover and mutation processes, as well as adaptive selection processes, will be exhaustively evaluated. Furthermore, the incorporation of other search and optimization techniques into the framework of the evolutionary algorithm will be evaluated.

References

1. Barkley, E. F., Cross, K. P., Howell-Major, C.: Collaborative learning techniques. John Wiley & Sons (2005)
2. Michaelsen, L. K., Knight, A. B., Fink, L. D.: Team-based learning: A transformative use of small groups in college teaching. Stylus Publishing (2004)
3. Belbin, R. M.: Team Roles at Work. Taylor & Francis (2011)
4. Alberola, J., Del-Val, E., Sanchez-Anguix, V., Palomares, A., Teruel, M.: An artificial intelligence tool for heterogeneous team formation in the classroom. Knowledge-Based Systems, 101(1), pp. 1–14 (2016)
5. Yannibelli, V., Amandi, A.: A deterministic crowding evolutionary algorithm to form learning teams in a collaborative learning context. Expert Systems with Applications, 39(10), pp. 8584–8592 (2012)
6. Eiben, A. E., Smith, J. E.: Introduction to Evolutionary Computing, Springer (2015)
7. Srinivas, M., Patnaik, L. M.: Adaptive probabilities of crossover and mutation in genetic algorithms. IEEE Transactions on Systems, Man and Cybernetics, 24(4), pp. 656–667 (1994)
8. Yannibelli, V., Amandi, A.: A Hybrid Algorithm Combining an Evolutionary Algorithm and a Simulated Annealing Algorithm to Solve a Collaborative Learning Team Building Problem. J.-S. Pan et al. (eds.), (HAIS), LNCS, Springer, Heidelberg, 8073, pp. 376–389 (2013)
9. Yannibelli, V., Amandi, A.: A memetic algorithm for collaborative learning team formation in the context of software engineering courses. Cipolla-Ficarra, F., Veltman, K., Verber, D., Cipolla-Ficarra, M., Kammüller, F. (eds.), (ADNTIIC), LNCS, Springer, 7547, pp. 92–103 (2012)
10. Cruz, W. M., Isotani, S.: Group Formation Algorithms in Collaborative Learning Contexts: A Systematic Mapping of the Literature. Baloian, N., Burstein, F., Ogata, H., Santoro, F., Zurita, G. (eds), Collaboration and Technology, (CRIWG), LNCS, Springer, 8658, pp. 199–214 (2014).
11. Ounnas, A., Davis, H. C., Millard, D. E.: A Framework for Semantic Group Formation in Education. Educational Technology & Society, 12(4), pp. 43–55 (2009)
12. Rodriguez, F.J., García-Martínez, C., Lozano, M.: Hybrid metaheuristics based on evolutionary algorithms and simulated annealing: taxonomy, comparison, and synergy test, IEEE Trans. Evol. Comput., 16(6), pp. 787–800 (2012)
13. Talbi, E.: Hybrid Metaheuristics. (SCI), Springer, 434 (2013)

ISSN 1870-4069

Contextual Variables Relationships and their Effect on Recommender Systems

Aram González, Jorge A. Ramírez

Tecnológico de Monterrey, Campus Estado de México,
Mexico

{a00965707@itesm.mx, juresti}@itesm.mx

Abstract. The internet in our daily lives has made a great impact in the availability of information. However, the quality and relevance of the data may vary depending on what each person is looking for. Since every person has different interests and priorities, data can mean nothing to someone while being a top priority for someone else. Information retrieval and recommender systems were created to alleviate this phenomenon. Traditional recommenders do this by considering the users' ratings and preferences over a set of items. Following the success of collaborative filtering and other methods, and as part of their natural evolution, recommender systems began to include more information into their learning process. Data about the users, items and their surroundings are common trends to improve the recommendation process. The inclusion of new data sources lead to what today is known as context-aware recommender systems. This type of systems include data regarding the situations in which the system, user and item are involved when dealing with the recommendation process in previous and current interactions. In this paper, we present a model for contextual data capable of finding and modeling the relationships between the context variables, the users' preferences and the items' characteristics. The system generates recommendations with specific data in the queries requested by the users. Our model considers a general situation of the user's contextualized preferences. After we generate the initial model, we use collaborative filtering techniques to deal with specific characteristics presented by each item to generate our recommendation.

Keywords: context-aware recommender systems, Bayesian model, user model.

1 Introduction

A recommender system's objective is to give the best available suggestions to users over a set of items. This is often achieved with methods such as collaborative filtering algorithms or regression models.

Thus, the main goal is simplified and becomes the prediction of the item's score. However, the metrics for evaluating such algorithms have been previously discussed and found that they are not entirely effective and meaningful [6].

As we know, data from different sources has been used to achieve this goal. However, most of the acquired information has been used as a filtering condition. Even with the current models generated by the mentioned data, they tend to only describe the user's data and do not explicitly consider the relation between the data.

The evolution of context-aware recommender systems is a good example of the inclusion of surrounding data with the hope that it will improve the system's recommendations. From the start, and because of collaborative filtering's popularity, the contextual data is commonly used as part of the filtering process.

The data inside recommender systems is traditionally represented as matrices. This allows researchers to treat them as input data for collaborative filtering algorithms. The problem with this approach is that it tries to look for similarities between users or items in order to predict a score, while it ignores or, in the best case scenario, implicitly finds the relations between the factors used to compare the similarities between the users or items.

When using this approach, few things can be done with that relation's knowledge. Using a different approach may prove useful. By having the concrete model, it can be adjusted and refined so that it becomes a better representation of the user.

The information from the users about their ratings over a set of items can be seen as an abstract representation of them, giving us simple user models to work with. However, by excluding contextual information, we lose the interpretive power that is available to us to determine the user's goals and objectives. This will also prevent us from giving the best choice in the best possible way [5].

The relevance of a proper user model is based on the fact that users decisions are not only affected by the presence of distinct factors, but also by the interaction between them. That is, for example, the weather context variable may greatly affect the user's mood when rain is present. But in a sunny day, the weather context variable may not have the same impact on the mood of the user. Both cases have an impact on the user's final decision, but the same context variable has a different weight depending on the current condition.

There are comparative studies between representative members of context-aware recommender systems' approaches. Paniello *et al.* [11] conclude that contextual modeling approaches are some sort of best bet when deciding for an approach, even if they do not exceed the rest in every aspect considered in their evaluation. We decided to follow the contextual modeling approach to take advantage of an existing explicit model for our users.

Being part of the decision process, the relations between the context variables seem to be a relevant part to be considered. This requires a structure that is capable to represent the relations.

For this purpose, we decided to use Bayesian networks as the base for our user model. Bayesian networks are a graphical representation of conditional probabilities tables. They can be used to establish a causal relation between its components.

2 Previous Work

Collaborative filtering and techniques that combine it with other approaches are the most common approach to traditional recommender systems. Collaborative filtering recommender systems generate suggestions based on the similarity existing between the users in the system. It comes from the human behavior of rating items based on the opinions of other people [13].

Collaborative filtering methods are designed to produce recommendations personalized for users based on patterns of ratings without external information for items or users. So, by design, the basic data for collaborative filtering methods mostly depends on the ratings.

Adomavicius and Tuzhilin [1] established that collaborative filtering recommender systems try to predict the utility of a specific item based on the scores given by other users. This a broad definition of this type of approach. This definition means that every system that uses data from distinct users to help determine a specific user's rating for an item can be associated with collaborative filtering.

This is a common approach given the lack of data (ratings), given by a single user over a extended period of time. For our approach, we use the data from different users to establish our initial model that is shared by all the users. This model changes over time by both, the implicit and explicit feedback from users. The implicit part requires users natural interaction with the system, this means it needs them to use and rate the items presented. This allows the system to recalculate both the network's structure and its probabilities over time.

The explicit part comes from the user directly modifying the set of rules presented to them. This can modify the way in which each query is procesed. While this doesn't directly modify the calculated probabilities, it changes the form in which they are used.

Up until this point we could consider our approach to be partially collaborative filtering. However, the algorithms commonly presented as collaborative filtering focus their research on solely on improving their accuracy on the predicted ratings.

This has proven to be a common research area that allows recommender systems to be very accurate on their predictions. In fact, it's still a relevant research area that is presented in the works of Liu *et al.* [8], Nilashi *et al.* [9], and Ramezani *et al.* [12], just to mention some recent works.

While we can produce recommendations and inference over the possible ratings for current items. Its main focus is on the relations that exist between the data it considers, it being a combination of user, item, and contextual characteristics. And the second main focus is on the personalization of the experience by giving two ways to modify the current process.

This gives our approach the chance to present useful information both to the traditional recommender system's user, but it also allows to generate a data analysis on the background information that it uses in order to produce its results.

The explicit relation between context variables has also been a subject of study. A recent case is the one proposed by Baltrunas *et al.* [3]. They also came to the conclusion that context variables have a different relevance. While it still lacked the focus on the relations between context variables, it ends up being another step into the course of action that we follow.

Ono *et al.* [10], proposed a Bayesian network approach as the main user model for a context-aware recommender system. In their paper, the model included data from the user, the context and the item. They managed to obtain a big number of attributes for each system's element. While the overall process is similar to our approach, the construction of the Bayesian networks is different. Ono *et al.* focus their efforts in dealing with a lot of attributes. In contrast, our main objective is to adjust the Bayesian network's structure and conditional probabilities so that both match the current user and context. Also our approach adds a feedback process to let each user further the personalization of his model.

3 Data

The lack of context representation in recommmender systems may lead to a loss in predictive power. Adomavicius and Tuzhilin [2], expose this point. By acknowledging this, we decided to work with a context-aware recommender system as our base.

While looking for a dataset, we were not able to find one with enough number of context variables and enough records to support the presented approach. The LDOS-CoMoDa dataset [7], satisfies most of the contextual requirements, but it lacks the amount of records required to show the actual differences between the created user models.

For this reason, we decided to develop our own dataset. We are currently working a vacation dataset. At the time this paper was written, we have 240 surveys collected. The answers are related to a group of people with similar characteristics as our movies dataset.

This dataset is focused in vacation recommendation. We asked about the place to which every user last traveled, the dates, companions, experiences, activities, lodging's characteristics as well as a rating for both the place and the experience. On this new dataset, we tried to focus more on the information that involves the entire trip and that can affect the traveler's experience.

The dataset is aimed at describing in the most detailed way possible the circumstances in which a person decides to take a vacation. So we included traditional contextual variables used on other works like season of the year, weather and time spent. But we also included data about the conditions during the trip, characteristics of the visited places and we also included the user's opinion about the lodging choices and what the elements people use to choose and rate them.

4 Bayesian Network as Model for Contextual Data

A Bayesian network allows us to visually represent the conditional probabilities of our system. A Bayesian network, even when data is missing, can learn the causal relationships between the domain's problems and characteristics. It also helps to predict future events by inferring how likely it is for them to appear [4].

We decided to use Hill Climbing and the Bayesian Information Criterion to obtain our conditional probabilities with the structure of our network. We decided to stick to the Bayesian Information Criterion because, after testing methods like the Akaike Information Criterion and Gaussian log-likelihood, we found no evidence that they outperformed the original score. In fact, Akaike results were really close to Bayesian information Criterion.

By analyzing the data, we found the same tendency in the expected error. This is, that it tends to diminish as we get more data. However, we also found that a correlation exist between the number of variables and the rate in which the expected error diminishes. Still, by using our approach, we are able to generate recommendations and generate some information about the decision process, albeit it might not be as accurate as when more data is available.

The rate in which the expected loss decreases is shown in Table 1 and in Figure 1. After some tests with 10 fold cross validation, the system showed a great amount of instability when tested with less than 150 answers.

Table 1. Expected loss function from our vacation dataset.

Data quantity	Expected loss
50	1104.629
75	∞
100	118.9803
125	1807.76
150	81.91782
175	64.09887
200	64.67456
225	73.79942
240	82.79571

The source of this behaviour can be explained as:

1. Amount of variables - The number of variables used included in this dataset is around 90. This includes the item's features as well as context variables. This makes it harder for the Bayesian network to accurately represent and predict the behaviour of the user.
2. Data outliers - While collecting our data, we found that some of the points are spread as shown in Figure 2. We have some data outliers that complicate the learning process of our model. Since it seems to be less identifiable patterns, the amount of data required for stabilization increases.

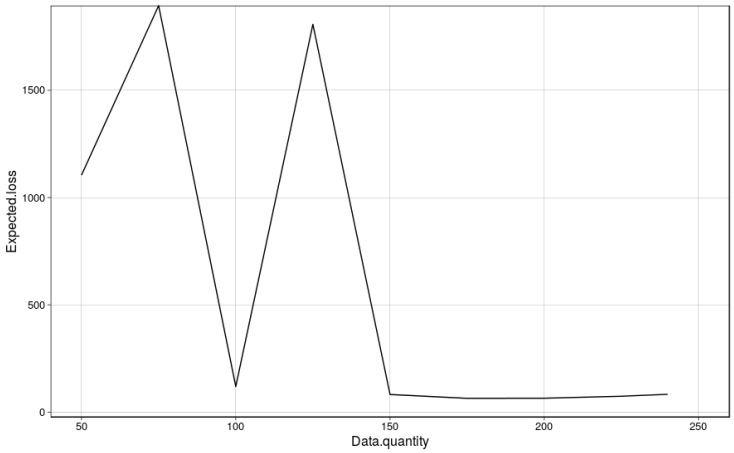


Fig. 1. Expected loss function from our vacation dataset.

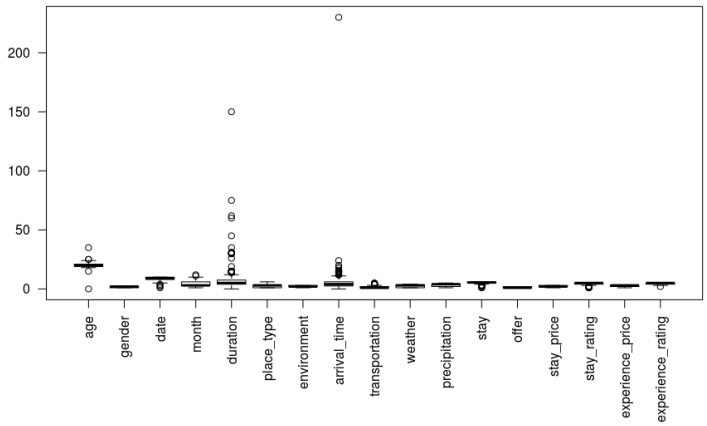


Fig. 2. Vacation dataset current status.

As an initial result, we obtain the Bayesian network's structure and the conditional probabilites behind it. Table 2 shows an example of the conditional probabilities of the types of places that users go on vacation.

We can begin to observe some of the effects of the variables by looking at the conditional probabilities tables generated with our Bayesian network. For example, Table 3 shows how the quality and rating of the place in which a person sleeps has a direct impact on the score the user gives to the whole trip. While some of this relations may seem obvious, it is relevant to ensure that our approach can determine them and that it can also find some new relationships that we might not really account for.

Table 2. Probabilies of the type of places that users will visit on vacation.

Place type	Probability
City	0.3064667
Mountain	0.1419333
Beach	0.1760667
Town	0.1562667
Remote place	0.08686667
Historical site	0.03406667

Table 3. Probability of experience scores given a move score of 4 out of 5 or more given the stay rating.

Stay rating	Probability
Between 0 and 1	0.155595
Between 1 and 2	0.306841
Between 2 and 3	0.4776664
Between 3 and 4	0.7134526
Between 4 and 5	0.9094099

4.1 Getting the First Step of the Recommendation

The first phase prediction of this particular recommender system is based on the inference process of the Bayesian networks. The inference is done by calculating the posterior probabilities from our conditional probabilities. The posterior probabilities can be calculated by applying the Bayesian rule (and the chain rule).

So we feed our particular queries with the information that is currently available. By using this type of inference, we can create complex queries for the system. The queries may include any of the variables we used for building our network and it will return us the probability of that being true. For example, if we wish to know if a user would like to have a swimming pool in his hotel in a winter vacation, we generate a query like this:

$$p(vacation\_rating >= 4 | season == summer | hotel\_pool == True).$$

An advantage of this approach is that we can get information about any of the variables presented in our network. So if we like to know which season is preferred by our users on winter, we can make a query to ask that. The queries can be as general or specific as we require them to be.

As a result we get a probability that indicates us how probable it is that our current user will enjoy a trip with those characteristics. While this approach is particularly suitable to ask for specific situations, it requires a query for each posibility when searching for each possible recommended item.

This gives our system a flexibility that is not present in other approaches. We can do queries with as little as no contextual data, and also do queries with up to every variable specified so that it can find the best match for our user.

This approach can also lead to obtain a better analysis of the users behaviours and the effect that contextual variables have in them. From a commercial view, this can also help a company to obtain a better understanding of their current user base so that they can offer services specially focused for them.

As flexible as this approach is, it is clear that this method does not recommend items in particular. The output are the categories that the user may enjoy. To deal with this issue, we use a different algorithm as an external filter.

5 Getting a Specific Item Recommendation

We follow two paths to get a specific item after the first phase of our recommendation process. We look for the item's characteristics and apply one of the following approaches:

- Rules - This are rules defined by the users that help the system personalize their results. Let's say our system knows that our user likes to spent winter vacations on the beach. This would lead our first phase to suggest such environment. However, to decide between Miami and Acapulco, we need more information. If the user sets that he likes to travel to a foreign country during vacation, then we have a new filter that will help us decide between our possible items. Of course this has the limitation that for being useful, it needs to have direct feedback from our user. This is were our second approach comes into play.
- Collaborative filtering - Collaborative filtering are a group of algorithms that can predict the score a user would give to a certain item based on the scores that similar users gave to similar or the same item. By looking at this, it alleviates our previous problem. We already have the score information for some items given by some users. We now need to apply the collaborative filtering to the set of items that meet the characteristics defined by our first phase. This helps us to reduce the search space and get a personalized result.

An advantage of dividing our approach in two phases is that we can work on both of them to either improve the predictions accuracy, or to get more information about the elements involved in the recommendation process.

In general, our model can be seen as shown in Figure 3. We first aquire our user's preferences and the context in which he is placed. Then a Bayesian network is used to determine the best characteristics of the items that can be recommended. Last, we use a filtering algorithm to determine the best match on the items that meet the same characteristics defined by our Bayesian network.

6 Conclusions

We presented a method based on Bayesian networks to represent and handle the different variables that can interact and affect the user's decision when looking for

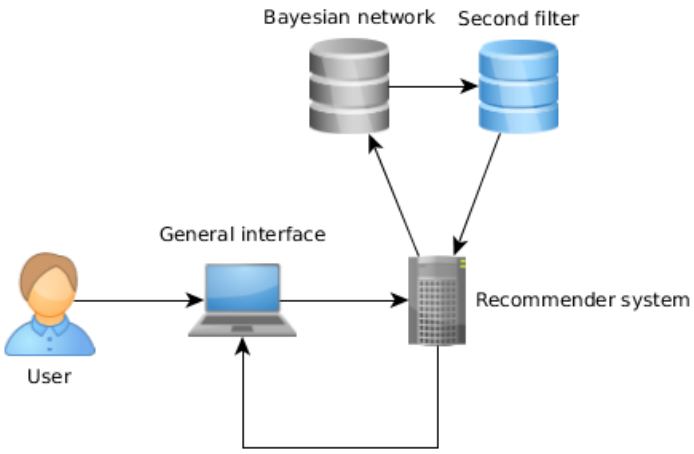


Fig. 3. General structure of our model.

a recommendation. We finish the recommendation process by using traditional methos such as collaborative filtering.

While the proposed approach can be used with datasets without contextual data, it will lose most of its benefits compared to other approaches.

The predictions about the user's preferences given a context are done directly over the Bayesian network. The query delivers the best options while still considering the selected values for the request. We can include as many constraints as we want or as data we have available at the moment.

Another advantage of our approach is that we can evaluate queries that do not necessarily involve the users' scores. We can learn more from their interactions, so that we can infer and even predict what they will do or buy giving the current context (or an hyphothetical one).

Finally, another advantage of having a visual method such as a Bayesian network as the base of the model is that it can be presented to the final user to let him do some adjustments if he does not feel satisfied by the current state of his model.

We also acknowledge that recommender systems are only helpers in the user's decision processs. Some users may want to have a more proactive role in the process, such as Van Setten *et al.* [14] expose, some users like to think and decide for themselves. For this reason we intended to give the user more control with feedback process. This way, the user is certain that his own beliefs are really considered and that the system covers his needs.

The work presented in this paper can be further improved and explored. We consider this as just a first step into better understanding the relationships between the available data inside a recommender system. Acknowledging this, we

would like to present the following points as possible steps into further developing this approach:

- Complete the vacation dataset - While we have some data collected from our suveys. Our dataset can be improved by getting more users and by including follow-up questions in order to test the evolution of our model.
- Adding more than one domain - We focused on a single domain for this paper. This is a common approach on recommender system's research since some domains might not share the same preferences. Finding the preferences and variables that can be shared might prove useful to build the models used for the predictions.
- To further test our personalization approach, a better tracking of specific users along an expanded time window would allow us to improve the system's models and feedback measures. This could also lead to conclude good parameters that indicate the frecuency in which the network should be relearned. This could also mean that the base case could be altered depending on the system's evolution.

The future work is not necessarily limited to the subjects we mentioned. We consider them as the natural progression from our work and some viable points to improve our current results.

The main concern in this type of approaches is the impending need for contextual datasets that can capture the detailed information of the interactions between users and items. Since we are representing the interaction, we can also evaluate the satisfaction that it meant to the user, not limiting our approach only to the items in the dataset.

References

1. Adomavicius, G., Tuzhilin, A.: Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE transactions on knowledge and data engineering 17(6), 734–749 (2005)
2. Adomavicius, G., Tuzhilin, A.: Context-aware recommender systems. In: Recommender systems handbook, pp. 217–253. Springer (2011)
3. Baltrunas, L., Ludwig, B., Peer, S., Ricci, F.: Context relevance assessment and exploitation in mobile recommender systems. Personal Ubiquitous Comput. 16(5), 507–526 (Jun 2012), http://dx.doi.org/10.1007/s00779-011-0417-x
4. Ben-Gal, I.: Bayesian networks. Encyclopedia of statistics in quality and reliability (2007)
5. Fischer, G.: User modeling in human–computer interaction. User modeling and user-adapted interaction 11(1-2), 65–86 (2001)
6. Herlocker, J.L., Konstan, J.A., Terveen, L.G., Riedl, J.T.: Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS) 22(1), 5–53 (2004)
7. Košir, A., Odic, A., Kunaver, M., Tkalcic, M., Tasic, J.F.: Database for contextual personalization. Elektrotehniški vestnik 78(5), 270–274 (2011)

8. Liu, H., Hu, Z., Mian, A., Tian, H., Zhu, X.: A new user similarity model to improve the accuracy of collaborative filtering. Knowledge-Based Systems 56, 156–166 (2014)
9. Nilashi, M., bin Ibrahim, O., Ithnin, N.: Multi-criteria collaborative filtering with high accuracy using higher order singular value decomposition and neuro-fuzzy system. Knowledge-Based Systems 60, 82–101 (2014)
10. Ono, C., Kurokawa, M., Motomura, Y., Asoh, H.: A context-aware movie preference model using a bayesian network for recommendation and promotion. In: User Modeling 2007, pp. 247–257. Springer (2007)
11. Panniello, U., Tuzhilin, A., Gorgoglione, M.: Comparing context-aware recommender systems in terms of accuracy and diversity: which contextual modeling, pre-filtering and post-filtering methods perform the best (2012)
12. Ramezani, M., Moradi, P., Akhlaghian, F.: A pattern mining approach to enhance the accuracy of collaborative filtering in sparse data domains. Physica A: Statistical Mechanics and its Applications 408, 72–84 (2014)
13. Schafer, J.B., Frankowski, D., Herlocker, J., Sen, S.: Collaborative filtering recommender systems. In: The adaptive web, pp. 291–324. Springer (2007)
14. Van Setten, M., Pokraev, S., Koolwaaij, J.: Context-aware recommendations in the mobile tourist application compass. In: International Conference on Adaptive Hypermedia and Adaptive Web-Based Systems. pp. 235–244. Springer (2004)

ISSN 1870-4069

Intelligent Geographic Information System for Decision Making in the Electricity Sector

Nayi Sanchez Fleitas[1], María Matilde García Lorenzo[2], Raúl Comas Rodríguez [3], Amanda Riverol Quesada[4]

[1] Empresa de Tecnología de la Información y la Automática, Cuba

[2] Universidad Central Martha Abreu de Las Villas, Computer Science Department, Cuba

[3] Universidad Regional Autónoma de los Andes, Research Directorate, Ecuador

[4] Universidad de Cienfuegos Cuba

raulcomasrodriguez@gmail.com

Abstract. In the Electricity Union works on the development of a Geographic Information System (GIS) that has a conceptual basis and response to the different requests of the user. For this, the objective of the research is: to develop a model for the management of geospatial data, with the use of Artificial Intelligence techniques, as support to decision making in the electric sector. As a first step a light ontology is elaborated that endows the conceptual base system. To achieve the automatic queries a Case Based System is developed. The case database contains the description of static queries previously made in the form of cases and responds to a three-level hierarchical organization, which favors the processes of access, recovery and learning of cases. Each consultation consists of eleven fundamental features, of which eight are predictors and three objectives. This system is applied in all the electric companies of the country. The results of field validation evidence the feasibility of the proposal.

Keywords: electricity, GIS, case-based system, ontologies.

1 Introduction

Energy is one of the pillars of development in productive processes, the social progress, the technological advance [1], the satisfaction of the needs of people and a means to

raise the standard of living of the population. The obtaining of electric power calls for a complex infrastructure.

The Electrical Union (UNE, Spanish acronym), in Cuba develops the Business Management System of the Electrical Union (SIGE), that focuses on the automation of electrical processes [2]. SIGE is composed of two main subsystems: The Integral System of Network Management (SIGERE) and the Integral Management System of the Electrical Industry Construction Enterprise (SIGECIE).

The functions of SIGERE and SIGECIE are to collect technical, economic and management data to convert them into information. The data collected facilitate and improve the efficiency in the operation, use, analysis, planning and management of the electricity distribution and transmission networks. The two systems constitute the databases of a Geographic Information System (GIS), that forms part of the SIGE.

The SIG's development begins in the 2001. The first version receives SIGOBE's name 1.0, with 220 options of quest. For the SIG's bringing up to date is join panel of experts with experience of the theme meets. An analysis of the literature is carried out and a group of experts on the subject is gathered, who detect the following limitations:

1. Rudimentary methods of n elaboration.
2. Functional relationships of the electrical system elements are not described in the database.
3. Lack of important concepts for the electrical system in the data base.

Given current information needs, interest in GIS has increased [3]and techniques and standards have been developed to ensure that everyone has access to this type of data[4].Therefore the objective of the investigation is: Developing a model for data management , with the use of techniques of artificial intelligence, on a deep conceptual scheme of dominion, that answer open into the petitions of users like support to the overtaking in the Union Electrical.

2 Paper Preparation

The development of geographic information systems is generally done focusing on a particular context and restricted to a specific domain that generates problems of interpretation. Different communities can define the same objects in different ways, according to different points of view and assumptions about the study domain, which causes communication problems due to lack of shared understanding [5]. One of the most important problems detected in the different investigations conducted are those derived from the heterogeneity and interoperability of the data [6]. For its solution, an increasingly dominant strategy in the organization of information is associated with the term "ontology" [7]. Neches argues that an ontology defines the basic terms and relationships that comprise the vocabulary of an area of interest, in addition to the rules of combining terms and relationships to extend that vocabulary [8].

The fundamental role of ontology is to structure and retrieve knowledge, to promote its exchange, and to promote communication [9].

Table 1. Representation of the network in vectorial scheme.

Points (Support Points)	**Lines (Circuits)**	**Polygons (Substations)**
Posts	Transmission Circuit	Distribution Substation
Transformer banks	Subtransmission Circuit	Transmission Substation
Capacitor banks	Primary Circuit	
Generator groups	Secondary Circuit	
Disconnectors Structures	Lighting Circuit	
Lamps		

In addition, relying solely on CBR for distributed and complex applications can lead to systems being ineffective in knowledge acquisition and indexing [10]. According to [11] use of ontologies in case-based reasoning gives the following benefits: It is an easy-to-use tool for case representation, queries are defined using daily terminology , it facilitates the assessment of similarity and it increases system performance.

Alight weight ontology is provided to the system to give a conceptual basis. In the conceptualization we have the concepts, their taxonomy and relations (objects properties); the remaining components of the ontology model (data properties, instances and axioms) are not developed, because the information is already in the database that feeds the system. The ontology was carried out using the Methontology methodology and the Protégé tool [12].

With the ontology the system is given a conceptual basis. But we still have a problem a weakness of the systems proposed is in the dissatisfaction with the queries carried out. If a static inquiry is developed for each problem that arises, the database begins to store a group of scarcely-used queries. In order to solve the problem, the system must be able to generate intelligent queries in real time, in which the knowledge obtained from previous ones is used.

Artificial Intelligence (AI), is the branch of computer science that attempts to reproduce the processes of human intelligence through the use of computers [13]. Within AI, the Expert Systems (ES), or Knowledge Based Systems (KBS), emerged in the 1970s as a field that deals with the study of: the knowledge acquisition, its representation and the generation of inferences about that knowledge.

In this sense, CBRs appear as a palliative to the process of knowledge engineering and are based on the premise that similar previous problems will have similar solutions [14].

With this principle as the basis, the solution to a problem is retrieved from a memory of solved examples. For each case, the most similar previous experiences that allow finding the new solutions are taken into account [15, 16]. Case-based reasoning contributes to progressive learning, so that the domain does not need to be fully represented [17]. The CBRs have three main components: a user interface, a knowledge base and an inference engine [18].

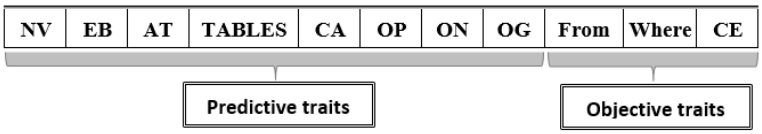


Fig. 1. Cases structure.

An analysis of the inquiries carried out, including those for SIGERE, allows establishing the structure of a case to solve the problem, which is divided into predictive traits and objective traits (Figure 1). For a better understanding of the proposed structure, in the Table 1 identifies the universe of discourse of the predictive and objective traits.

SIGERE and SIGECIE are considered complex systems that have 36 modules in use, and a database of: 716 tables, 1303 procedures stored and 74 functions. In addition, other functionalities are in development phase. An action on transformers involves approximately 9 tables that have more of 150 different attributes. These actions are represented in the case base only for 6 queries.

The NV trait determines the database to reference (SIGERE or SIGECIE), deletes a group of installations and sets the final scale at which the results are displayed. For example, the transmission process involves several provinces, subtransmission several municipalities of a province, while the rest of the levels are located within a single inhabited place.

The EB trait represents the classification of the basic elements of the electric network according to their correspondence of their characteristics in the vectorial scheme. Table 2 shows the representation of the basic elements according to their characteristics in points, lines and polygons.

The ontological traits ON and OG, represented by descriptive logics, have the greatest weight within the case. A possible value of the ON trait would be: T ∩ TPot ∩ TMonofasic -- SSecondary This range expresses that the element is a monophasic primary transformer with no secondary output. OG works similarly, but their relationship is spatial.

In the present research, a hierarchical structure is used because it favors the system in the process of access and retrieval to the most similar examples to the real time query. For this, an analysis of the traits is performed taking into account which allows discriminating more options in each case. In the Figure 2 shows the organization of the case database where:

1. the predictor trait NV is the root node;
2. in the second and third level the EB and OP traits are added, respectively, since they are the most discriminative elements;
3. in the leaf nodes, sub-sets of cases that represent those examples where the value of NV, EB, OP match.

Table 2. Universe of discourse of predictive and objective traits.

Trait	Possible values	Type
	Predictive traits	
NV	Secondary, Primary, Subtransmission, Transmission	Symbolic and single-valued
EB	Posts, Transformer banks, Capacitor banks, Generator groups, Disconnectors, Structures, Lamps, Transmission Circuit, Subtransmission Circuit, Primary Circuit, Secondary Circuit, Lighting Circuit, Distribution Substation, Transmission Substation	Symbolic and single-valued
AT	Attributes to be returned by the inquiry (code, voltage, name, etc ...)	Set
Tables	Tables of the SIGERE involved in the inquiry (Accessories, Actions, Connection, Interruptions, Line, CurrentSupplyPrimary, etc ...).	Set
CA	Element to compare (Attribute being compared)	Symbolic and single-valued
ON	Operator ($\cup$, $\cap$, $\leq$ $\geq$ $=$, like , etc)	Symbolic and single-valued
OP	Ontology (descriptive logic) $T \cap TP \cap TMonophase \neg SSecondary$	Ontology
OG	Spatial constraint (descriptive logic)	Ontology
	Objective traits	
From	Returns the From of the inquiry to the SIGERE	String
Where	Returns the Where of the inquiry to the SIGERE	String
CE	Returns the GIS inquiry	String

2.1 Inference Engine

The recovery module is responsible for extracting from the case database the case or cases most similar to the current situation.

Global similarity is the result of the weighted sum of the distances between the value of each trait in a case and the value that it acquires in the problem case. This similarity is determined by equation 11. The distances are weighted considering the expert criterion. The most important traits are the ontological ones:

$$\mathrm{SimGlobal}(bc_j, P) = \sum_{i=0}^{8} w_i * d_i(bc_{ji}, P_i) \Big/ n. \quad (1)$$

Table 3. Results of SICUNE by area.

Queries	Total	Correctly	Bad	Retain	%
Office	230	221	9	23	96,08%
Engineering department	189	178	11	40	94,18%
Customer service area	147	140	7	18	95,23 %

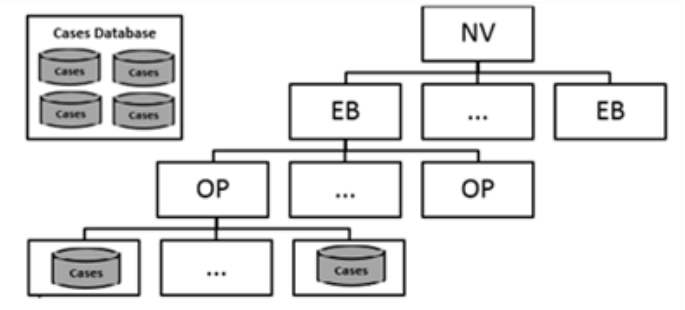


Fig. 2. Cases database structure for the UNE.

Local distance is determined by the type of data of the traits. In the case presented here there are three types of data for which different distance measures are used that are described below.

The traits NV, EB, CA and OP are symbolic single-valued type, the distance used is of Boolean type. The AT and Tables traits are of the set type. As a result of the tests performed on the system, the Jaccard distance is implemented.

The OG and OE traits represent the general and spatial ontologies. The similarity measure that give the best results based on a control sample and a field test was the Jaro-Winkler distance.

$$d_i(bc_{ji}, P_i) = \begin{cases} d_i(bc_{ji}, P_i) = \begin{cases} 0 \text{ Si } bc_{ji} == P_i \\ 1 \text{ Si } bc_{ji} \neq P_i \end{cases} \quad i = 1,2,5,6, \\ d_i(bc_{ji}, P_i) = \frac{|bc_{ji} \cap P_i|}{|bc_{ji} \cup P_i|} \; i = 3,4, \\ d_i(bc_{ji}, P_i) = \frac{1}{3}\left(\frac{c}{|bc_{ji}|} + \frac{c}{|P_i|} + \frac{c - t/2}{c}\right) \; i = 7,8. \end{cases} \quad (2)$$

3 Results

SIGOBE is national in character and is applicable to the different areas that divide electricity companies. It offers a group of facilities such as:

1. Locate complaints from the population, a failed installation or with abnormal parameters,
2. organize the tour of the cars, display customer voltages on the map,
3. make a study of equipment faults in rural areas,
4. optimize the use of networks and optimize their use,
5. at certain scales allows drawing the sketch of new projects with the necessary accuracy.

With the development of the ontology, it achieves greater efficiency in the software. For the implementation of the RBC was developed within SIGOBE the module SICUNE (Intelligent System of Consultation for the UNE); Which uses the transformational analogy on previously made queries, retrieved by a case-based reasoned, to answer user questions.

The stage of retention of cases is in preliminary phase since the current size of the base of cases does not presuppose reissues of the cases because it still results of average size. New cases are needed as new SIGERE modules are developed.

4 Experimental Study

To test the SICUNE, three departments of an Electric Company are selected that use information from different areas of the database and achieve greater coverage in the information contained. The work of these areas is operational and needs the functionality proposed for their daily work:

1. Office,
2. Engineering department,
3. Customer service area.

In all three areas, SICUNE tests are carried out for a period of one month for its validation. Table 3, shows the results by area.

The general results of the application of the SICUNE, with an effectiveness of 95.23%. In order to strengthen SICUNE, it is necessary to incorporate new cases, especially those related to the engineering area, due to the complexity of its work.

5 Conclusions

The following conclusions can be made:

1. A case-based system on type problem solver was designed, using as an initial case database, the 265 static queries registered in SIGERE. The queries are described by eight data-type predictive traits and three objective traits. The case database responds to a three-level hierarchical organization, which favors the processes of access, recovery and learning of cases.
2. The similarity between two cases was determined by the weighted sum of the distance of their traits.

3. In the research was developed ontology for the processes of distribution and transmission of electrical energy.
4. The case retention stage is in preliminary phase, since the current size of the case database does not presuppose reissues of cases, because it is still medium-sized.
5. An intelligent real-time queries system is implemented for the UNE (SICUNE), achieving the generation of automatic queries that allow the system to respond to any type of queries in real time.
6. The experimental study shows the feasibility of the proposal.

References

1. Castillo, Y., Gutiérrez, M. C., Vanegas-Chamorro, M., Valencia, G., Villicaña, E.: Rol de las Fuentes No Convencionales de Energía en el sector eléctrico colombiano. Prospectiva, 13, pp. 39–51 (2015)
2. Fernández-Álvarez, R.: Informatización de la Gestión de las Redes Eléctricas. Tesis Doctoral, Facultad de Ingeniería Eléctrica (2011)
3. Yusoff, A., Abdullah, S., Din, N. M.: A Taxonomy on Knowledge-based Geographical Information System (GIS) for a Cloud-based Disaster Management Environment. The 3rd National Graduate Conference (NatGrad) (2015)
4. Carrera-Calderón, F. A., Martínez-Vargas, M. Á.: La Accesibilidad Web en el Proyecto de Infraestructura de Datos Espaciales – GTIDE CEDIA, UNIANDES EPISTEME. Revista de Ciencia, Tecnología e Innovación, 3 (2016)
5. Durango-Vanegas, C. E.: Asociación de datos espacio-temporales en bases de datos Oracle, Ingenierías USBmed, 5, pp. 100–108 (2015)
6. Machado-García, N., González-Ruiz, L., Balmaseda-Espinosa, C.: Recuperación de objetos geoespaciales utilizando medidas de similitud semántica. Revista Cubana de Ciencias Informáticas, 8, pp. 132–143 (2014)
7. Smiraglia, R. P.: Domain Analysis for Knowledge Organization Tools for Ontology Extraction. Elsevier (2015)
8. Neches, R., Fikes, R. E., Finin, T., Gruber, T. R., Senator, T., Swartout, W. R.: Enabling technology for knowlege sharing. AI Magazine, 12, pp. 36–56 (1991)
9. Myrgioti, E., Bassiliades, N., Miliou, A.: Bridging the HASM: An OWL ontology for modeling the information pathways in haptic interfaces software. Expert systems with applications, 40, pp. 1358–1371 (2013)
10. Akmal, S., Shih, L. H., Batres, R.: Ontology-based similarity for product information retrieval. Computers in Industry, 65, pp. 91–107 (2014)
11. Bouhana, A., Zidi, A., Fekih, A., Chabchoub, H., Abed, M.: An ontology-based CBR approach for personalized itinerary search systems for sustainable urban freight transport. Expert systems with applications, 42, pp. 3724–3741 (2015)
12. Sánchez-Fleitas, N., Comas-Rodríguez, R., García-Lorenzo, M. M., Riverol-Quesada, A.: Modelo de manejo de datos, con el uso de inteligencia artificial, para un sistema de información geográfica en el sector energético. Enfoque UTE, 7, pp. 95–109 (2016)
13. Russell, S. J., Norvig, P.: Artificial Intelligence. A Modern Approach. Englewood Cliffs (1995)

14. Gillespie, K., Gupta, K. M., Drinkwater, M.: Case-Based Object Placement Planning. International Conference on Case-Based Reasoning, pp. 170–184 (2014)
15. Moreno-Laverde, R., Joyanes-Aguilar, L., Giraldo-Marín, L. M., Duque-Méndez, N. D., Tabares-Morales, V.: Modelo para personalización de actividades educativas aprovechando la técnica de Razonamiento basado en Casos (RbC), 4, pp. 118–127 (2015)
16. Wimmer, H., Rada, R.: Good versus bad knowledge: Ontology guided evolutionary algorithms. Expert systems with applications, 42, pp. 8039–8051 (2015)
17. Fitzgerald, T., McGreggor, K., Akgun, B., Thomaz, A., Goel, A.: Visual Case Retrieval for Interpreting Skill Demonstrations. 23rd International Conference, (ICCBR) (2015)
18. Cordero-Morales, D., Ruiz-Constanten, Y., Torres-Rubio, Y.: Sistema de Razonamiento Basado en Casos para la identificación de riesgos de software. Revista Cubana de Ciencias Informáticas, 7, pp. 222–239 (2013).

ISSN 1870-4069

Life Time Cycle in Power Electronics for Fuzzy Logic Speed Controller in Brushless Motors

Manuel García-López[1], José A. Rosales-Martinez[2], Pedro Ponce-Cruz[2], Arturo Molina-Gutiérrez[2], José J. Rodríguez Rivas[1]

[1] Instituto Politécnico Nacional, Mexico

[2] Tecnológico de Monterrey, Ciudad de México, Mexico

{magarcial, jjrodriguezr}@ipn.mx, {antonio.rosales, pedro.ponce, armolina}@itesm.mx

Abstract. Power electronic modules are subject to a variety of temperature profiles caused by cyclic thermo-mechanical stress on joints causing device failure. This paper shows how the life time cycle of the power electronic converter is affected in a fuzzy speed controller for a brushless DC motor drives (BLDC). Although there are several papers that show the performance of fuzzy logic speed controllers, there are not enough papers that show how the life time in power electronics is affected according to the power electronic stage losses. Thus, the losses in the power electronic stage are not included into the design of the controller. As a result, the industrial process in which brushless motor drives are installed, decrement their life time. This paper estimates the lifetime cycle in the power electronics stage when a fuzzy logic speed controller is implemented in BLDC. The analysis is based on co-simulation between LabVIEW and Multisim to calculate the response, stablished by the fuzzy logic controller, of temperature because to power losses conduction/switching of the semiconductor (IGBTs), on power electronic stage and the speed control. A comparison between a conventional linear controller and a fuzzy logic controller also is conducted and the results show a superior response in the speed response. In addition, the losses in the power electronic stage are reduced when the fuzzy logic speed controller is implemented. Hence, the fuzzy logic design is evaluated according to the speed response and lifetime in the power electronic stage.

Keywords: power electronic, fuzzy logic controller, life time cycle, brushless DC motor.

1 Introduction

Brushless DC motors (BLDCM), are widely used in the industry since they provide advantageous properties as large lifetime, noise robustness, good enough torque-

velocity ratio, smaller size and higher efficiency than inductions motors, and rotor losses in BLDCM are minor compared with those in induction machines [1, 2]. Position and speed regulation and/or tracking are commonly the control objectives for a BLDCM i.e. position control is employed in space crafts and servomechanism while speed control is important in robotics, [3, 4]. However, the power electronics (power converter), essential to drive the BLDCM and accomplish the control objectives, is not studied or analyzed deeply even this is neglected in most of the cases.

Commutation frequency of power electronics, supply voltage, and tracking commands, are vital conditions for power electronics in BLDCM control applications [5]. These conditions generate a variety of operation modes producing temperature profiles causing cyclic thermo-mechanical stress on power electronic devices [6, 8]. The lifetime expectations of power electronic devices, is defined in terms of the frequency and magnitude of the thermo-stress cycles [6,8]. The consideration of these operation modes in terms of thermal stress on the design of control strategies, can extend hugely the lifetime of the power electronic devices having direct effects in economic savings [5, 6, 7].

Discrete power semiconductors are applied on different areas depending of power requirements and operation frequency. For example, SCRs and GTOs are employed on high-power applications while IGBTs and MOSFET are used on high-frequency applications. Then, considering the are application and control requirements, a variety of power semiconductors are available to be utilized, see [12].

Linear and nonlinear controllers have been designed to ensure the control goals despite disturbances and uncertainties demanding a high performance to power electronics guarantying accuracy, precision, and robustness, in return for a shorter lifetime of power electronics [7]. In this paper, a Fuzzy logic control to regulate the speed on a BLDCM, considering thermo-stress to estimate lifespan of power electronics (h-bridge), is presented. Main advantages of Fuzzy control are: facility to control nonlinear systems without mathematical model; work as adaptive control, commonly Fuzzy techniques are combined with classical PID and nonlinear controller techniques to adapt controller's gains; potential to work as multi-input controllers; and the most important is the capability to encapsulate information from operators adding human knowledge and experience in the performance of the controller [9].

In this research work, the analysis and estimation of lifetime in power electronics for BLDCM speed control is validated via co-simulation LabView™-Multisim™. Co-simulation is understood as the integration of two simulation platforms to form an integrated system where a constant exchange of data between both simulations platforms is maintained. For emulation of BLDCM speed control, co-simulation LabView™-Multisim™ is a valuable option since Multisim™ contain accurate models of DC drives and power electronics (including thermal analysis), while LabView™ is an useful software for control systems simulation [10]. The contribution of this paper is the design of a fuzzy control to regulate the speed of a BLDCM, dealing with temperature profiles caused by desired tracking commands, lifetime of power electronics is estimated using the temperature profiles produced by the control specifications, the temperature profiles are obtained from the co-simulation LabView™-Multisim™.

It is highly recommended that a fuzzy logic controller is designed the speed response has to be achieved but also the thermal response of the power electronics stage has to be analyzed during the design stage. Moreover, it is required that a co-simulation validates the thermal losses in the power electronics stage. If thermal losses are not studied, the lifetime of the electric drive is reduced.

2 Brushless Motor and Converter

The dynamics of BLDCM are described by equation (1):

$$\dot{x} = \begin{bmatrix} -\frac{R_s}{L_1} & 0 & 0 & -\frac{\lambda_p}{J} f_{as(\theta_r)} & 0 \\ 0 & -\frac{R_s}{L_1} & 0 & -\frac{\lambda_p}{J} f_{bs(\theta_r)} & 0 \\ 0 & 0 & -\frac{R_s}{L_1} & -\frac{\lambda_p}{J} f_{cs(\theta_r)} & 0 \\ \frac{\lambda_p}{J} f_{as(\theta_r)} & \frac{\lambda_p}{J} f_{bs(\theta_r)} & \frac{\lambda_p}{J} f_{cs(\theta_r)} & -\frac{B}{J} & 0 \\ 0 & 0 & 0 & \frac{P}{2} & 0 \end{bmatrix} \begin{bmatrix} i_{as} \\ i_{bs} \\ i_{cs} \\ \omega_m \\ \theta_r \end{bmatrix} + \begin{bmatrix} \frac{1}{L_1} & 0 & 0 & 0 \\ 0 & \frac{1}{L_1} & 0 & 0 \\ 0 & 0 & \frac{1}{L_1} & 0 \\ 0 & 0 & 0 & \frac{1}{J} \\ 0 & 0 & 0 & 0 \end{bmatrix} u, \quad (1)$$

where $u = [v_{as} \quad v_{bs} \quad c_{cs} \quad T_l]^T$.

The movement of BLDCM is driven by the current applied on the stator coils that generates a magnetic field orthogonal in the direction of the magnetic field produced by the permanent magnets. Once the movement begins, the rotor position changes constantly, and the position of the rotor must be measured to excite the corresponding coil to the permanent magnet [10,11]. A common circuit to drive BLDCM´s (composed by voltage source, 3-phase inverter, and three Hall Effect sensors) is the presented in Fig. 1.

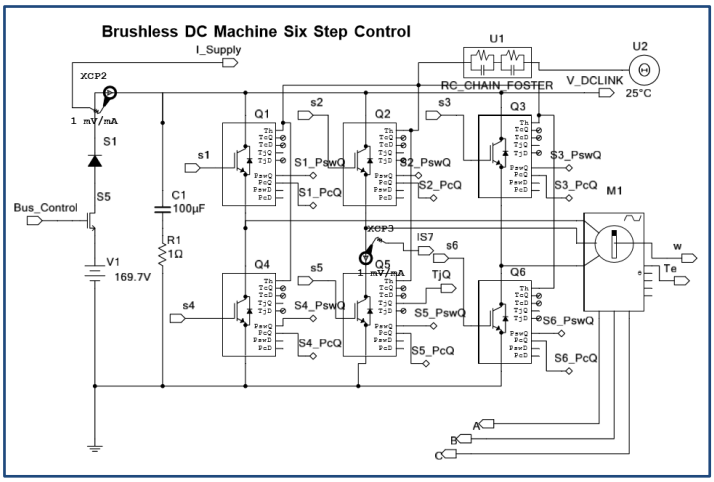


Fig. 1. BLDCM drive co-simulation in Multisim ™.

1.1 Semiconductor Losses and Lifetime

Power losses in IGBT module consists of the sum of power losses of IGBT chips and FWD chips. Power losses can be classified as either on-state losses or switching losses. The *on-state power losses of the IGBT* are calculated by $P_{CT} = u_{CEO} * I_{Cav} + r_c I^2_{Crms}$and can be estimated using the output characteristic shown in Fig. 2(a). The *diode on-state power* losses are calculated by $P_{CD} = u_{DO} * I_{Dav} + r_D I^2_{Drms}$ and can be estimate using the output characteristic shown in Fig. 2(b). *The turn-on energy losses in IGBT* are calculated by $E_{onT} = \int_{ton}(u_{CEO} * i_C(t))dt$, the turn-off energy losses by $E_{offT} = \int_{toff}(u_{CEO} * i_C(t))dt$ and the turn-on energy in the diode mostly of the reverse-recovery energy is calculate by $E_{onD} = \int_{ton}(u_D(t) * i_F(t))dt$.

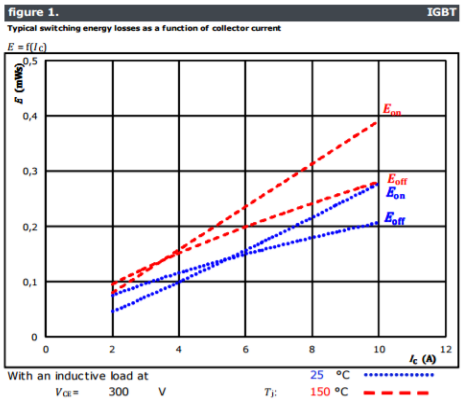
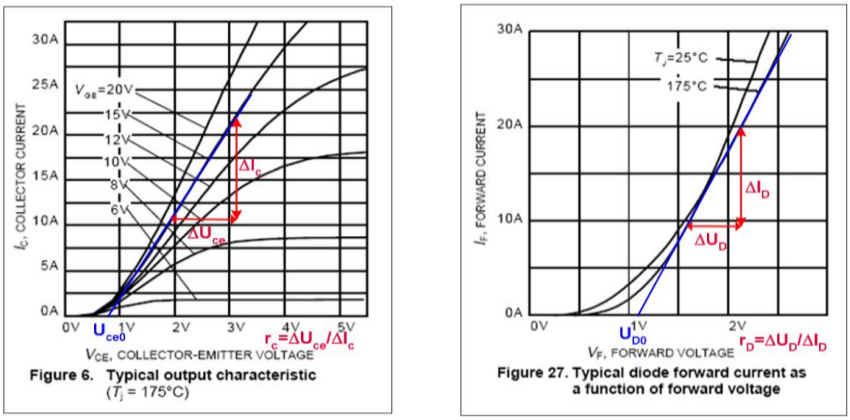


Fig. 3. Typical switching energy losses as a function of collector current.



a) b)

Fig. 2. (a) Reading the u_{CEO} and r_c $(r_c = \Delta U_{ce}/\Delta I_c)$ from the data-sheet diagram, **(b)** Reading the u_{DO} and r_D $(r_d = \Delta U_{ce}/\Delta I_c)$ from the data-sheet diagram:

The switch-off losses in the diode are normally neglected $E_{offD} \approx 0$. The switching losses in the IGBT are calculated by $P_{swT} = \left(E_{onT} + E_{offT}\right) * F_{sw}$ and in the diode are the product of switching energies and the switching frequency (F_{sw}) and are calculated by $P_{swD} = \left(E_{onD} + E_{offD}\right) * (F_{sw}) \approx E_{onD} * (F_{sw})$. The losses total are calculated by equation (2):

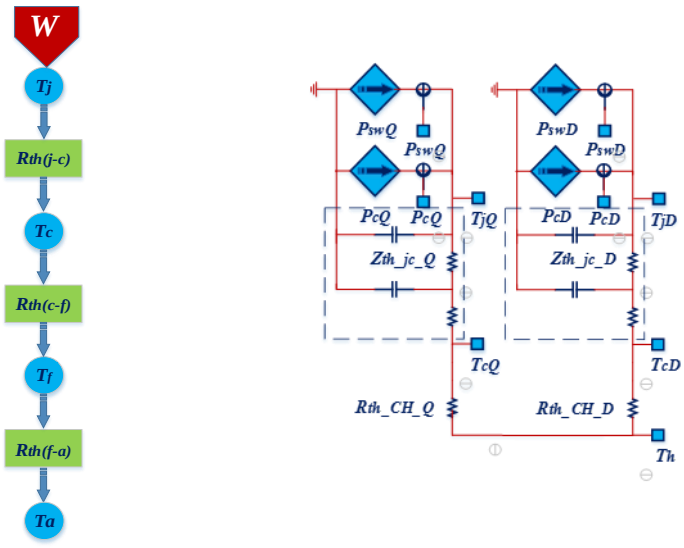
$$W = P_{CT} + P_{swT} + P_{CD} + P_{swD}. \tag{2}$$

The switching losses in the IGBT can be calculated from switching vs. collector current characteristics as it is shown the Fig. 3. The diode switching losses can be calculated in analogous manner [15].

The *heat conduction of a semiconductor* can be simulated in an electric circuit with one IGBT module mounted on a heat sink this circuit is shown in Fig. 4(a). Using the equivalent circuit of Fig. 4(a), where W is module power loss, T_j is junction temperature IGBT chip, T_f is temperature of heat sink, T_c is module case temperature, T_a is ambient temperature, $R_{th(j-c)}$ is thermal resistance between case and heat sink, $R_{th(c-f)}$ is contact thermal resistance between case and heat sink, and $R_{th(f-a)}$ is thermal resistance between heat sink and ambient air.

The junction temperature T_j can be calculated using the thermal equation (3) [15]:

$$T_j = W * \left(Rth(j-c) + Rth(c-f) + Rth(f-a)\right) + Ta. \tag{3}$$



a) b)

Fig. 4. (a) Thermal resistance equivalent circuit, (b) Electro-Thermal model.

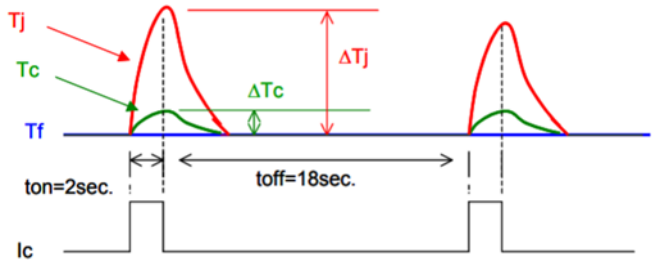


Fig. 5. Pattern diagram of current flow of ΔT_j power cycle and temperature change.

Modeling of commutation and conduction losses as well as the temperature profiles on IGBT junction is done by electro-thermic networks considering averages on a modulation period [14], see Fig. 4(b).

The *power cycle life* can be calculated from the power cycle capability curve that shows the relation between the temperature change ΔT_j and the number cycles. The power cycle capability curve is the life curve made when the junction temperature rapid rises and falls.

In these curves, failure caused by deterioration at the interface between the aluminum bonding wire and chip surface interconnection is observed. Fig. 5. shows the patterns of current flow in the ΔT_j power cycle test. During the ΔT_j power cycle test the junction temperature goes up and down in a short time cycle; therefore, outstanding temperature difference between silicon and bonding wire results in thermal stress. The ΔT_j power cycle lifetime is mainly limited by the aluminum bonding wire joints. Fig. 13(a) shows the power cycle capability curve of the IGBT module to $\Delta T_{jmin} = 25℃$ and to $\Delta T_{jmax} = 150℃$ [15].

The power cycles capability can be calculated by equation (4):

$$CF = 541162959016419 * \Delta T^{-5.12121}, \tag{4}$$

where $\Delta T = P_t Z_{th}$, $P_t = I_{rms}^2 R_{on}$, and $Z_{th} = 2.3354 F_r^{-0.165}$. The Time Before Failure (TBF) can be calculate by equation (5):

$$TBF = \frac{CF}{Fr} * 60 * 24 * 265\ years. \tag{5}$$

where F_r is the frequency of the thermal oscillations.

3 Fuzzy Logic Speed Control Design

Fuzzy control is a qualitative control strategy emulating expert knowledge. In order to satisfy the control objective, the control law in equation (6) is applied:

$$u = \sum_{i=1}^{4}\sum_{j=1}^{4}\sum_{k=1}^{4} v_i\, w_j q_k \big(a_{0ijk} + a_{2ijk}e_I + a_{3ijk}e_D + a_{4ijk}e_P e_I + a_{1ijk}e_P + a_{5ijk}e_P e_D + a_{6ijk}e_I e_D + a_{7ijk}e_P e_I e_D\big), \tag{6}$$

where the logic variables are v_i, ω_j, q_k contain the information of the input membership functions with, $v_1 = \begin{cases} 1 \text{ for } e_P \le e_{P1} \\ 0 \quad \text{otherwise} \end{cases}; v_2 = \begin{cases} 1 \text{ for } e_{PI} < e_P \le e_{P2} \\ 0 \quad \text{otherwise} \end{cases}; v_3 = \begin{cases} 1 \text{ for } e_{P2} < e_P \le e_{P3} \\ 0 \quad \text{otherwise} \end{cases}; v_4 = \begin{cases} 1 \text{ for } e_{P3} \le e_P \\ 0 \quad \text{otherwise} \end{cases}$. other variables can be computed similarly as w_j, q_k.

A useful methodology to design fuzzy logic controllers is based on a linguistic phase plane [13] (see Fig. 6). Computing a fuzzy control using the linguistic plane in Fig. 6, with the rules presented in Table 1, it is possible design a controller with small/big rise time as well as small/big overshoot. This design methodology can be applied to design a robust and fast controller decreasing lifespan of power electronics or a moderate control effort in order to extending the lifetime.

Linguistic phase plane has the capability to moderate the control effort and to increase the lifetime of power electronics in BLDCM speed control. Control engineers and researchers should take into account this methodology to add interesting characteristics to their controllers, not given in terms of control performance, but in terms of economic savings, as well as a large lifetime in power electronic.

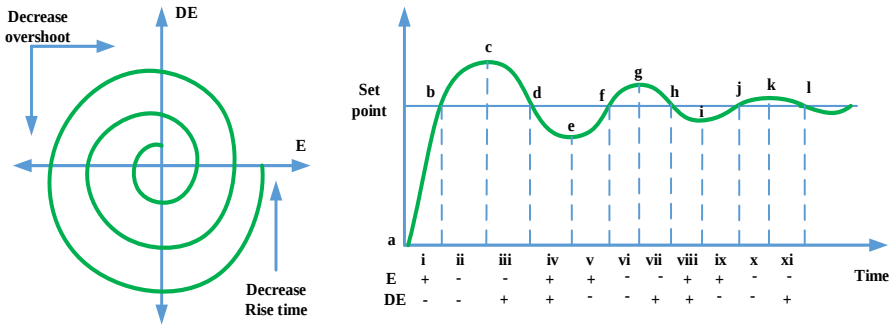


Fig. 6. Linguistic phase plane.

Table 1. Fuzzy control rules for Fig. 6 (N: Negative, Z:Zero, P:Positive, sp:set point, rt:rise time, os:overshot).

	Reference Point					Reference Point					
Rules	1	2	3	4	5	6	7	8	9	10	11
E	P	Z	N	Z	Z	P	N	N	P	P	N
$\dot{DE}$	Z	N	Z	P	Z	N	N	P	P	N	P
CI	P	N	N	P	Z	P	N	N	P	Z	Z
	a,e,i	b,f,j	c,g,k	d,h,l	sp	i(rt)v	ii(os)vi	iii,vii	iv,viii	ix	xi

4 Co-simulation Topology

Fig. 7 shows the block diagram implemented in Labview to execute co-simulation, this block diagram contains the block *control and simulation loop* useful to simulate control systems applications, and the block *multisim design* to stablish the communication with Multisim™ software, where the power electronics and BLDCM are implemented. The electric circuit used in Multisim is presented in Fig. 1.

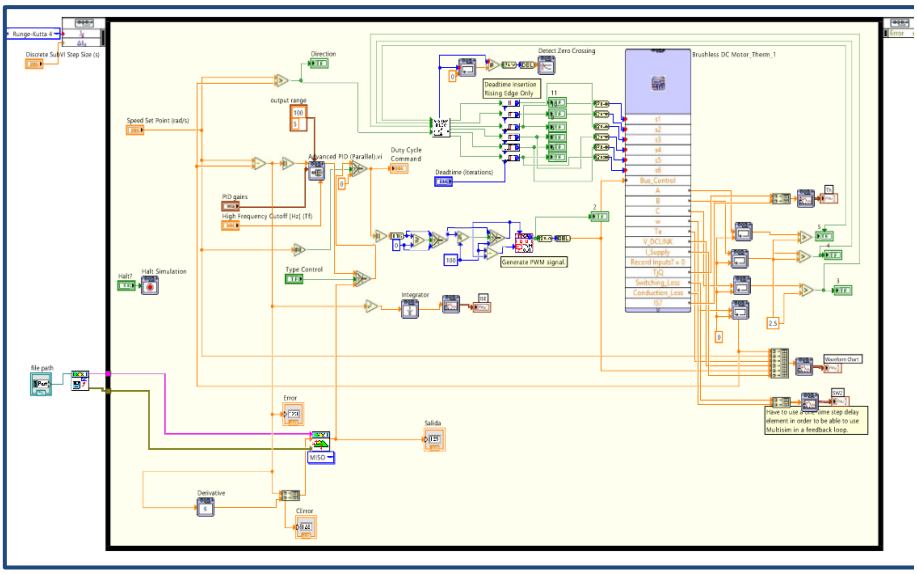


Fig. 7. Control system implemented in LabView™.

The parameters of the BLDCM are presented in Table 2. The fuzzy control in equation (13) is implemented using the Fuzzy Logic Toolbox, where input-output membership functions of Fig. 8 were programed. Centroid method defuzzification process is used in the rules shown in Fig. 9.

Table 2. Parameter of BLDCM.

	Magnitude	**Unit**
Stator Inductance	0.15	mH
Stator Resistance	0.6	Ω
Velocity constant	0.03	Vs/rad
Torque constant	0.03	Nm/A
Number of poles	*2*	

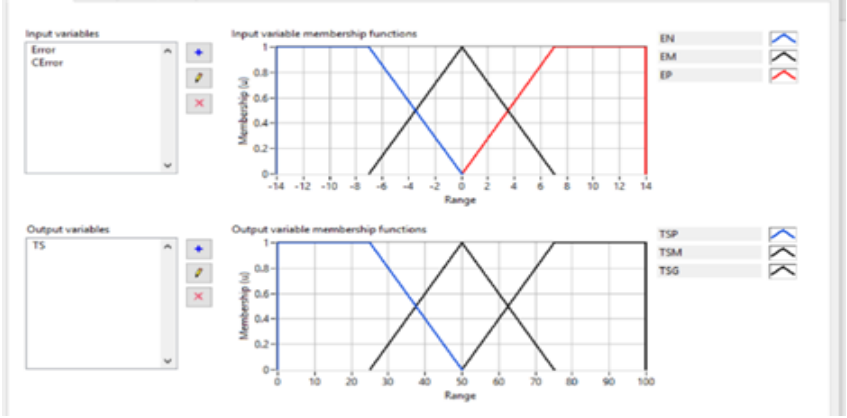


Fig. 8. Input-output membership functions.

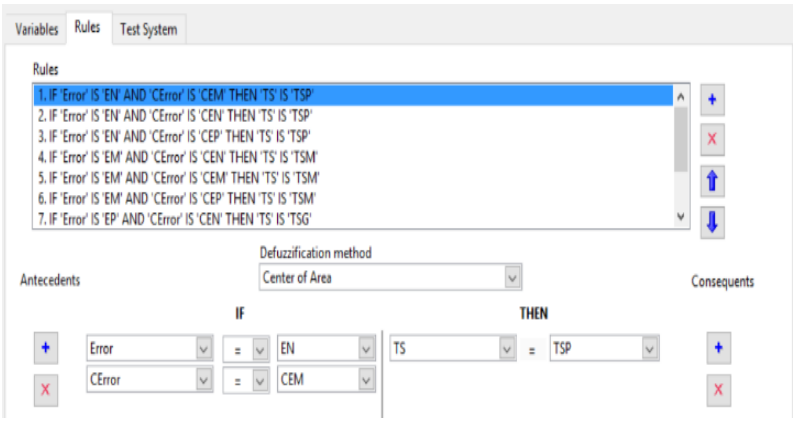


Fig. 9. Defuzzification rules.

5 Results

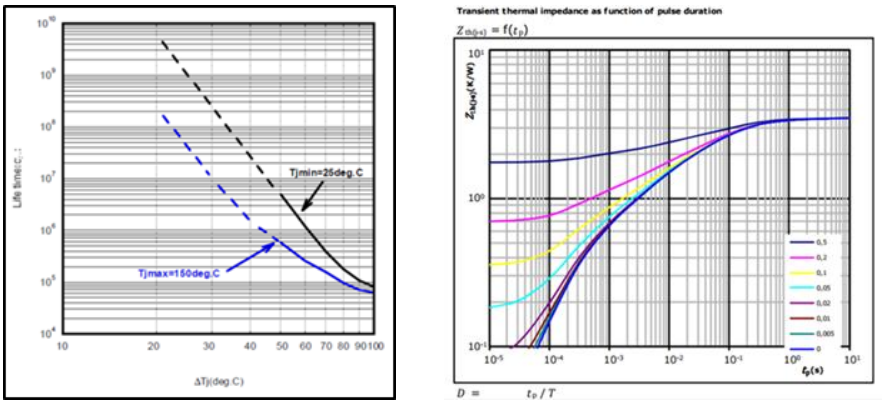
The parameters of the IGBT are presented in Table 3, and for determining the thermal conditions and life time cycle in the IGBTs, it is necessary to use the thermal information.

Thus, the relationship between the thermal impedance as function of pulse duration (see Fig. 10 (b)) and the power cycling lifetime curve are required to calculate the life cycles of the IGBTs in the power converter (see Fig. 10 (a)).

Applying the thermal information of the IGBT, it is possible to calculate the Cycles Before Failure (CBF), by equation (4). Once CBF is computed, the lifetime in years can be calculated by equation (5).

Table 3. Maximum ratings of IGBT 10-0B066PA00Sb-M992F09.

Parameter	Value	Unit
V_{CE}	600	V
$R_{CE(ON)}$	0.1	Ω
I_C	$8-6$	A
T_J, Junction Temperature	80 a 175	℃
R_{J-S}. Thermal Resistance Junction to Sink	3,50	K/W



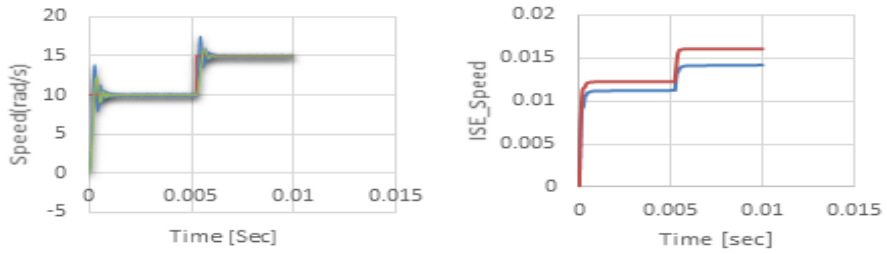
(a) (b)

Fig. 10. (a) Power cycling lifetime curve, (b) Transient thermal impedance as function of pulse duration.

The brushless dc drive is evaluated tracking a desired speed with a fuzzy controller and a proportional control. The gain of the proportional control is $K = 500$ and the fuzzy logic controller is designed in accordance to Figs. 8-9. The performance of proportional controller and the Fuzzy logic controller is presented in Fig. 11, where the fuzzy logic controller presents fast speed response and a lower controller index Integral Square Error (ISE).

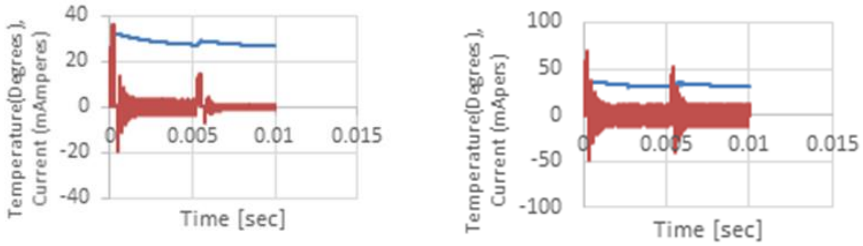
On the other hand, the maximum temperature ripple is reached when the motor starts. Besides the increments in temperature are presented when the reference speed is changed, the proportional control and fuzzy logic controller track the reference speed (see Fig. 11 and Fig. 12).

As it is observed, the fuzzy logic controller allows to reach the speed reference with a lower ISE. Moreover, the fuzzy logic controller is able to maintain a lower value of



(a) (b)

Fig. 11. (a) Speed response: proportional control (blue), Fuzzy logic controller (green), and set point (red); (b) Controller index Integral Square Error (ISE), proportional (red), Fuzzy logic (blue).



(a) (b)

Fig. 12. Current and temperature into an IGBT when fuzzy controller is running (a) y proportional controller is running (b).

temperature when is tracking the reference speed. Fig. 13 shows the temperature response when the fuzzy logic and proportional controller are running.

The lifetime for this example is computed as follows: the fuzzy logic controller has a temperature maximum of Tjmax=32.382, then ΔT=Tjmax-25°=7.3382. Using equation (7), it is obtained $CBF = 19\,973\,870\,671$. Therefore, from equation (8) $MTBF = 190.01\ [years]$. For the proportional control the computation is: $Tjmax = 35.6447$, $\Delta T = Tjmax - 25° = 10.6467$, $CBF = 541162959016419 * \Delta T^{-5.12121} = 2\,969\,916\,551$. Then MTBF=28.25 years. The results are shows at the Table 4.

6 Conclusions

This paper demonstrates one way of increasing the lifespan of power electronics by means of implementation of fuzzy logic control. The results are verified in the application of fuzzy control for regulate the speed of a BLDCM. It is concluded that on-state and switching losses, as well as maximum temperature swing, are lower using fuzzy controller than using proportional controller. The proposal and comparison are validated via co-simulation Labview-Multisim in order to use accurate electric models of Multisim and advanced control design tools of Labview. Furthermore, Multisim

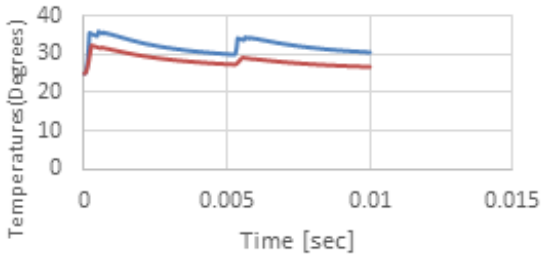


Fig. 13. Temperature into an IGBT: conventional control (blue); fuzzy control (red).

Table 4. Comparison of life time estimated with Fuzzy and Proportional control.

	Maximum Temp.	**$\Delta T = Tjmax - 25°$**	**Power cycle**	**Years**
Fuzzy	Tjmax = 32.382	7.3382	19 973 870 671	190.01
Proportional	Tjmax = 35.6447	10.6467	2 969 916 551	28.25

simulations gives estimation of temperature profiles necessary to computation of lifetime of power electronics.

Acknowledgement. This research is a product of the Project 266632 "Laboratorio Binacional para la Gestión Inteligente de la Sustentabilidad Energética y la Formación Tecnológica" ["Bi-National Laboratory on Smart Sustainable Energy Management and Technology Training"], funded by the CONACYT SENER Fund for Energy Sustainability (Agreement: S0019201401).

References

1. Xia, C. L.: Permanent magnet brushless dc motor drives and controls. John Wiley & Sons (2012)
2. Pillay, P., Krishman, R.: Application Characteristics of Permanent Magnet Synchronous and Brushless DC| motor for Servo Drives. IEEE transactions on industrial applications, 27(5) (1991)
3. Premkumar, K., Manikandan, B. V.: Speed control of Brushless DC motor using bat algorithm optimized Adaptive Neuro-Fuzzy Inference System. Applied Soft Computing, 32, pp. 403–419 (2015)
4. John-Prabu, M., Poongodi, P., Premkumar, K.: Fuzzy supervised online coactive neuro-fuzzy inference system-based rotor position control of brushless DC motor. IET Power Electronics, 9(11), pp. 2229–2239 (2016)
5. Ma, K., Vernica, I., Blaabjerg, F.: Advanced Design Tools for the Lifetime of Power Electronics – Study Case on Motor Drive Application. Proceeding of IEEE 8th International Power Electronics and Motion Control Conference (2016).
6. Power and productivity for a better world™: Load-cycling capability of HiPak IGBT modules (2014)

7. Parker, M. A., Soraghan, C., Giles, A.: Comparison of power electronics lifetime between vertical- and horizontal-axis wind turbines. IET Renew, Power Gener, 10(5), pp. 679–686 (2016)
8. Batunlu, C., Alrweq, M., Albarbar, A.: Effects of Power Tracking Algorithms on Lifetime of Power Electronic Devices Used in Solar Systems. Energies, 9, pp. 884, DOI:10.3390/en9110884 (2016)
9. Ponce-Cruz, P., Ramírez-Figueroa, F. D.: Intelligent Control Systems with LabVIEW™. Springer (2010)
10. Introducción a la Co-Simulación Digital y Analógica entre NI LabVIEW Y NI Multisim: http://www.ni.com/white-paper/13663/es/ (2017)
11. Krishnan, R.: Electric Motor Drives: Modeling, analysis and control (2001)
12. Park, J. M.: Novel Power Devices for Smart Power Applications. Phd. Dissertation, University in Vienna, Austria (2014)
13. Lee, C. C.: Fuzzy Logic in Control Systems: Fuzzy Logic Controller. IEEE Transactions on systems, man, and cybernetics, 20(2), pp. 404–418 (1990)
14. INFINEON: Transient Thermal Measurements and thermal equivalent circuit models (2015)
15. Fuji: IGBT modules application manual. Fuji Electric Co., Ltd. URL. http//www.fujielectric.co.jp/products/semiconductor/ (2017)
16. http://www.springer.com/lncs, last accessed (2016)

ISSN 1870-4069

Automatic Volumetric Segmentation of Encephalon by Combination of Axial, Coronal, and Sagittal Planes

Rodrigo Siega, Edson J. R. Justino, Jacques Facon,
Flavio Bortolozzi, Luiz R. Aguiar

Pontificia Universidade Catolica do Parana(PUCPR),
Curitiba, Parana,
Brazil

http://www.pucpr.br

Abstract. This paper describes a method of automatic volumetric segmentation of the human encephalon by Magnetic Resonance Imaging (MRI) using three anatomical planes of visualization (axial, coronal and sagittal). For mapping the volumetric topography of the encephalon we developed a set of algorithms for managing the different planes. It is intended for the segmentation of magnetic resonance images with $T1$ weighting, Inversion Recovery (IR) and Gradient Echoes GRE ($T1$ IR GRE). By combining filtering techniques and techniques of adaptive multiscale representation, directional transformation, and morphological filters, the method generates separated masks of the encephalon in the axial, coronal, and sagittal planes. Based on the masks of the three planes, reconstruction and rendering of the encephalon surface, which reveal the cortical mantle, are carried out. Tests performed using a database containing $DICOM$ images of 30 volunteers show that the proposed method of automatic volumetric segmentation is promising for the study described in this paper.

Keywords: magnetic resonance imaging, brain, anatomical planes, $T1$ IR GRE.

1 Introduction

The cerebral cortex is the outermost layer of the brain in vertebrates. It is replete with neurons and is the location where the most sophisticated and distinguished neuronal processing takes place (Figure 1) [1].

The human cortex is 2 to $4mm$ thick, with a planar area of approximately $0.22m^2$. It plays the lead role in complex brain functions such as memory, attention, consciousness, language, pereption, and thought [12].

[1] This image is a work of the National Institutes of Health, part of the United States Department of Health and Human Services. As a work of the $U.S$ federal government, the image is in the public domain

The cortical mantle is composed of hemispheres that have many convolutions. Each convolution is called a gyrus, and each depression between gyri is called a sulcus. Particularly, deep sulci may still be called fissures. This mechanism of using sulci and gyri is a way to increase the brain area without increasing its volume. Consequently, about two thirds of the surface is hidden.

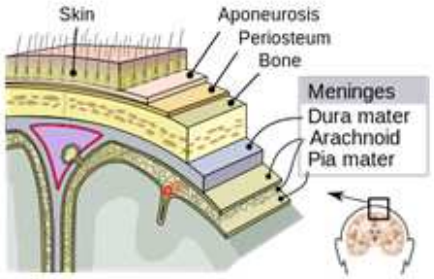


Fig. 1. Schematic diagram of different tissues involving the brain.

This study presents a method of automatic segmentation of the encephalon by MRI with relaxation time ($T1$), gradient echoes (GRE) and inversion recovery (IR). The choice of this type of imaging was motivated by the lack of technical segmentation of the brain when using MRI with inversion recovery ($T1\ IR\ GRE$). Another innovative aspect of this work is the use of three visualization planes for MR images (axial, coronal, and sagittal), providing a higher level of accuracy in the process in comparison with segmentation only by a plane. The proposed method is divided into four main stages.

First, pretreatment of the images in the three anatomical planes (axial, coronal, and sagittal) is carried out. After that, segmentation of the encephalon surface is carried out in three planes evenly and separately. Then, post-processing is carried out for, among other purposes, reconstructing the volumetric sagittal plane, based on the three anatomical planes. Finally, rendering of the surface of the encephalon, which reveals the cortical mantle, is carried out.

The rest of the work is organized as follows. First, the materials and methods used in this work are described in section 3. Section 4, describes the experiments and the discussion of the results.

2 Background

MRI contrast is influenced by several characteristics of the tissues and other factors such as relaxation time $T1$, $T2$ and density of protons PD, as shown in Figure 2. The relaxation rates affect the signal-to-noise ratio (SNR) of the image and are directly linked with the parameters of the signal, which are time of repetition (TR) and echo time (ET).

These can be adjusted to highlight imaging contrast properties such as weighting [4]. Spin-echo (SE) pulse sequences use radio frequency (RF) excitation pulses of 90° and 180° to tilt the effective magnetization vector (EMV) in transverse and inverted longitudinal planes, respectively.

The GRE -like pulse sequence uses a variable RF pulse and tilts the EMV to any angle, but not to 90° or 180°. Exams conducted with GRE pulse sequences are faster than those conducted with SE pulse sequences. However, since there is no compensation for disturbances in field homogeneity, which are constant in GRE sequences, they are more susceptible to the occurrence of unwanted artifacts [4]. Therefore, it is always important to use a filter to improve the signal-to-noise ratio.

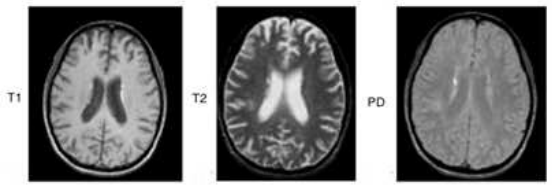


Fig. 2. Images in axial sections, $SEpure, withweightingsinT1(TE/TR = 14/500ms), T2(TE/TR = 80/500ms), andPD(TE/TR = 30/500ms)*$ ms (milliseconds).

Historically, IR techniques were widely used in the early days of the MRI (1980 1985). They produced excellent image contrast, especially for $T1$ -weighted images. However, they demanded too much time (usually 15 to $20min/sequences$). Subsequently, SE techniques largely replaced IR in most applications. Nonetheless, in the late 1990, a generation of fast SE signals (FSE) allowed high quality IR -FSE sequences to be carried out over a more reasonable period (5 to $10min/sequences$). Today, IR techniques are widely used in all branches of MRI , especially in neuroradiology and cardiac imaging [9].

MR images of the brain using $T1$ IR GRE inversion recovery sequences with isotropic volumetric images have the advantage of highlighting the difference between gray and white tissue, and water and fat.

Such highlights may significantly improve the revealing of sulci and gyri, and are also useful for surgical procedures, and functional diagnostic studies [2,9].

3 Materials and Method

The process of MRI segmentation of the encephalon, and consequently of the cortical mantle, poses numerous challenges. The presence of reduced thickness tissues, such as pia mater, turns the segmentation process into a delicate procedure, even for a specialist who intends to segment it using manual procedures.

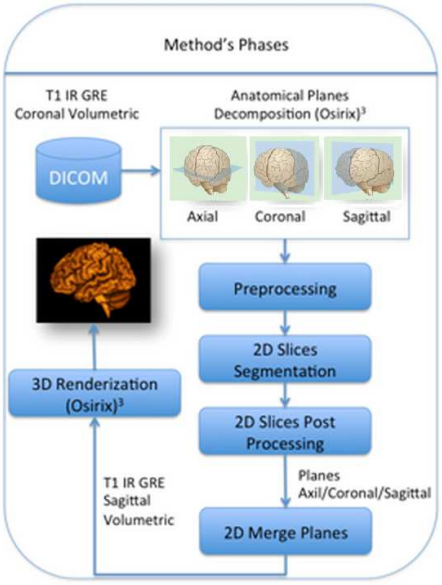


Fig. 3. Flow chart of proposed method.

The low resolution of the MRI (between 128 and 512 pixels, typically) is among the factors that interfere with this process. This hinders the identification of more subtle tissues.

Specific segmentation procedures should be considered in these cases. Despite their complexity that characterized hybrid methods [6], these have the advantage of not taking only one approach, but associating the advantages of multiple techniques used at different stages of the process, to meet the processing needs of each tissue involved more effectively. Figure 3 shows the flow chart of the proposed method.

The diagram is divided into four main stages: preprocessing of images of the three anatomical planes (sagittal, axial, and coronal), segmentation of $2D$ slices, post-treatment of segmented slices, and volumetric sagittal reconstruction. Two other steps are performed by the Osirix tool. The first one consists of a decomposition process of the slices in the three anatomical planes, and the second focuses on the rendering of the brain.

3.1 *MRI* Database

In this study, a database in the Digital Imaging and Communications in Medicine ($DICOM$) standard was employed for MRI. The $DICOM$ were obtained in two volumetric sequences: $T1$ GRE and $T1$ IR GRE, with Magneton Symphony $1.5T$ Siemens equipment. The base is composed of 30 sets of $DICOM$ files of adult volunteers, aged 25 years on average. Of these, 16 volunteers were women, and 14 volunteers were men.

The collection of images was performed after approval by the Research Ethics Committee of $PUCPR$[2], protocol number $746,028$. For this study, only $T1$ IR GRE images were used.

3.2 MRI Preprocessing

During preprocessing, each image of the $DICOM$ package (slices) is initially decomposed into the three anatomical visualization planes (axial, coronal, sagittal), according to Figure 3.

Plane redundancy is used by the merge algorithm to generate the final volumetric plane. The generation of images in different planes was made using the Osirix tool [3]. At the end of the segmentation process, individual planes are grouped again to generate a new $DICOM$ package with the volumetric plane. Any of the three planes can be used for reconstruction. In this study, the sagittal plane was chosen. Then, this plane may be rendered and displayed in a $DICOM$ viewer, which for this study was Osirix.

The set of slices of each anatomical plane undergoes a sequence of filters with the aim to prepare it for the segmentation phase. Each step of the process is presented below.

Denoising. An important step in the preparation of MRI is the attenuation of noise inherent in image acquisition. The $T1$ IR GRE weighting offers $TR = 4,000ms$, which provides a high signal-to-noise SNR ratio. Still, a procedure for noise reduction was adopted with the aim of minimizing the influence of the unwanted artifacts of subsequent steps. The most common acquisition method for obtaining MRI is acquisition in the frequency domain [8] . The acquired signal $Y(\mu, v)$ may be decomposed into two components: the signal of interest $S(\mu, v)$ and noise $N(\mu, v)$, in horizontal μ and vertical v frequencies, respectively:

$$Y(\mu, v) = S(\mu, v) + N(\mu, v). \tag{1}$$

The noise $N(\mu, v)$ is a complex Gaussian white noise. This noise derives from the temperature of the tissue. Therefore, the structure of this noise depends on the signal $S(\mu, v)$. Thus, the noise in MRI depends on the signal, whose properties are best described by a Rician distribution [1,3].

In this study, the wavelets domain was chosen for the noise image filtering process [5,11], since the processed discrete $2D$ wavelets ($2D$-DWT) tend to concentrate the energy of the desired S signal in a reduced set of coefficients. Therefore, the transformed wavelet of a noisy image consists of a small number of coefficients with high SNR, which is desirable. In addition, a high number of coefficients with low SNR ratios are disposable. After excluding the coefficients with low $SNRs$, the image is reconstructed using the inverse wavelet transform.

[2] Pontifcia Universidade Catolica do Parana, Brazil.
[3] Osirix v.5.7.1, 32 bit / Free Version, Pixmeo Sarl.

The challenge to be faced with regard to MRI noise is to determine which factors have a high SNR ratio.

For our purposes, the transform wavelet was used with a $HAAR$ base function with $level - 2$ hard thresholding. The choice was based on the results already published in other works [5,7,11].

Histogram Equalization. Histogram equalization was necessary in order to establish a better balance of gray levels, thus homogenizing the areas. This makes easier the removal of unwanted artifacts, which is typical of GRE acquisition. Consider a set of MR images ϕ_i in which i is a $DICOM$ set, acquired in $T1$ IR GRE in 256 gray levels.

The histogram in question highlights the region of interest (ROI), the brain, which is represented as a concentration of gray levels corresponding to approximately 25% of total 256 levels represented in the image. The histogram equalization in 25 levels enables the resizing of the level scale corresponding to the ROI . This procedure highlights the gray levels of the ROI to the detriment of other regions.

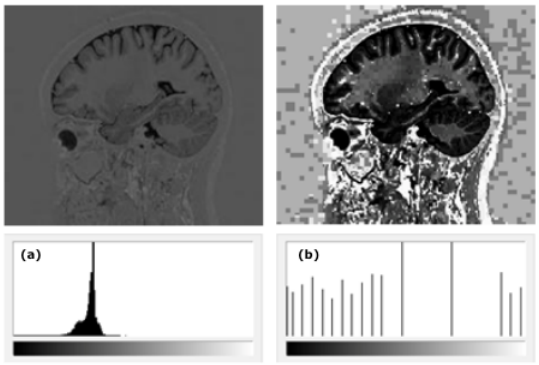


Fig. 4. (a) Image $T1$ IR GRE and its histogram, (b) image equalized and its histogram.

Grayscale Morphological Reconstruction. Morphological reconstruction in gray levels aims to reestablish the connectivity between adjacent pixels and to attenuate the influence of artifacts typical of GRE acquisition. The process occurs in a controlled manner through successive expansions [7]. The control of geodesic dilatation is accomplished through an image marker ϕ_i, that is, the desired image. This is carried out under a base image g, which is an image with a black background and edges from the image ϕ_i in this case 2:

$$\phi_i' \leftarrow g^{T^i} D^{\phi_i} \leftarrow (\phi_i \oplus_g D)^{\infty}. \tag{2}$$

Then, a nonlinear median filter was applied with the aim of attenuating the noise resulting from the reconstruction step.

It is very effective in eliminating image impulse noises (Poisson noise) and preserving edges without introducing new values of gray levels in the image [13]. The nonlinear median filter chooses the element in a median position on the ordered list of values of gray levels in a window (usually 3×3) around the center pixel.

Outline Smoothing. Candes and Donoho [5] developed a multiscale transform called a curvelet. This transform was designed to represent edges and other singularities along curves, and is much more efficient than traditional transforms. The curvelet transform, as well as the wavelet transform, has indexed elements of structure by parameters of scale and location. Unlike the wavelet transform, the curvelet has directional parameters, and the curvelet pyramid contains elements with a high degree of directional specificity. In addition, it is anisotropic in scale, unlike the wavelet. In this study, the Fast Discrete Curvelet Transform ($FDCT$) and the inverse ($IFDCT$) [13] were used to remove possible discontinuities and unwanted noise in the contour region of the ROI .

In this study, $2D$ $FDCT$ by wrapping was used, since it is the simplest transform. It has low redundancy, and it is quick. The most significant 10% coefficients were used, thus eliminating small edge distortions, and for smoothing out the variability of the bottom of the image [13].

3.3 2*D* Slice Segmentation

In order to separate the ROI from the rest of the slice image, a mask was created by the binarization of images δ_i of each slice i. The binarization process for each slice took into consideration the different tissues constituting the brain, and the position of the slice of the $DICOM$ set. The most central region of the brain is composed of white tissue (predominantly gray levels closer to white) and a peripheral region composed of gray tissue (predominantly shades of gray closer to black). Each slice was treated according to mean indexes β observed in a range consisting of a set of slices, as shown in Figure 5. The values were obtained by analyzing samples from six different volunteers. The binarization algorithm is based on the average values in the three planes, considering similarities in the variability of thresholds. The binarization process by a fixed threshold in β of the image δ_i occurs in each slice in the three different planes. The β value represents the percentage of normalized gray levels between 0 (white) and 1 (black). Finally, a process of reversing the image takes place to keep black the background 3:

$$\delta_i' \leftarrow inverse\{treshold((\beta, \delta_i)|0 \leq \beta \leq 1\}. \tag{3}$$

***ROI* Mask Selection.** After the binarization process, an algorithm for filling the area with a closed contour was applied. Therefore, many λ_Θ candidate regions

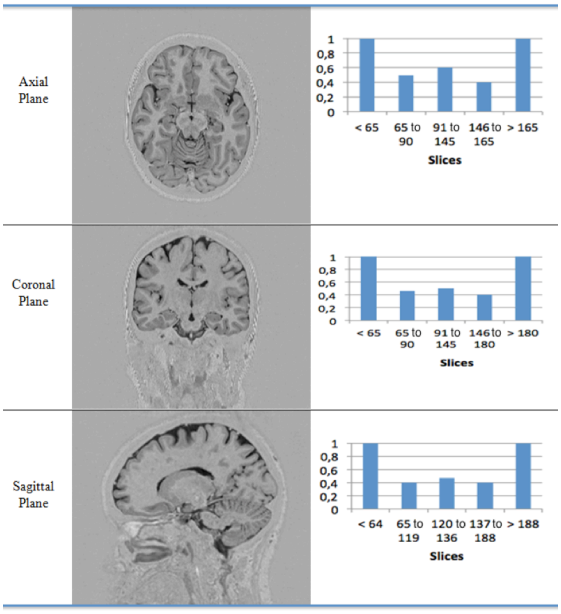


Fig. 5. Gray-level variations in $T1\ IR\ GRE$ images from three planes (image brightness is +50% for better viewing).

to mask the ROI of i were formed. The selection of a candidate was made using the selection of the greater area of set:

$$ROI_i \leftarrow max\{\lambda_\Theta | \Theta \in \mathbb{N}\}. \tag{4}$$

3.4 2*D* Slice Post-Processing

From the mask ROI_i, a combination of the original image ϕ_i may be performed. Consequently, ROI may be delimited. However, in this first stage of segmentation, it was not possible to highlight regions of sulci and gyri in the cortical mantle adequately. The region in question is known as subarachnoid space and involves the entire surface of the brain, Figure 1.

To remove the subarachnoid space, repetition of three previously mentioned steps in the process is necessary. The steps are as follows: thresholding with cutoff obtained by the algorithm of [10] with the aim of removing the subarachnoid space, application of a 3×3 median filter for smoothing the contour, and the selection of the greater area for the removal of unwanted artifacts that persist, as shown in Figure 6. The new mask ROI_i' is ready to be merged between anatomical planes ϕ_i.

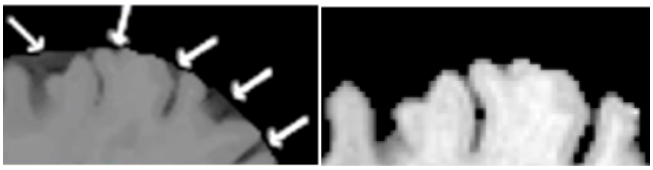


Fig. 6. Post-processing result.

3.5 2*D* Merge Planes

The process of pairing is a pixel to pixel combination between images of the axial, coronal, and sagittal planes after slice-by-slice segmentation, Figure 7. This is carried out to generate an image whose area is more likely to belong to the encepholon region. Reconstruction of the volumetric sagittal plane is then achieved by the combination of the three anatomical segmented planes. Reconstruction is based on the principle of decision by majority vote. That is, each pixel viewed in the sagittal plane image is identified in the three anatomical planes. If the pixel is identified as belonging to the *ROI* in at least two of them, it is maintained in the volumetric plane. Otherwise, it is eliminated. Given that the central slices of each anatomical plane are less susceptible to undesirable content in comparison with the edges, a weight is given to each slice, decreasing from the central position to the periphery. If there is only one anatomical plane, this indicates that the pixel belongs to the *ROI* , and an assessment of the weight of this slice is carried out before disposing it. If the weight of this slice, receiving a positive result, is greater than the other two, the pixel is maintained.

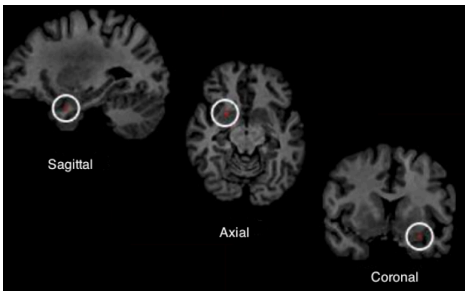


Fig. 7. The vision of the same pixel in the three planes of projection.

3.6 3*D* Renderization

A rendering of the volumetric sagittal plane, resulting from the merge algorithm between the three anatomical planes, can be read by a *DICOM* standard visualization tool. In this study, the Osirix tool was used. Figure 8 presents

the results for a set of volunteers from the database. The volumetric images, seen in the three anatomical planes, emphasizing the presence of the sulcus and gyrus of the cortical mantle.

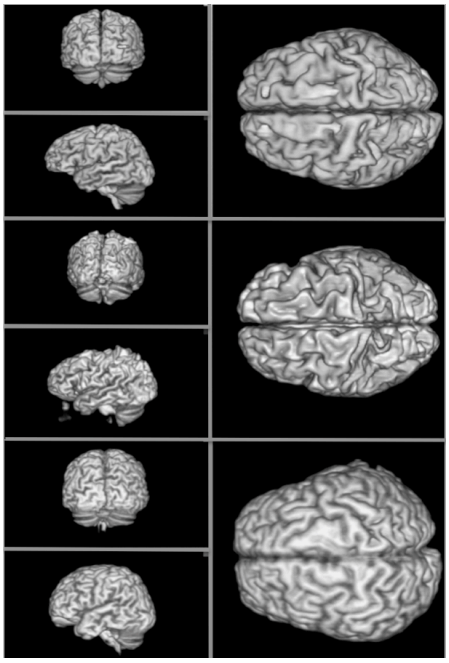


Fig. 8. The view of the cortical mantle on the three viewing planes.

4 Experiments and Analysis

Evaluating the results of the segmentation proved to be a laborious task. The three anatomical planes were processed. Considering that each plan has 250 images (slices) on average, generating a perfect image (ground truth) of each segmented slice proved to be impractical. Another possible solution was to seek an available tool to perform this type of comparison. However, owing to the use of $T1$-type images with recovery of $T1$ IR GRE gradient-echo inversion, which is unprecedented, no tool was found. This also represents one of the contributions of this study. Thus, the assessment was visual and was performed as follows. Each image segmented by the proposed method was examined visually, by a field specialist. If the image displayed artifacts resulting from the method, segmentation of the image would be deemed erroneous.

The visual assessment suggested that the method was promising. It had an exact segmentation rate in 80% of the slices. As mentioned in the introduction

of this paper, the processes of segmentation are strongly linked to weights used in the acquisition of the slices, which does not guarantee flexibility in this regard. The similarity between texture tissue and the degree of gray levels is another factor that affects the efficiency of the method. We observed that the proposed method did not always succeed in segmentation. It displayed artifacts originating from inadequate segmentation. Furthermore, in other cases, it had failures resulting from the improper disposal of certain tissues.

Figure 9 depits some examples. The majority of problems occurred in a slice or in a reduced group of slices. Nevertheless, this affects not only the quality of slice segmentation but also the rendering process, incorporating nonexistent artifacts into the cortical mantle. As previously mentioned, the method was promising, even though it requires in depth studies to improve not only the steps of segmentation but also the plane merge algorithm.

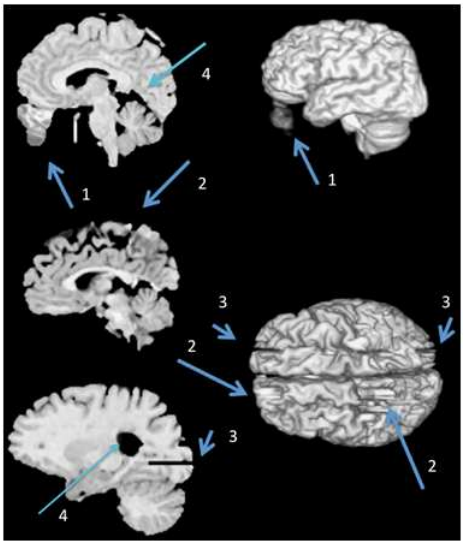


Fig. 9. Examples: (1) slice with artifacts resulting from inadequate segmentation, (2, 3, and 4) slices with inadequate removal of tissue

.

5 Conclusion and Future Works

This paper presented an automatic volumetric segmentation algorithm for the encephalon. It used $T1$ type magnetic resonance images with recovery of inverted IR, gradient echoes GRE and involving the three anatomical planes of visualization (axial, coronal, and sagittal). The use of a complex but reproducible sequence of processes (using techniques such as denoising, contrast enhancement,

morphological reconstruction, outline smoothing, and slice binarization) to generate the encephalon mask in three axial, coronal, and sagittal planes separately proved to be efficient. This enabled the reconstruction and rendering of the encephalon surface with high precision. Experiments performed in a database with 30 volunteers showed an average rate of 80% in slices segmentation of each $DICOM$ set belonging to the database.

The first contribution of this work is the use MRI in $T1$ $IR\,GRE$ acquisition, which was of paramount importance for obtaining such results.

Another contribution is the use of three visualization plans of anatomic projection. Using the three plans, there is the possibility of an additional plan complementing the results of the others when the region of interest does not have appropriate segmentation. Consequently, this provides a greater level of accuracy for the process in comparison with segmentation by a single plan.

A third contribution from a computational point of view is the combination of several techniques that provided an innovative method. This makes it possible for studies on anatomical patterns of the encephalon to be conducted with greater confidence and agility. These techniques display more detailed images than the slices analyzed in a single plane, and they facilitate obtaining images of patients by automated means of segmentation.

However, despite the benefits of using this type of image, the algorithm was not able to segment all slices properly, maintaining artifacts in some slices and removing part of the tissue in others. This is an indication of the size of the challenge to be faced and overcome.

In our future work, we intend to perfect both the segmentation and the merge algorithms, with the aim of correcting the mistakes found.

References

1. A., A.: Wavelet methods in statistics: Some recent developments and their applications. Statistics Surveys 1, 16–55 (2007)
2. A., M.: Noise in mri. Magn Reson Med 36(3), 494–497 (1996)
3. Abramovich F., Bailey T. C., S.T.: Wavelet analysis and its statistical applications. Journal of the Royal Statistical Society Series 49(6810), 1–29 (2000)
4. B., G.C.: Physical Principles of Computed Tomography and Magnetic Resonance Imaging, vol. 1. Williams and Wilkins, 2 edn. (2000)
5. Candes E. J., D.D.I.: Curvelets: A surprisingly effective non adaptive representation for objects with edge. Vanderbilt Univ. Press 1, 105–120 (2000)
6. Despotovic I., Goossens B., P.W.: Mri segmentation of the human brain: Challengers, methods, and applications. computational and mathematical methods in medicine. Computational and Mathematical Methods in Medicine 49(45341), 23 (2015)
7. Dougherty E. R., L.R.A.: Hands-On Morphological Image Processing, Tutorial Texts in Optical Engineering, vol. TT59. SPIE Press, 1 edn. (2003)
8. Kim S., Richter W., U.K.: Limitations of temporal resolution in functional mri. Magn Reson Med 37(1), 631–636 (1997)
9. Mancall E., B.D.: Gray's Clinical Neuroanatomy: The Anatomic Basis for Clinical Neuroscience, vol. 1. Elsevier Saunders, 1 edn. (2010)

10. N., O.: A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man and Cybernetics 1(SMC-9), 62–66 (1979)
11. R., N.: Wavelet-based rician noise removal for magnetic resonance imaging. IEEE Transactions on Image Processing 8(10), 1408–1419 (1999)
12. Tamraz J.C., C.Y.: Atlas of Regional Anatomy of the Brain Using MRI - With Functional Correlation, vol. 1. Springer-Verilag, 1 edn. (2006)
13. Xiaojun W., L.W.: Ct/mri medical image based on sub-band coefficients selection through curvelet transform method. Journal of Chemical and Pharmaceutical Research 6(3), 1083–1088 (2014)

ISSN 1870-4069

Wavelet Local Feature (WLF) Pattern Recognition System

Carolina Barajas-García, Selene Solorza-Calderón

Universidad Autónoma de Baja California, Facultad de Ciencias,
Tijuana-Ensenada, Baja California,
Mexico

{cbarajas,selene.solorza}@uabc.edu.mx

Abstract. In this paper is presented a pattern recognition methodology based on local feature extraction. The purpose of this system is to identify and locate, in three different scale pyramids, key points that represent relevant information of the image; this information is stored in a descriptor which is used to compares the key points of two images and know if they have similar information, or if they are the same images. This methodology uses the Haar wavelet transform to generate the three scale pyramids. This transform is used because it has several properties, such as noise elimination, multi-resolution analysis, and detection of diagonal, horizontal and vertical edges. The performance of this system was tasted using images with different scales and comparing the results with the Scale Invariant Feature Transform (SIFT) and Speeded-Up Robust Features (SURF) methodologies. The WLF system showed to has the highest percentage of correct point-matching.

Keywords: wavelets, local features, pattern recognition system, SIFT, SURF.

1 Introduction

Since the middle of last century, the extraction of features in image have been an active area in the pattern recognition field. A feature is a piece of information that is relevant for the resolution, through digital images, of some specific computational task. The features can be structures in the image like points and edges. These features are included in a descriptor, which specifies elementary properties of the object, such as shape, color, texture, among others. An example of a system based in feature extraction is SIFT (Scale Invariant Feature Transform), introduced by Lowe in 2004 [1,2].

SIFT is widely used, because of the innovative idea of using key points to represent the relevant information, allowing to work with images containing

several objects, scenarios and even fragments of objects, unlike other patterns recognition systems as Principal Component Analysis (PCA) [3], or the binary mask system based on one-dimensional signatures [4,5]. However, SIFT have several disadvantages, the computation time is considerable and has low performance with images that present few amount of noise or inhomogeneous illumination. In addition, it only allows the use of grayscale images and its generalization for the classification of color images is complicated and uses a considerable computational time.

There are variants of SIFT that try to improve its performance and their deficiencies. The most important is SURF (Speeded-Up Robust Features), presented by Herbert Bay in 2006 [6], which is a fast version of SIFT, it reduces the computational cost time but decreases its efficiency considerably. Another variant is WSIFT, proposed by Lim and collaborators in 2009 [7], based on the wavelet transform. This work shows a comparative analysis of the performance of the system using the Haar, Daubechies and Gabor wavelets transforms. The WSIFT system is not an improvement of SIFT, since it continues to maintain the same problems.

The proposal for this work is to construct a pattern recognition systems based on local features extraction using the Haar wavelet transform, since it has several properties such as excellent performance to detect and eliminate noise, 2D multiresolution analysis, scale and translation variables, that allow us to improve the system's efficiency. The rest of this work is organized as follows: Section 2 presents the Haar wavelet transform. Section 3 describes the procedure for the WLF pattern recognition system. Section 4 shows the tests and results. Finally, conclusions are given in Section 5.

2 The Haar Wavelet Transform

Given a function $f(t) \in L^2(\mathbb{R})$, that is f is an square-integrable function, the wavelet transform is defined as:

$$W_{\psi_{a,b}}\{f(t)\} = \int_{-\infty}^{\infty} f(t)\overline{\psi_{a,b}(t)}dt, \tag{1}$$

where

$$\psi_{a,b}(t) = \frac{1}{\sqrt{|a|}}\psi\Big(\frac{t-b}{a}\Big), \tag{2}$$

being ψ the wavelet analysis and a, b the scaling and translational parameters, respectively.

In this work the Haar wavelet was used, which is defined as follows:

$$\psi(t) = \begin{cases} 1, & 0 \leq t < \frac{1}{2}, \\ -1, & \frac{1}{2} \leq t < 1, \\ 0, & \text{other way.} \end{cases} \tag{3}$$

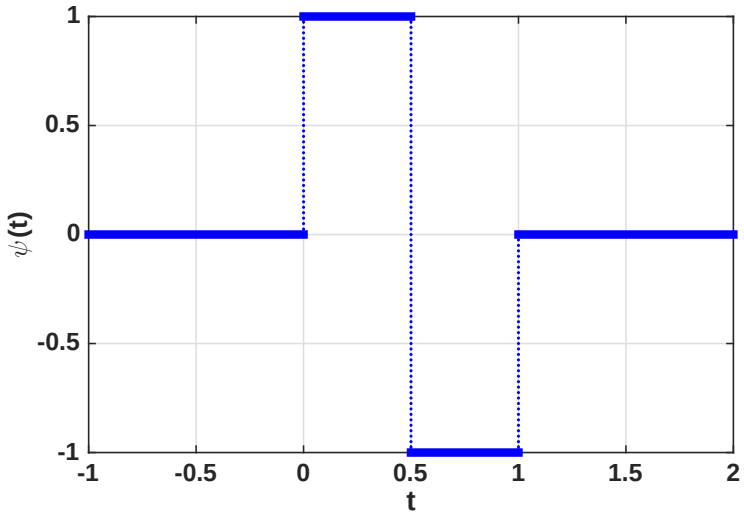


Fig. 1. Graph of the Haar wavelet transform.

The graph of the Haar wavelet transform can be seen in Fig. 1.

When the wavelet is applied in two dimensions, the image is splitted into four sub-images (that are half of the size of the given image), with different frequencies: high-high (HH), high-low (HL), low-high (LH) and low-low (LL). To apply the wavelet transform on more than one level, one of the four sub-images is selected and the same procedure is performed, this procedure is known as 2D multiresolution analysis (2D MRA). The 2D MRA procedure sketch is given in Fig.2. In this work it was used the sub-images HH, LH and HL, Fig. 3 shows that.

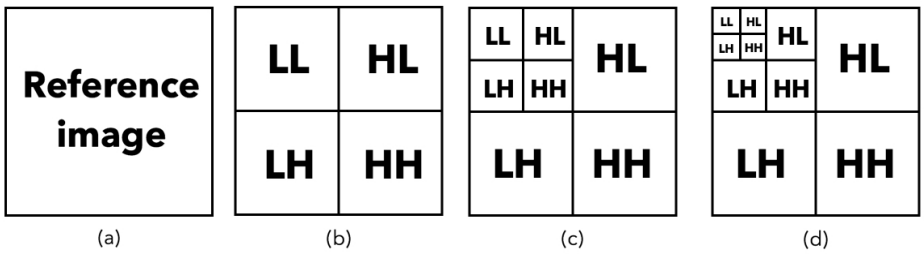


Fig. 2. Sketch of the wavelet transform levels (2D multiresolution analysis). (a) Reference image. (b) First level of the wavelet transform. (c) Second level. (d) Third level.

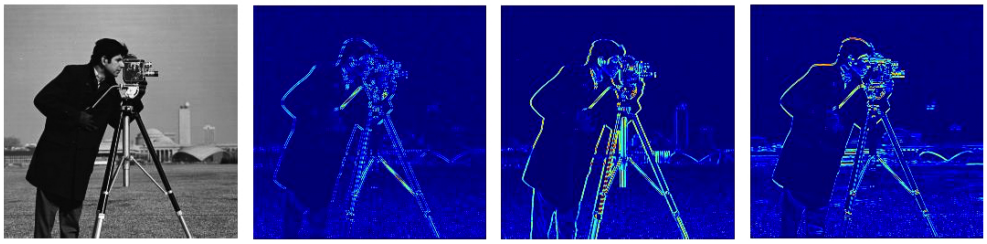


Fig. 3. Sub-images of the Haar wavelet transform. (a) Reference image. (b) HH frequencies. (c) LH frequencies. (d) HL frequencies.

3 Wavelet Local Feature (WLF) System

This section describes the procedure to build the WLF system, which consists of three steps:

1. Generation of wavelet pyramids.
2. Localization of key points.
3. Generation of key point descriptor.

3.1 Generation of Wavelet Pyramids

The purpose of construct a wavelet pyramid is to create a space where the most important details of an image are highlighted, using the wavelet transform this is possible, since this transform enhances the edges of the image. Three wavelet pyramids will be generated, using the three sub-images obtained from the Haar wavelet transform (Fig. 3), as was done in section 2.

The procedure for generating the first pyramid, P_{HH}, consists of taking the reference image and applying the Haar wavelet transform in 5 levels using only the HH frequencies, in this way the diagonal details of the image are enhanced, Fig. 4a. The second pyramid, P_{LH}, is generated by taking only the LH frequencies, highlighting the vertical details, Fig. 4b. Finally, the third pyramid, P_{HL}, is created using the HL frequencies, enhancing the horizontal edges, Fig. 4c.

3.2 Localization of Key Points

The first step for locating the key points is filters each element into the pyramid with a binary edge mask. For example, the first level of P_{LH} is filtered using the binary edge mask M^1, that is of the same size of P^1_{LH}. Mathematically this is:

$$F^1_{LH} = P^1_{LH} \circ M^1, \tag{4}$$

where the super index indicates the level of the pyramid and $\circ$ means an element-wise product or Hadamard product [8]. The process is shown in Fig.5.

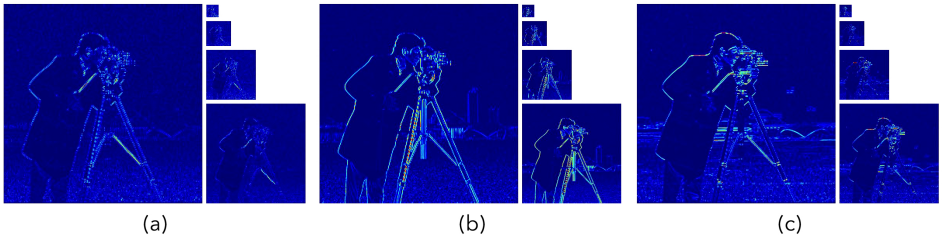


Fig. 4. Wavelet pyramids. (a) P_{HH} pyramid. (b) P_{LH} pyramid. (c) P_{HL} pyramid.

The filtering process is performed in the three pyramids like:

$$F^n_{HH} = P^n_{HH} \circ M^n, \tag{5}$$

$$F^n_{LH} = P^n_{LH} \circ M^n, \tag{6}$$

$$F^n_{HL} = P^n_{HL} \circ M^n, \tag{7}$$

where $n = 1, ..., 5$.

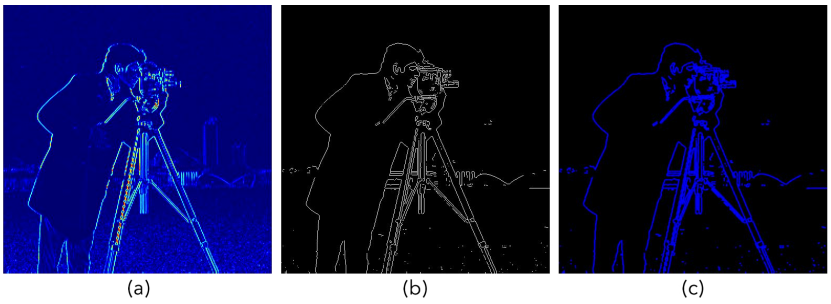


Fig. 5. (a) P^1_{LH}. (b) Edge binary mask M^1. (c) Filtered image.

Once the three pyramids are filtered with the edge binary masks, the next step is to analyze if these points are maximum or minimum.

This is done by comparing each pixel with its 8 neighbors at the same level and the 9 neighbor's into each adjacent scales. If the selected pixel is the maximum or minimum of the 27 points, then it is taken as a key point. Figure 6 shows the key points obtained for the three wavelet pyramids.

3.3 Generation of Keypoint Descriptor

The first step to generate the descriptor is to take a neighborhood of 20×20 pixels around the key point and divide it into 16 sections of 5×5 pixels (Fig. 7a).

Fig. 6. (a) Key points detected in P_{HH}. (b) Key points detected in P_{LH}. (c) Key points detected in P_{HL}.

Then a Haar-like filters d_x and d_y (Fig. 7b and c, respectively), are applied to each of the sections:

$$fx_n = S_n \circ d_x, \tag{8}$$

$$fy_n = S_n \circ d_y, \tag{9}$$

where S_n is the 5×5 pixel sections and $n = 1, ..., 16$.

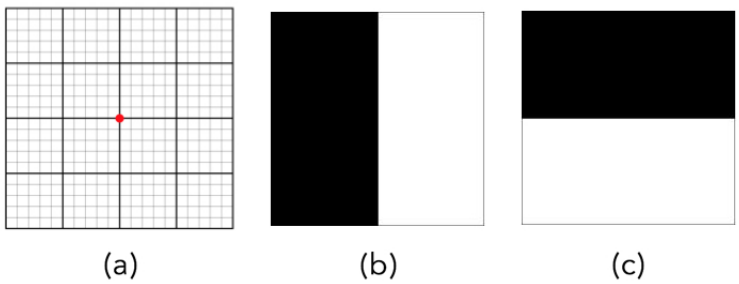


Fig. 7. (a) Neighborhood around a key point. (b) Haar-like filter d_x. (c) Haar-like filter d_y.

The addition of all intensity values of fx_n, fy_n, $|fx_n|$ and $|fy_n|$ are obtained, in this manner 4 scalar values are given for each of the 16 sections, having a total of 64 values. These 64 values form the key point descriptor [6].

4 Tests and Results

In order to be able to compare two images and know if they share information, the key points of the reference image and the problem image were located. Then,

the descriptors for all these key points were generated. After that, the euclidean distance of one descriptor of the reference image with all descriptors of the problem image are calculated.

The pair that has the minimum distance and also if that minimum distance satisfies to be less than a threshold of 0.15, the pair will be labeled as a match. The same procedure is applied to the rest of the descriptors of the reference image.

The WLF system was tested using the reference images of Fig. 8 with 10% and 15% reduced versions of that images. The performance of the WLF system was compared with SIFT and SURF systems, noticing that the thresholds values for the euclidean distance for SIFT and SURF are 0.8 and 0.3, respectively. Table 1 shows the results of the WLF system comparing the reference images in Fig. 8 with their reduced versions. The third row shows the number of key points that made match. In the fourth row are the number of key points that made match and are correct. In the last row the percentage of correct matched key points are given. In Table 1 is observed that for the 10% reduced image the percentage of correct matched key points is 92% and for the 15% reduced image the percentage is approximately 91%. Tables 2 and 3 show the results obtained for the SIFT and SURF systems, respectively, showing that the SIFT system presents a percentage of 89% for a 10% reduced image and 85% for a 15% reduced image. Finally, for the SURF system, 75% and 64% was obtained for the 10% and 15% reduced images, respectively.

Fig. 8. (a) Reference image: Camera. (b) Reference image: Lena.

Another result that is observed in Tables 1 to 3, is the number of key points that are detected in the different systems. The WLF system detects almost three times more points than SIFT, this is to be expected since in WLF there are three pyramids to locate key points while in SIFT there is only one. For this reason there are more key points that represents more information of the image, which leads to a higher percentage of correct matched key points. The SURF system detects few points, compared to WLF and SIFT, so the percentage of correct matched points is low. Here it was reaffirmed that by reducing the computational cost, the efficiency decrease considerably.

Table 1. WLF system.

	Camera		Lena	
	10%	15%	10%	15%
# matched key points	840	705	993	734
# correct matched key points	775	641	915	670
% correct matched key points	92.26%	90.92%	92.15%	91.28%

Table 2. SIFT system.

	Camera		Lena	
	10%	15%	10%	15%
# matched key points	255	221	289	249
# correct matched key points	229	184	259	222
% correct matched key points	89.8%	83.26%	89.62%	89.15%

Table 3. SURF system.

	Camera		Lena	
	10%	15%	10%	15%
# matched key points	171	166	151	151
# correct matched key points	136	112	109	90
% correct matched key points	79.53%	67.47%	72.18%	59.6%

5 Conclusions

The WLF pattern recognition system presented in this work shows a higher percentage of correct matched key points, which means that the system is correctly matching two images even if one of them is reduced by 10 % or even 15 %. This system uses the Haar wavelet transform for its great performance by enhancing edges and highlighting relevant image information. This wavelet transform is used to generate the wavelet pyramids which are useful for locate the key points. To compare images, a descriptor was created using Haar-like filters. The methodology proposed in this work was compared with SIFT and SURF systems, showing that the WLF is the system that presents a greater percentage of correct matched points.

References

1. Lowe, D. G.: Object recognition from local scale-invariant features. The proceedings of the seventh IEEE international conference, 2, pp. 1150–1157 (1999)
2. Lowe, D. G.: Distinctive image features from scale-invariant key points. International journal of computer vision, 60(2), pp. 91–110 (2004)
3. Jolliffe, I.: Principal component analysis. John Wiley & Sons (2002)
4. Barajas-García, C., Solorza-Calderón, S., Álvarez-Borrego, J.: Classification of fragments of objects by the Fourier masks pattern recognition system. Optics Communications, 367, pp. 335–345 (2016)

5. Ventura, A. S., Álvarez-Borrego, J., Solorza, S.: Adaptive nonlinear correlation with a binary mask invariant to rotation and scale. Optics Communications, 339, pp. 185–193 (2015)
6. Bay, H., Tuytelaars, T., Van-Gool, L.: Surf: Speeded up robust features. Computer vision?, ECCV, pp. 404–417 (2006)
7. Lim, J., Kim, Y., Paik, J.: Comparative analysis of wavelet-based scale-invariant feature extraction using different wavelet bases. Signal Processing, Image Processing and Pattern Recognition, pp. 297–303 (2009)
8. Golub, G. H., Van-Loan, C. F.: Matrix Computations. 3 (2012)

ISSN 1870-4069

Contrast Enhancement of Color Images Using a Multi-Objective Optimization Framework

Luis G. Moré[1], Diego P. Pinto-Roa[1], José Luis Vázquez Noguera[1,2]

[1] Universidad Nacional de Asunción, Facultad Politécnica, Paraguay

[2] Universidad Americana del Paraguay, Paraguay

{lmore, dpinto, jlvazquez}@pol.una.py

Abstract. Contrast Enhancement (CE) is a fundamental preprocessing step for several applications, and also for further decision making processes related. This task has been addressed successfully for gray-scale images using pure Multi-Objective Optimization (MOO); nevertheless, difficulties arise when performing MOO for color images. This paper presents a pure MOO approach with automatic CE for color images, taking into account evaluation metrics better suited for color spaces, which are designed to achieve the improvement in contrast and also control the noise introduced because of the contrast variation seen during the process. A series of experiments were conducted in order to assess the correctness of this approach, and the results consist of a set of contrast enhanced images, with different compromise rates between contrast modification and noise introduction. It appears that the results obtained are promising, and the numeric values of the optimization metrics are analyzed using correlation tables and discussed using the Pareto Front obtained from these values.

Keywords: Multi-objective optimization, contrast enhancement, MOPSO, CLAHE, color spaces.

1 Introduction

Contrast Enhancement (CE) is a fundamental preprocessing step for several image processing applications such as Medical Imaging (Computer Aided Diagnosis [2], Computerized Tomography Imaging [8], Magnetic Resonance Imaging [4] and others), Remote Sensing [11], and so on.

Techniques based on Histogram Equalization have been extensively proven to be valid when addressing CE problems [5,15,18,10]. Meta-Heuristics such as Mono-Objective Optimization, and also Multi-Objective Optimization (MOO) have been tested successfully in order to solve CE problems on gray-scaled images [12,13,16,7]. However, MOO applied to color images poses additional difficulties because it is necessary to preserve color information present therein.

Our proposal consist in testing images transformed from RGB color space to $YCbCr$ in order to perform MMO-based CE.

Contrast Limited Adaptive Histogram Equalization (CLAHE) is applied over the Y channel of the test image in order to modify contrast, and the resultant image is transformed back to RGB in order to evaluate the similarity between color channels.

The rest of the paper is organized as follows: in Section 2, the fundamental concepts for this work are presented, in Section 4 the CE problem is posed, and our approach is presented, in Section 5 the results achieved are discussed in detail, and finally in 6 some final points are remarked.

2 Theoretical Framework

This sections presents a brief introduction of the concepts used in the paper.

2.1 Color Spaces Adopted

Original images are represented using the RGB color space [6], which is a $N \times M \times 3$ array of color pixels. Every color pixel is represented by an element $[z_r\ z_g\ z_b]$ of the array previously mentioned, where z_r, z_g, z_b are the red, green, and blue components of the color pixel in a specific location. Original images are then transformed to the $YCbCr$ color space [6], which is a representation widely used in digital video.

The main advantage is that the Y component here represents the luminance information of the image, meanwhile the Cb component represents a difference between the blue component and a reference value, and the Cr component is the difference between the red component and a reference value. Another important advantage of this representation is that the conversion from RGB, and back to RGB is straightforward:

$$\begin{bmatrix} Y \\ C_b \\ C_r \end{bmatrix} = \begin{bmatrix} 16 \\ 128 \\ 128 \end{bmatrix} + \begin{bmatrix} 65.481 & 128.553 & 24.966 \\ -37.797 & -74.203 & 112.000 \\ 112.000 & -93.786 & -18.214 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \tag{1}$$

$$\begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} Y + 1.402 \cdot (C_r - 128) \\ Y - 0.34414 \cdot (C_b - 128) - 0.71414 \cdot (C_r - 128) \\ Y + 1.772 \cdot (C_b - 128) \end{bmatrix}. \tag{2}$$

2.2 Contrast Limited Adaptive Histogram Equalization (CLAHE)

Contrast Limited Adaptive Histogram Equalization (CLAHE) [18], is a well known CE algorithm, designed for broad applicability in the context of digital image processing. CLAHE is a variation of the *Adaptive Histogram Equalization (AHE)*[15], CE algorithm. In AHE, an image is processed transforming each pixel using a function based on the histogram of its surrounding pixels, defined by a *Contextual Region* $(\mathscr{R}_x, \mathscr{R}_y)$. CLAHE limits the CE by clipping the resultant histogram based in a coefficient called *Clip Limit* $\mathscr{C}$.

2.3 Multi-Objective Particle Swarm Optimization (MOPSO)

Multi-Objective Particle Swarm Optimization ($MOPSO$) [14] is a widely known metaheuristic algorithm. It is a bio-inspired metaheuristic which mimics the social behavior of bird flocking. In PSO, every potential solution of the problem being approached is called a *particle* and the actual population of solutions is called a *swarm*. Every particle $\vec{x}$ performs a search within a search space Ω, and for every generation t, every solution $\vec{x}$ is updated according to:

$$\vec{x}_i(t) = \vec{x}_i(t-1) + \vec{v}_i(t), \tag{3}$$

where $\vec{v}$ is a factor known as the velocity, and is given by:

$$\vec{v}_i(t) = w \cdot (t-1) + C_1 \cdot r_1 \cdot (\vec{x}_{p_i} - \vec{x}_i) + C_2 \cdot r_2 \cdot (\vec{x}_{g_i} - \vec{x_i}), \tag{4}$$

where $\vec{x}_{p_i}$ is the best solution that $\vec{x}_i$ has found so far, $\vec{x}_{g_i}$ is the best solution that the entire swarm has found at the current iteration, w is a coefficient known as the *inertia weight*, which controls the search speed rate of PSO; r_1 and r_2 are random numbers between $[0, 1]$. Finally, C_1 and C_2 are coefficient which control the weight between global and local particles during the search.

In $MOPSO$, a *constriction coefficient* χ is adopted in order to control the particle's velocity, as described below:

$$\chi = \frac{2}{2 - \varphi - \sqrt{\varphi^2 - 4\varphi}}, \tag{5}$$

where

$$\varphi = \begin{cases} C_1 + C_2 & \text{if } C_1 + C_2 > 4, \\ 0, & \text{if } C_1 + C_2 \leq 4. \end{cases} \tag{6}$$

Furthermore, the velocity in $MOPSO$ is bounded by the following *velocity constriction* equation:

$$v_{i,j}(t) = \begin{cases} delta_j & \text{if } v_{i,j}(t) > delta_j, \\ -delta_j, & \text{if } v_{i,j}(t) \leq delta_j, \\ v_{i,j}(t), & \text{otherwise,} \end{cases} \tag{7}$$

where

$$delta_j = \frac{upper\_limit_j - lower\_limit_j}{2}. \tag{8}$$

2.4 Entropy of Image

Entropy of image [9], is a metric that measures how much information is represented within an image. Entropy and contrast are closely related to the intensity distribution of images, so this metric is able to assess contrast variations as a consequence of image transformations.

First, we need to define the *Histogram* of intensities of an image H as follows: Let $c_1, c_2, ..., c_n$ the count of pixels with intensity $i_1, i_2, ..., i_n$ respectively, and also let:

$$p_i = \frac{c_i}{N}, \qquad \sum_{i=1}^{n} c_i = N, \qquad i = 1, 2, ..., n, \tag{9}$$

where N is the total sum of pixels shown in an image I and n is every intensity level representable by the color space of I. Then H is defined as a probability distribution in which every p_i represents the probability of occurrence of an intensity i. Then, Entropy of Image is defined as below:

$$\mathscr{H} = -\sum_{i=0}^{n-1} p_i \log_2(p_i) \qquad \mathscr{H} \in \{0, ..., \log_2(n)\}. \tag{10}$$

2.5 Structural Similarity Index

The *Structural Similarity Index* ($SSIM$) [17] is a well known metric that measures important image's attributes such as *Luminance, Contrast* and *Structure*. SSIM main aim is to measure the distortion added to the image as a consecuence of the CE proccess. $SSIM$ is calculated by windows, so given two images I_x and T_y which represent an original and an enhanced image, respectively, the $SSIM$ index is defined as below:

$$SSIM(I, T) = \frac{(2\mu_{I_x}\mu_{T_y} + E_1)(2\sigma_{I_x T_y} + E_2)}{(\mu_{I_x}^2 + \mu_{T_y}^2 + E_1)(\sigma_{I_x}^2 + \sigma_{T_y}^2 + E_2)}, \qquad SSIM \in [0, 1], \tag{11}$$

where μ_{I_x}, μ_{T_y} is the intensity averages of I_x and T_y, respectively; $\sigma_{I_x}^2$ and $\sigma_{T_y}^2$ are the intensity variances for I_x and T_y, respectively; $\sigma_{I_x T_y}$ is the covariance between I_x and T_y intensities. $E_1 = (K_1 L^2)$, where L is the dynamic range of intensities of image's pixels, and $K_1 \ll 1$ is a small constant; $E_2 = (K_2 L)^2$, and $K_2 \ll 1$; both E_1 and E_2 are constants used to stabilize division when denominator is close to zero.

3 Formulation of the Problem

Given an color input image I, with $M \times N$ pixels, and a vector $\vec{x} = (\mathscr{R}_x, \mathscr{R}_y, \mathscr{C})$, where $\mathscr{R}_x$ and $\mathscr{R}_y$ are contextual regions and $\mathscr{C}$ is the *Clip Limit*, a set of non-dominated solutions $\mathscr{X}$, which simultaneously maximize the objective functions f_1, f_2, f_3, f_4:

$$\mathscr{F} = [f_1(I, \vec{x}), f_2(I, \vec{x}), f_3(I, \vec{x}), f_4(I, \vec{x})]; \qquad f_1, f_2, f_3, f_4 \in [0, 1], \tag{12}$$

where:

- T_y is the enhanced intensity map, when applying $\vec{x}$ to I_y; this is: $T_y = CLAHE(\vec{x}, I_y)$. T_y and I_y are the Y channel in the $YCbCr$ representation of I and T, respectively,
- $f_1(I, \vec{x}) = \frac{\mathscr{H}(T)}{\log_2 L}$ is the normalized Entropy of the enhanced intensity map T_y, as described above,
- $f_2(I, \vec{x}) = SSIM(I_R, T_R)$ is the $SSIM$ measure between I_R and T_R. I_R and T_R are the R channel of the RGB representation of I and T, respectively,
- $f_3(I, \vec{x}) = SSIM(I_G, T_G)$ is the $SSIM$ measure between I_G and T_G. I_G and T_G are the G channel of the RGB representation of I and T, respectively,
- $f_4(I, \vec{x}) = SSIM(I_B, T_B)$ is the $SSIM$ measure between I_B and T_B. I_B and T_B are the B channel of the RGB representation of I and T, respectively.

Bounded to:

- $\mathscr{R}_x \in [2, ..., M]$ for the $\mathbb{N}$ numbers,
- $\mathscr{R}_y \in [2, ..., N]$ for the $\mathbb{N}$ numbers,
- $\mathscr{C} \in (0, ..., 1]$ for the $\mathbb{R}$ numbers.

4 Proposal

Algorithm 1 MOPSO-CLAHE

Require: Input image I, amount of particles Ω, iterations t_{max}
Initialize ω, c_1, c_2, $t = 0$, $lower\_limit_1$, $lower\_limit_2$, $lower\_limit_3$, $upper\_limit_1$, $upper\_limit_2$, $upper\_limit_3$, $\mathscr{X}$
while $t < t_{max}$ **do**
  for every i-th particle **do**
    Calculate new velocity $\overrightarrow{v_i}^t$ of the particle using equations (4) and (7)
    Calculate new particle position $\overrightarrow{x_i}^t$ using expression (3)
    T = CLAHE($\overrightarrow{x_i}^t$, I)
    $f_i^t = f(I, \overrightarrow{x_i}^t)$
    if $\overrightarrow{x_i} \succ \overrightarrow{x_{p_i}}$ **then**
      replace $\overrightarrow{x}_{p_i}$ by $\overrightarrow{x_i}^t$
    end if
    if $\overrightarrow{x_i} \succ \overrightarrow{x_{g_i}}$ **then**
      Update the Pareto set $\mathscr{X}$
    end if
    $t = t + 1$
  end for
end while
Ensure: $\mathscr{X}$

Algorithm 1 shows how Color Multi-Objective PSO-CLAHE ($CMOPSO-CLAHE$) is implemented, in order to tune parameters of $CLAHE$. The parameters received by $CLAHE$ are stored by a particle $\vec{x} = (\mathscr{R}_x, \mathscr{Y}_x, \mathscr{C})$, the original image I is transformed to its $YCrCb$ representation,and $\vec{x}$ is applied to the Y channel, in order to obtain a Y_T intensity map, which is used to transform back to RGB, to obtain the resulting image T.

The resulting images are evaluated according to the metrics $\mathscr{H}_Y$, $SSIM_R$, $SSIM_G$, $SSIM_B$, which are the entropy of resulting images measured in the Y

channel of the $YCrCb$ representation of these, and $SSIM_R, SSIM_G, SSIM_B$ are the $SSIM$ measures for original and resulting images using the R, G, B channels of the RGB representations of these. The non-dominated solutions are then stored in the Pareto set. $CMOPSO - CLAHE$ proccess is repeated until a criterion stop is reached.

5 Results and Discussion

Table 1. Initial parameters for CMOPSO-CLAHE.

Parameter	Value	Parameter	Value
$lower\_limit_{\mathscr{R}_x}$	2	$upper\_limit_{\mathscr{R}_x}$	$M/2$
$lower\_limit_{\mathscr{R}_y}$	2	$upper\_limit_{\mathscr{R}_y}$	$N/2$
$lower\_limit_{\mathscr{C}}$	0	$upper\_limit_{\mathscr{C}}$	0.5
Ω	100	t_{max}	100
$c_1\ min$	1.5	$c_1\ max$	2.5
$c_2\ min$	1.5	$c_2\ max$	2.5
$r_1\ min$	0.0	$r_1\ max$	1.0
$r_2\ min$	0.0	$r_2\ max$	1.0

Tests were performed using 8 color images from the available dataset[1]. Table 1, shows how $SMPSO$ was configured for the tests. $SMPSO$ implementation is available at [3], meanwhile the implementations for $CLAHE$, $\mathscr{H}$ and $SSIM$ are available at [1]. For every test image, 50 test were performed, and 10 non-dominated solutions were found in average. From Figures (5,5,5), it is noticeable how CE is achieved; there is also a compromise relation between $\mathscr{H}$ and $SSIM_R, SSIM_G, SSIM_B$.

It is noteworthy from Figure (5) how higher values of $\mathscr{H}$ degrade images severely, so it is necessary to find the correct balance between $\mathscr{H}$ and $SSIM_R, SSIM_G, SSIM_B$. In Figure (5) it is shown the resultant image enhanced using the proposal described in [12]; it is noticeable that the resultant image does not achieve good CE; this is because the mono-objective approach does not use color information properly, and this result is the same for other test images.

In Table 3, the non-dominated metric coefficients are shown, and in the last line it is shown the metric coefficients for image (5), enhanced using the mono-objective proposal. Although its metrics seem to fall in the Pareto Front, the visual information obtained is not enough to state that the mono-objective proposal is feasible for color images. These results are similar for every test image used.

[1] http://www.vision.caltech.edu/archive.html

(a) Original Image. $\mathscr{H}_{\mathscr{Cy}} = 0.207231$, $SSIM_R = 1$, $SSIM_G = 1$, $SSIM_B = 1$.

(b) Enhanced Image. $\mathscr{H}_{\mathscr{Cy}} = 0.611275$, $SSIM_R = 0.00897331$, $SSIM_G = 0.00823064$, $SSIM_B = 0.00851013$.

(c) Enhanced Image. $\mathscr{H}_{\mathscr{Cy}} = 0.0350595$, $SSIM_R = 0.416776$, $SSIM_G = 0.403636$, $SSIM_B = 0.417654$.

(d) Enhanced Image using [12]. $\mathscr{H}_{\mathscr{Cy}} = 0.788927$, $SSIM_R = 0.000204143$, $SSIM_G = 0.0000526475$, $SSIM_B = 0.0000518143$.

Fig. 1. Original and resultant images of House 1.

Table 2. Correlation table between metrics. Data was taken from Table 3.

Metrics	$\mathscr{H}_{\mathscr{Cy}}$	$SSIM_R$	$SSIM_G$	$SSIM_B$
$\mathscr{H}_{\mathscr{Cy}}$	1			
$SSIM_R$	-0.9826	1		
$SSIM_G$	-0.9823	0.9999	1	
$SSIM_B$	-0.9826	0.9999	0.9999	1

Figure (2), shows the Pareto Front created from the data in Table 3, and also Table 2 shows the correlation between metrics, analyzed from the results in Table 3. It is remarkable that there is a strong positive correlation between $SSIM_R$, $SSIM_G$ and $SSIM_B$; and there is a negative correlation between the previously mentioned metrics and $\mathscr{H}_{\mathscr{Cy}}$.

These correlations indicate that the channels R, G, B of images are directly affected by the process that modifies Y channel (see Algorithm (1)). This also indicates that CE of color images can be posed as a bi-objective optimization problem, using only $\mathscr{H}_{\mathscr{Cy}}$ and $SSIM$ applied over Y channel.

Table 3. Metric coefficients obtained using our approach for some non-dominated results from image in Figure (1), and the coefficients obtained using the approach of [12], shown in the last line.

	$\mathscr{H}_{\mathscr{y}}$	$SSIM_R$	$SSIM_G$	$SSIM_B$
Result 1	0,455146	0,9844962	0,9859005	0,9850636
Result 2	0,341423	0,99448887	0,99505806	0,99470544
Result 3	0,9574285	0,605344	0,619333	0,60158
Result 4	0,9634576	0,598325	0,611372	0,597308
Result 5	0,9649405	0,583224	0,596364	0,582346
Result 6	0,388725	0,99102669	0,99176936	0,99148987
Result 7	0,9657106	0,579052	0,591965	0,578109
Result Mono	0,211073	0,999795857	0,9999473525	0,9999481857

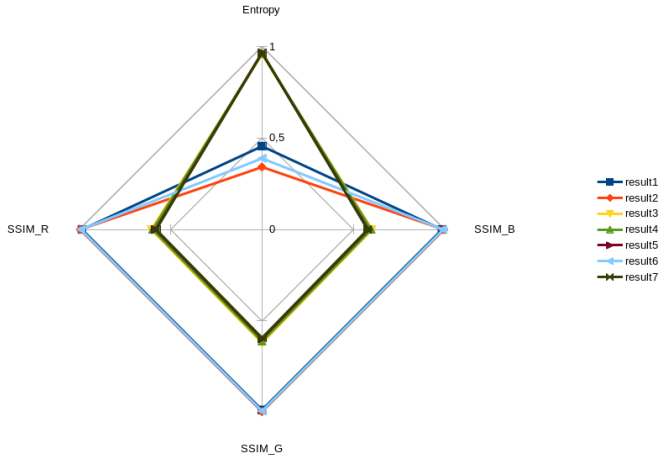


Fig. 2. Pareto front drawn using data from Table 3.

6 Conclusion

A Multi-Objective approach for Contrast Enhancement of color images is presented, which takes into account intensity and color information as Multi-Objective metrics. This approach achieves several resultant images, with different compromise rates between contrast and structural-similarity, in order to maximize information available for further analysis.

The authors are still performing test with similar images found in the database. As future work, it would be useful to analyze the parameters used for the meta-heuristics, the use of non-marginal metrics to assess the resultant images obtained with the approach, and perform tests posing CE of color images as a bi-objective optimization problem.

References

1. Bradski, G.: The opencv library. Dr. Dobb's Journal: Software Tools for the Professional Programmer 25(11), 120–123 (2000)
2. Doi, K.: Computer-aided diagnosis in medical imaging: historical review, current status and future potential. Computerized medical imaging and graphics 31(4), 198–211 (2007)
3. Durillo, J.J., Nebro, A.J., Alba, E.: The jmetal framework for multi-objective optimization: Design and architecture. In: Evolutionary Computation (CEC), 2010 IEEE Congress on. pp. 1–8. IEEE (2010)
4. Edelman, R.R., Warach, S.: Magnetic resonance imaging. New England Journal of Medicine 328(10), 708–716 (1993), http://dx.doi.org/10.1056/NEJM199303113281008, pMID: 8433731
5. Gonzalez, R.C., Woods, R.E.: Digital Image Processing (2nd Ed). Prentice Hall (2002)
6. Gonzalez, R.C., Woods, R.E.: Processing (2002)
7. Hoseini, P., Shayesteh, M.G.: Efficient contrast enhancement of images using hybrid ant colony optimisation, genetic algorithm, and simulated annealing. Digital Signal Processing 23(3), 879 – 893 (2013), http://www.sciencedirect.com/science/article/pii/S1051200412003107
8. Kak, A.C., Slaney, M.: Principles of computerized tomographic imaging. SIAM (2001)
9. Khellaf, A., Beghdadi, A., Dupoisot, H.: Entropic contrast enhancement. IEEE Transactions on Medical Imaging 10(4), 589–592 (Dec 1991)
10. Kim, Y.T.: Contrast enhancement using brightness preserving bi-histogram equalization. IEEE Transactions on Consumer Electronics 43(1), 1–8 (Feb 1997)
11. Lillesand, T., Kiefer, R.W., Chipman, J.: Remote sensing and image interpretation. John Wiley & Sons (2014)
12. Moré, L., Brizuela, M.: Pso applied to parameter tuning of clahe based on entropy and structural similarity index
13. More, L.G., Brizuela, M.A., Ayala, H.L., Pinto-Roa, D.P., Noguera, J.L.V.: Parameter tuning of clahe based on multi-objective optimization to achieve different contrast levels in medical images. In: Image Processing (ICIP), 2015 IEEE International Conference on. pp. 4644–4648. IEEE (2015)
14. Nebro, A.J., Durillo, J.J., Garcia-Nieto, J., Coello, C.C., Luna, F., Alba, E.: Smpso: A new pso-based metaheuristic for multi-objective optimization. In: Computational intelligence in miulti-criteria decision-making, 2009. mcdm'09. ieee symposium on. pp. 66–73. IEEE (2009)
15. Pizer, S.M., Amburn, E.P., Austin, J.D., Cromartie, R., Geselowitz, A., Greer, T., ter Haar Romeny, B., Zimmerman, J.B., Zuiderveld, K.: Adaptive histogram equalization and its variations. Computer vision, graphics, and image processing 39(3), 355–368 (1987)
16. Saitoh, F.: Image contrast enhancement using genetic algorithm. In: Systems, Man, and Cybernetics, 1999. IEEE SMC '99 Conference Proceedings. 1999 IEEE International Conference on. vol. 4, pp. 899–904 vol.4 (1999)
17. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image quality assessment: from error visibility to structural similarity. IEEE transactions on image processing 13(4), 600–612 (2004)
18. Zuiderveld, K.: Contrast limited adaptive histogram equalization. In: Graphics gems IV. pp. 474–485. Academic Press Professional, Inc. (1994)

ISSN 1870-4069

Signal Timing Model for Traffic Intersection Control

Diego Uribe[1], Enrique Cuan[1], Salvador Ibarra[2], Javier Diaz[3]

[1] Instituto Tecnológico de la Laguna,
División de Estudios de Posgrado e Investigación, Torreón, Coah.,
Mexico

[2] Universidad Autónoma de Tamaulipas,
Mexico

[3] Instituto Tecnológico de Celaya,
Mexico

diego@itlalaguna.edu.mx, kcuand@itlalaguna.edu.mx,
sibarram@docentes.uat.edu.mx, javier.diaz@itcelaya.edu.mx

Abstract. In this article we analyze the basic elements to be considered for the control of a traffic light by using modelling based on agents. More specifically, we model the factors that affect the way traffic engineers determine an appropriate sequence of signal indications at a busy intersection. In this model the user controls the flow of vehicles coming from each direction, the speed of the cars and the cycle-length at the traffic light. By trying different values of traffic flow while keeping the cycle-length fixed, the user can analyze the impact of the traffic flow in each direction under a particular cycle-length at the traffic signal. Also, this work makes clear how the use of agent-based modeling to simulate the control of a traffic light proves to be a useful tool for the analysis previous to the final installation of a traffic signal.

Keywords: Agent-based modeling, crossroad, intersection, traffic light, cycle-length, phase.

1 Introduction

It was in 1914 when the installation of the first traffic light in the city of Cleveland brought a significant improvement in the control of a transportation system. This visual signal device created for orderly managing the movement of vehicles was a watershed in the transportation industry: it offered us efficient and safe transportation. But nowadays, the vehicular explosion in big cities is the cause of many urban problems: wasted time as a consequence of traffic-related delays, poor fuel efficiency, high pollution, and unfortunately, an increase of people under stress [5]. So the need for a network of intelligent transportation system to be able to cope with these multiple problems is evident.

One of the most important activities of a traffic engineer is to measure the concentration of vehicles at points where two or more roads meet in order to avoid vehicle conflicts as well as to optimize the vehicle flow. Among the multiple strategies for handling traffic intersections, traffic light is the most common alternative to regulate vehicular traffic in all major cities around the world. One of the reasons why traffic signals are so commons is because they can be programmed for the orderly movement of vehicles and pedestrians. In other words, for a traffic engineer the control of the traffic signal is responsive to the demands of the traffic flow.

A vehicular traffic simulator plays a central role in the toolkit of a traffic engineer: it is an essential tool because it allows analysing hypothetical scenarios corresponding to several possible vehicular flows and the estimation of the cycle-length at the traffic light. In fact, by using a traffic simulator, a traffic engineer embarks on the pre-analysis of an urban area to be investigated in order to make decisions such as expanding the number of lanes, installing traffic lights, or altering the scheduled times of existing traffic lights. Thus, the purpose of using a traffic simulator is to establish various hypotheses to be verified, or disproved, based on the road traffic information obtained in real time.

According to the definition of a complex system as a system characterized by an environment in which multiple individual and independent elements interact with each other giving rise to an emerging phenomenon [6,8], to analyze vehicular traffic from this perspective is rather a plausible approach. In fact, taking into account the multiple factors to be considered when driving a vehicle (e,g. the presence of other vehicles, road conditions, traffic lights, traffic accidents, pedestrian crossing), a traffic system may also be characterized as a complex system.

In this way, since agent based modeling is a computational methodology that allows us to model complex systems [18], we analyze in this work the basic elements to be considered in vehicular traffic by using modeling based on agents. To be more specific, the vehicles and the traffic lights in our model are represented as agents, whereas the roads of the intersection are represented as the environment in which the vehicles travel and interact. Also, a set of basic parameters is defined to regulate the behavior of the agents and to observe the impact of different values on the interaction of multiple distributed elements.

For example, in this model the user controls the flow of vehicles coming from each direction, the speed of the cars and the cycle-length at the traffic light. By trying different values of traffic flow while keeping the cycle-length fixed, the user can analyze the impact of the traffic flow in each direction under a particular cycle-length at the traffic signal. Last but no least, this work makes clear how the use of agent-based modeling to simulate the control of a traffic light proves to be a useful tool for the analysis previous to the final installation of a traffic signal.

The theoretical framework of the methodology adopted in the traffic model is described in section 3. As it is briefly mentioned lines above, the model represents the simulation of a traffic intersection control based on agents so the properties,

behavior and interactions between autos and traffic signals are detailed. Then, the section 4 describes the impact of each parameter on the waiting time of cars through the intersection by adjusting some values while keeping fixed others when we run the simulator. But first we briefly review in the next section other works that have been developed for the control of a traffic intersection.

2 Related Work

In this section, we briefly narrate other models for describing traffic flows. Traffic flow theory is a scientific field relatively young. The first traffic problems in the past century were solved by the implementation of rule of thumb methods. However, at the beginning of the 1950s, the turning point in the field of traffic engineering came when Glen Wardrop described traffic flows by using mathematical representations [14].

The first popular traffic flow model, commonly known as the LWR model, was based on fluid dynamics, a discipline for studying fluids (liquids and gases) in motion [9,12]. In fact, traffic flow model makes use of numerical analysis and algorithms to solve and analyze problems that involved fluid flows. The interactions of liquids and gases are used as an analogy to model the interactions among vehicles. Another interesting work was the traffic model implemented by Prigogine and Herman who drew an analogy between gas-kinetic and the velocity distributions [11]. It is basically a stochastic model of traffic flow in which the "microgoals" of each driver are modified by the interaction with other drivers. Then, the central idea is to predict how much the "microgoals" of the driver have been modified by making use of a kinetic equation defined in terms of the desire of the driver and the interactions with other vehicles.

Among a myriad of interesting works about traffic control, Gershenson proposes the use of self-organizing methods as strategy for traffic signal control [4]. Since traffic densities change constantly, this research is based on the premise that self-organizing methods offer an adaptive alternative to traditional and rigid control methods. Traffic lights respond to car densities by implementing rules to give priority to cars waiting for a long period. As a self-organizing system is described as one in which elements interact in order to achieve a global function, in this work we can see how cars and traffic lights interact to share the control of the signals for improving traffic flow.

Another interesting adaptive intersection control system is based on cooperative conflict resolution techniques. Specifically, Ball and Dulay proposed a distributed approach using vehicles as intelligent objects to encourage interaction to safely and efficiently regulate intersections [1]. By using vehicles capable of both sensing and actuating within their local space, they are able to communicate with each other using wireless technology. Thus, vehicles cooperate using ad-hoc messaging to share journey plans to safely travel over a shared intersection. In this way, the intersection control system contributes to a plausible reduction in the number of start-stops and the total travel distance.

3 Theoretical Framework

In this section we describe the theoretical framework that supports the basic signal timing model to be analyzed in this article. First, we define a set of essential terms to be considered in the design of a proper timing of the signals for the efficient operation of the traffic light. Then, we briefly present the staple concepts of agent-based modelling, the methodology for designing and building the signal timing model.

3.1 Signal Timing Concepts

Intersections, places where two or more roads join or cross each other, can be divided in terms of their form and their handling or operation. Since the control or operation of the intersection is the research topic of this paper, different strategies for the control of conflicting traffic movements are available to the engineer: priority control, space-sharing, time-sharing and grade-separated [10].

Priority intersections denote a crossroad where the minor road traffic enters the main road stream during spare time gaps. The access to the main road is normally controlled by stop or give way signs and markings on the minor road. The principal advantage of this type of intersection is that the stream of vehicles on the main road are not delayed.

Space-sharing intersections are commonly known as roundabout intersections. These particular type of intersections denote a place where three or more roads join and traffic must go around a circular area in the middle, rather than straight across. The access to the intersection is normally controlled by give way markings since vehicles circulating in the roundabout have priority.

Grade-separated intersections are also known as interchange intersections. These particular type of intersections denote a system of connector roads for the interchange of vehicular traffic between two or more roads on different levels. The access to the main stream is through the use of connection roads known as slip roads. The principal advantage of this type of intersection is that minimizes the number of conflict points.

Time sharing intersections represent the type of intersection to analyze in this paper. These particular type of intersections operate on the basis that separate time periods are allotted to conflicting traffic streams so that each can make safe and efficient use of he intersection at different times. Since the operation of this intersection relies on separate time periods for the orderly movements of vehicles, the definition of a proper timing of the traffic light signals is imperative. In fact, the definition of the intervals of a traffic signal is based on a set of essential terms to be described next [2,13]:

- *Cycle*: it is defined as a complete color sequence of signal indications. For example, the sequence green-yellow-red denotes a cycle.
- *Cycle length*: the time required to complete a cycle is defined as cycle length. For example, the cycle length is the time that elapses from the start of the green indication to the end of the red indication.

– *Phase*: set of traffic streams having the right-of-way simultaneously before the release of another conflicting set of movements at an intersection.
– *Interval*: a period of time during which no signal indication changes.

The cycle has four main components:

– *Green interval*: a period of time during which a set of traffic streams can effectively move. All other movements have red indication.
– *Yellow interval*: a period of time required for a driver traveling on a particular approach to realize the permission for movement is about to end so the driver can safely stop or safely go through the intersection.
– *Clearance interval*: this is a red indication provided after the yellow interval to give additional time to those vehicles that enter the intersection on yellow before conflicting flows are released. It is also called the "all red interval."
– *Red interval*: a period of time during which a set of traffic streams are not allowed to move while other conflicting flow streams are allowed.

For the sake of illustration of the above terms and concepts, Figure 1 shows a simple four-arm intersection where two approaches meet: East/West and North/South. Figure 1 also shows traffic movements in a two phase signal system with straight traffic without left turning movements. Since East and West streams are non-conflicting movements, they are grouped in a single phase, whereas the non-conflicting North and South streams are grouped in a second phase.

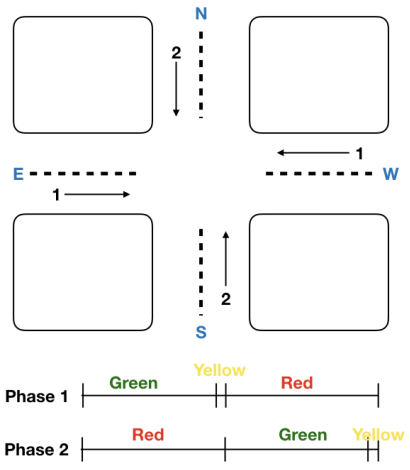


Fig. 1. Two-phase signal system.

Once the number of phases in the intersection have been defined, as well as the yellow and clearance interval together with the cycle length, an adaptable signal timing light can be programmed. First, the critical volume for each each

phase and the total critical volume are determined. The critical volume is the maximum of the flows corresponding to each phase (cvp), whereas the total critical volume (cvt) is just the sum of the critical volumes.

$$cvt = \sum_{i=1}^{n} (cvp_i), \tag{1}$$

where n denotes the number of phases and

$$cvp_i = \max(f_i), \tag{2}$$

where f_i denotes a particular flow or volume for p phase.

Second, the green time for each phase is calculated. Basically, the green interval represents the proportion of the cycle length to be allocated to each phase. Thus, the length of the green period for each phase (gp_i) is obtained from:

$$gp_i = \frac{cvp_i}{cvt} \times cl, \tag{3}$$

where cl denotes the cycle length.

Finally, the red interval for each phase is calculated in terms of the yellow and clearance interval as well as the green interval corresponding to the rest of the phases.

3.2 Agent-based Modeling

Agent based modeling is a computational methodology that allows us to model complex systems [18]. A complex system is defined as a system characterized by an environment in which multiple individual and independent elements interact with each other giving rise to an emerging phenomenon [6] [8]. ABM is a computational modeling paradigm that enable us to describe a complex system in terms of agents, environment, and interactions. While agents denote the basic ontological unit of the model, the environment represents the world in which the agent lives. In this work, we make use of agent based modeling to represent the interaction of vehicular flows concurring in a traffic intersection. To be more specific, vehicles and traffic signals are represented as agents, whereas the junction of the streets is represented as the environment in which the vehicles travel and interact. Simple functions such as acceleration and breaking characterize the vehicles behavior, whereas to change the signals according to a fixed or variable strategy characterises the behavior of a traffic light.

4 Signal Timing Simulations and Discussion

Once the basic concepts to be considered in the design of a proper signal control have been described, we present the experimentation implemented with two multi-agent models that represent the two principal types of signal control: pre-timed and actuated. As these agent-based models enable us to analyze

the range of behaviors exhibited by multiple simulations, we first present the results obtained from fixed signal timing simulations, and then, the results obtained from adaptable signal timing simulations. In both cases, we describe the parameters of the model, as they are used as the primary tuning knobs to determine particular settings for specific vehicular flows concurring in the intersection.

Now, in order to analyze the simulation results, we need to measure how effective the settings of traffic signals are. As one of the primary functions of traffic control signals is to reduce the vehicular wait for the right-of-way, delay is the most common measure of effectiveness for signalized intersections [15]. Thus, when we run one of the models, the computer simulation provides the overall number of stopped cars. By counting the number of vehicles in the queue at fixed intervals of time, the traffic engineer is able to measure the effectiveness of a particular configuration of the model's parameters. We now proceed to the description of the results obtained from fixed signal timing simulations.

The intersection control models used to illustrate the types of traffic signal control have been developed with NetLogo [17]. NetLogo is a programming language used to create models based on agents and has also proved to be a well suited tool for modeling complex systems evolving over time. One of the most appealing features of NetLogo is the capacity to investigate the connections between the micro-level behavior of individuals and macro-level patterns that emerge from their interactions [7].

4.1 Fixed Signal Timing Simulations

In this case, we extended the traffic intersection model presented by Wilensky [16]. Such model represents a 2-phase independent intersection with a pre-timed control characterized by a repetitive cycle and split timing. Figure 2 shows the context in which, on the left side, three purple buttons allow to establish the scenario (setup) and execute the model (go and go once). Likewise, a set of green sliders allow the configuration of the traffic conditions to be simulated. The first three sliders concern the control of the speed of the vehicles (speed-limit, max-accel and max-brake), the next two sliders determine the traffic volume of each approach (flow-North and flow-East), and the last two sliders allow to set up the time in seconds corresponding to the green and yellow interval respectively.

The results shown in Table 1 were obtained for a particular green interval (16) and three different values of traffic volume distribution (60, 80 and 100) for the North approach, whereas the flow for the East approach was fixed at 100. Initially, when we have a traffic volume of 60 at the North approach, we are simulating an intersection where the North approach represents a minor road and the East approach denotes the major street. When we increase the traffic volume at the North approach with a value of 80 we still have an intersection with a minor and major street. However, when we finally increase the traffic volume at the North approach with a value of 100, we are simulating an intersection of streets with equal traffic volumes.

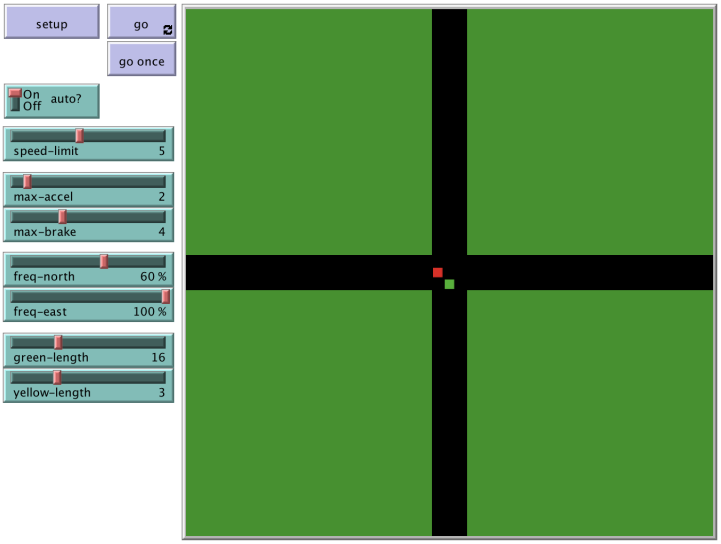


Fig. 2. Intersection-green-2-phases.

Another important point to be noticed in the Table 1 is the column labeled as Ticks. This column represents the time in seconds to run each simulation and the values in parenthesis denotes the number of cycle length to be observed. For example, 600 corresponds to 15 times 40 where 15 denotes the number of signal cycles and 40 represents the value of the cycle length when the green interval is 16, yellow interval is 3 and clearance interval is 1. Thus, we run simulations under complete rotations of the signals for each approach so the same number of color signals was allotted to each street. The upshot of this small but significative extension of the model has been to be able to more precisely measure the number of stopped cars and therefore, to understand the circumstances under which a fixed signal control is appropriated.

We now see how, as long as the traffic volume on the North approach (values of 60 and 80) is less than the traffic volume on the East approach (values of 100), the number of stopped cars on the major street (East road) is higher than on the minor street (North road). As a consequence of this traffic distribution, traffic on the major street forms into platoons of slow-moving vehicles which may stop and start (denoted by values greater than one). Thus, the green interval for the traffic volume on the major street is not enough. On the other hand, as the concentration of traffic on the North approach is low, drivers on the minor road are not disturbed by additional delays at the traffic light (denoted by values lower than one). In other words, the green interval for the traffic volume on the minor street is sufficient.

Now, when we finally increase the traffic volume at the North approach with a value of 100, we are simulating an intersection of streets with equal traffic volumes. Since the traffic distribution at both overlapping approaches is high

Table 1. Signal-fixed 16.

		Flow of cars		Stopped cars			
Traffic Distribution	**Ticks**	**East**	**North**	**East**	**North**	**%East**	**%North**
60 North - 100 East	600 (15)	229	136	414	42	1.81	0.31
	1200 (30)	443	264	840	86	1.90	0.33
	2400 (60)	862	516	1696	162	1.97	0.31
80 North - 100 East	600 (15)	215	171	440	70	2.05	0.41
	1200 (30)	420	336	869	99	2.07	0.29
	2400 (60)	841	672	1718	206	2.04	0.31
100 North - 100 East	600 (15)	223	231	419	413	1.88	1.79
	1200 (30)	423	443	872	849	2.06	1.92
	2400 (60)	848	863	1718	1695	2.03	1.96

and alike, the number of stopped cars is also similar. By running the model we can see how platoons of slow-moving vehicles are formed at both streets and also disturbed by delays. As the simulation results show, the green interval for high traffic volume on both streets is not enough: a substantial increase on the green interval is recommended.

4.2 Adaptable Signal Timing Simulations

In this case, our signal control model represents a 2-phase independent intersection based on actuated control characterized by responsive operation. While the preceding model assigns the right of way according to previous records of traffic demands, the right of way in this model is based on current traffic conditions. As in the preceding model the green interval for each phase is predefined, in this model the green interval for each phase is defined on the basis of ongoing traffic conditions. In other words, the actuated control implemented in this model reproduces a signal control more adaptable to changing traffic volumes.

Figure 3 shows the scenario in which this model is portrayed. As we can see, Figures 2 and 3 are very similar. In both figures the sliders and controls are almost the same except that we have removed from Figure 2 the last two sliders to set up the green and yellow interval respectively. Instead of these two sliders, a new slider has been defined for the customization of the cycle length. In this way, this slider allows to set up the time in seconds to complete one full cycle of all signal indications.

The columns labeled as Green-East and Green-North in Table 2 represent the distribution of the green interval for each approach. Said in another way, these

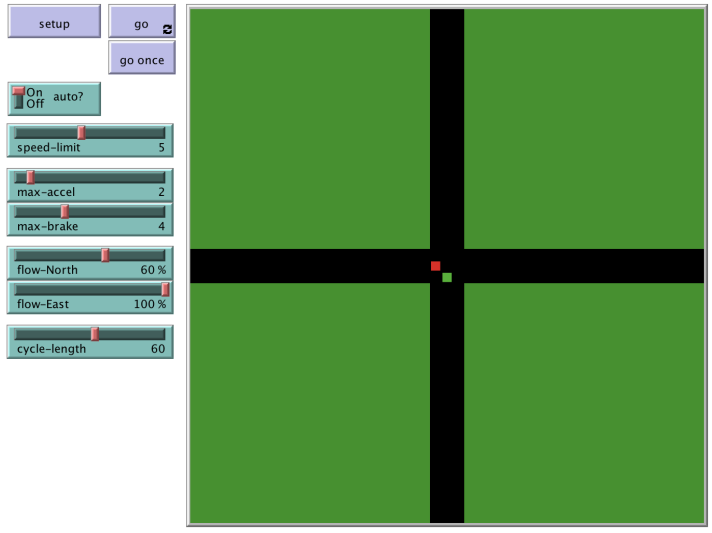


Fig. 3. Intersection-cycle-2-phases.

columns illustrate how the green interval for each phase is based on current traffic conditions. The values of these columns have been obtained from the equation 3 to calculate the green interval in terms of the cycle-length. For example, for a cycle-length of 32, when the traffic distribution in the intersection is 60% on North and 100% on East, the green values are 12 and 20 respectively. Thus, we have in this model a more efficient distribution of the green interval for each phase: at higher traffic volume, higher green interval.

However, when the traffic volume is increased to 80% on the North approach, the representation of the intersection as junction of a minor street and a major street is not so obvious: the difference in traffic volume between the streets has now decreased. So in this new traffic distribution, and under the same cycle-length, the green intervals obtained are 14 and 18 respectively. Even though a higher green interval has been allotted to the approach with higher volume, the green interval for the major street has been decreased. And lastly, when the traffic volume is increased to 100% on the North approach, we are simulating an intersection of streets with equal traffic distribution. In this case, the green intervals obtained are 16 and 16 respectively. Since the traffic volume on each approach is identical, the green interval assigned for each approach is exactly the same.

Table 2 also shows the number of stopped cars for each traffic distribution. From one perspective, the number of stopped cars are similar to the values shown in Table 1: as long as the traffic volume on the North approach (values of 60 and 80) is less than the traffic volume on the East approach, the number of stopped cars on the major street (East road) is higher than the number of stopped cars on the minor street (North road). However, an important difference is worth to

Table 2. Cycle-length 32.

		Flow of cars		Stopped cars				Green	
Traffic Distribution	**Ticks**	**East**	**North**	**East**	**North**	**%East**	**%North**	**East**	**North**
60 North - 100 East	600 (15)	272	162	415	186	1.53	1.15	20	12
	1200 (30)	532	319	807	298	1.52	0.93		
	2400 (60)	1046	627	1647	530	1.57	0.85		
80 North - 100 East	600 (15)	248	196	414	312	1.67	1.59	18	14
	1200 (30)	480	383	833	565	1.74	1.48		
	2400 (60)	941	752	1674	894	1.78	1.19		
100 North - 100 East	600 (15)	223	231	419	413	1.88	1.79	16	16
	1200 (30)	423	443	872	849	2.06	1.92		
	2400 (60)	848	863	1718	1695	2.03	1.96		

be noticed: the green interval assigned for the traffic volume on both streets is not enough. And of course, things getting worst when the traffic volume on both streets is equal. Platoons of slow-moving vehicles are formed around both streets and also disturbed by a delay. Thus, a substantial increase on the cycle length is recommended.

4.3 Discussion

In the previous section, the results of the simulations have shown the performance of two different signal control methods. The results were analyzed by comparing different traffic distributions for each signal control model. Now, in this section we examine the results by comparing the same traffic distributions under two different signal control methods. In other words, instead of analyzing the values within a particular table of simulations, we now inspect the values between the tables of different signal control results.

We look at Table 1 and Table 2 how, as long as the traffic volume on the North approach (values of 60 and 80) is less than the traffic volume on the East approach (value of 100), the number of stopped cars on the major street (East) is higher than the number of stopped cars on the minor street (North). However, even tough this uniformity is observed in both signal control methods, two important differences must be noticed. First, the number of stopped cars on the East approach in Table 2 is less than the corresponding value in Table 1. Second, the number of stopped cars on the North approach in Table 2 is greater than the corresponding value in Table 1. What does this mean? On the one hand, the adaptable signal control decreases the delay on the East approach, but on the other hand the delay has been increased on the North approach.

Even though the problem of density still exists on the major road, the formation of platoons starts to arise on the minor road.

Now, when the traffic distribution on both streets is equal, the problem of density has been extended to both roads. By simulating a traffic intersection of streets with equal traffic volumes we clearly see how the number of stopped cars on the North approach in Table 1 is similar to the corresponding value in Table 2. And we can observe the same regularity on the East approach. Thus, a plausible option to avoid platoons of slow-moving vehicles around both streets is a substantial increase on the current cycle length.

To verify this assumption, Table 3 shows the results of the simulation for a cycle length of 48. Regardless of the traffic distribution, a substantial decrease in the number of stopped cars on both streets is observed when we compare the corresponding values of Table 2 and Table 3. For example, when the traffic volume in the intersection is represented as a junction of a minor and a major street the vehicles move without delay. Even when traffic volume on both streets is equal the vehicles do not experiment a disturbing delay. In this way, an increase on the cycle length from 32 to 48 has proved to be a good alternative for a less congested flow.

Table 3. Cycle-length 48.

		Flow of cars		Stopped cars				Green	
Traffic Distribution	**Ticks**	**East**	**North**	**East**	**North**	**%East**	**%North**	**East**	**North**
60 North - 100 East	840 (15)	385	231	407	185	1.06	0.80	30	18
	1680 (30)	753	449	814	303	1.08	0.67		
	3360 (60)	1501	898	1625	634	1.08	0.71		
80 North - 100 East	840 (15)	356	281	406	270	1.14	0.96	27	21
	1680 (30)	697	554	809	672	1.16	1.21		
	3360 (60)	1373	1089	1638	1476	1.19	1.36		
100 North - 100 East	840 (15)	315	319	416	407	1.32	1.28	24	24
	1200 (30)	615	626	830	821	1.35	1.31		
	3360 (60)	1228	1238	1662	1664	1.35	1.34		

According to the results of multiple simulations, we can clearly see what conditions are more appropriated for a particular signal control method. As Gartner claims each signal control method has its unique advantages and disadvantages [3]. Since our experimentation was carried out to analyze the implications of multiple traffic demands, we were able to verify why a fixed signal control is more appropriated when we have a predictable traffic demand. In our case, when the intersection is represented as a junction of a minor and a major street, that

is, and the traffic flow on the North approach is significantly less than the East approach, a fixed green interval of [16 - 18] for the North approach is enough. As for the major street, that is, the East approach, a fixed green interval of [28-30] is acceptable.

However, when the traffic distribution on both streets of the intersection is almost equal, an adaptable signal control is the best option. Since the traffic volume in a particular street might be higher than the other one in a specific moment, and it might also be lower a few moments later, the traffic current conditions are crucial for setting the time interval for each signal indication. In our case, when the traffic volume on both streets is similar, we tried two cycle-lengths: 32 and 48. As we can see in Table 3, an adaptable and better distribution of the green intervals has been obtained whit a cycle length of 48. As we previously said, a cycle length of 48 has proved to be a good alternative for a less congested flow.

5 Conclusions and Future Work

This paper has investigated the basic elements to be considered for the control of a traffic light by using modelling based on agents. We focus our attention in how the flow of vehicles in each street of the junction influences the control of a traffic light. By trying different values of traffic flow while keeping the cycle-length fixed, the user can analyze the impact of the traffic flow in each direction under a particular cycle-length at the traffic signal. In this way, the traffic engineer is able to verify hypothesis as to how a particular configuration of the pertinent parameters works.

There are two directions in which our model could be improved: to increase the number of phases in the traffic junction and the coordinated operation of two or more signalized intersections. To extend the model to not only increase the number of phases but also include left turning movements sounds rather interesting. Likewise, to extend the model to consider parameters such as the distance between two or more signals and the required offsets to coordinate the operation of adjacent intersections on the same cycle length sounds also appealing. To work in these directions will contribute to the robustness of our signal timing model.

References

1. Ball, R., Dulay, N.: Enhancing traffic intersection control with intelligent objects. In: Urban Internet of Things Towards Programmable Realtime Cities (2010)
2. Garber, N., Hoel, L.: Traffic and Highway Engineering. Cengage Learning (2014)
3. Gartner, N.H.: Traffic management and control (July 2013), `www.victorknoop.eu/isttt/Tutorials/gartner/TMC4_Freeways.pdf`
4. Gershenson, C.: Self-organizing traffic lights. Complex Systems 16(1), 29–53 (2004)
5. Greengard, S.: Smart transportation networks drive gains. Communications of the ACM 58(1), 25–27 (2015)

6. Holland, J.: Hidden Order: How Adaptation Builds Complexity. Helix Books/Addison-Wesley (1995)
7. Janota, A., Rastocny, K.: Multi-agent approach to traffic simulation in netlogo environment level crossing model. In: 5th International Conference Transport Systems Telematics TST05- 2005 (2005)
8. Kauffman, S.: At Home in the Universe: The Search for the Laws of Self-Organization and Complexity. Oxford University Press (1995)
9. Lighthill, M., Whitham, G.: On kinematic waves: Ii. a theory of traffic flow on long crowded roads. In: Proceedings of the Royal Society. vol. A229, pp. 317–345 (1955)
10. O'Flaherty, C.: Transport Planning and Traffic Engineering. Taylor & Francis (1996)
11. Prigogine, I., Herman, R.: Kinetic Theory of Vehicular Traffic. Elsevier, New York (1971)
12. Richards, P.I.: Shockwaves on the highway. Operations Research 4, 42–51 (1956)
13. Roess, R., Prassas, E., McShane, W.: Traffic Engineering. Pearson/Prentice Hall, 4th edn. (2010)
14. Wardrop, J.: Some theoretical aspects of road traffic research. In: Proceedings of the Institution of Civil Engineers. vol. 1 (1952)
15. Webster, F.: Traffic signal settings. Road Research Laboratory, Department of Scientific and Industrial Research (1958)
16. Wilensky, U.: Netlogo traffic intersection model. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL (1998), `http://ccl.northwestern.edu/netlogo/models/TrafficIntersection`
17. Wilensky, U.: Netlogo. Center for Connected Learning and Computer-Based Modeling, Northwestern University, Evanston, IL (1999), `http://ccl.northwestern.edu/netlogo/`
18. Wilensky, U., Rand, W.: An Introduction to Agent-Based Modeling. The MIT Press. Cambridge Massachusetts (2015)

ISSN 1870-4069

Reconocimiento de objetos cuasi-planos mediante un sistema de tratamiento digital de imágenes embebido en una plataforma tipo Raspberry Pi

Julián Jerónimo[1], Humberto Sossa[2,3]

[1] Instituto Tecnológico Superior de Coatzacoalcos, Coatzacoalcos, Veracruz, México

[2] Instituto Politécnico Nacional, CIC, Ciudad de México, México

[3] Tecnológico de Monterrey, Zapopan, Jal., México

julianjb@protonmail.com, humbertososa@gamil.com

Resumen. En este trabajo se describe un sistema embebido para el reconocimiento de objetos cuasi-planos. El sistema combina técnicas estándar de tratamiento digital de imágenes, lo que permite describir cada objeto a través de un vector de atributos numéricos. Un objeto detectado a través de la cámara del sistema es reconocido mediante un clasificador típico de distancia euclidiana. El sistema completo es embebido en una plataforma tipo Raspberry Pi con una cámara conectada a él. El sistema es probado en varios escenarios.

Palabras clave: Reconocimiento de objetos, sistemas embebidos, momentos invariantes.

Recognition of Quasi-plane Objects Using Digital Image Processing System Embedded in a Platform of Raspberry Pi Type

Abstract. In this work, an embedded system for the recognition of quasi-plane objects is described. The system combines standard techniques of digital image processing, which allows describing each object through a vector of numerical attributes. An object detected through the system's camera is recognized by a typical Euclidean distance classifier. The complete system is embedded in a platform of Raspberry Pi type with a camera connected to it. The system is tested in several scenarios.

Keywords: Recognition of objects, embedded systems, invariant moments.

1. Introducción

El desarrollo de aplicaciones de reconocimiento de objetos mediante tratamiento digital de imágenes embebidas en sistemas de bajo consumo, plantea retos importantes a la comunidad de visión artificial. Las características limitadas en el hardware con respecto a la capacidad de procesamiento, la cantidad de memoria y el número reducido de puertos en estos sistemas complican el uso de técnicas de reconocimiento de objetos en tiempo real debido a su alto costo de procesamiento y al hardware especializado que estas requieren. Por lo tanto, la implementación de sistemas de reconocimiento de objetos en tiempo real en plataformas embebidas requiere de un importante nivel de eficiencia y un adecuado aprovechamiento de las capacidades de los componentes del hardware.

A pesar de las evidentes carencias que este tipo de sistemas presentan, en comparación con los equipos de cómputo convencionales, cuentan con una ventaja importante que ha llevado a aumentar su popularidad en las comunidades de desarrolladores: son sistemas relativamente económicos y fáciles de adquirir. De esta manera, un sistema de reconocimiento de objetos embebido en una plataforma Raspberry Pi 3 modelo B conectado a una cámara para procesamiento en tiempo real puede ser desarrollado con una inversión cercana a los 100 dólares, la cual está muy por debajo del costo promedio de un equipo de cómputo convencional.

Este tipo de sistemas de reconocimiento de objetos embebidos en plataformas tipo Raspberry Pi ofrecen una solución al problema de la falta de equipo en instituciones educativas de nivel medio superior, superior y posgrado, las cuales requieren de equipos de cómputo para el aprendizaje de conceptos básicos de técnicas procesamiento de imágenes y visión artificial por parte de los alumnos, sin la necesidad de realizar una inversión elevada que pueda afectar el presupuesto de estas instituciones. Asimismo, otra ventaja de este tipo de sistemas es su facilidad de uso y su tamaño reducido, lo cual representa una opción ideal para su uso en actividades educativas, ya que son menos propensos a presentar alguna falla en sus componentes durante su manipulación.

Un ejemplo de una implementación de reconocimiento de objetos se describe en [1]. En este trabajo, los autores utilizan marcas de referencia y segmentación de color, así como la detección de contornos. Una de las limitantes que presenta dicha implementación es que no puede procesar más de un objeto a la vez, además, dicho objeto debe aparecer en una posición determinada de acuerdo a las marcas de referencia utilizadas, de lo contrario el reconocimiento no funciona.

En este trabajo se describe un sistema de reconocimiento embebido en una plataforma tipo Raspberry Pi mediante una cámara conectada, el cual es compatible con múltiples clases de objetos cuasi-planos procesados simultáneamente en tiempo real. Para el desarrollo de este sistema se utilizó un conjunto de técnicas de procesamiento de imágenes, comenzando por la medición del valor promedio del color en el cuadro procesado como se propone en [2] que permita la elección adecuada del método de umbralado como se muestra en [3] y [4]. El resultado de este método adaptativo permite generar regiones de interés utilizando un análisis de componentes conectados como se ilustra en [5]. De las regiones de interés se realiza el cálculo de los momentos centrales normalizados como se menciona en [6]. A partir de estos

momentos se obtienen los momentos invariantes de Hu como se hace en [7]. Para determinar la clase de pertenencia de un objeto, se utiliza un clasificador de distancia euclidiana, el cual, ofrece una solución efectiva para el problema en cuestión como se describe en [8].

El resto del artículo está organizado de la siguiente forma: en la sección 2 se presenta el desarrollo del sistema propuesto. Los resultados obtenidos a partir del funcionamiento del sistema propuesto y la medición de su desempeño se muestran en la sección 3. Finalmente, en la sección 4 se mencionan las conclusiones alcanzadas y las propuestas de trabajo a futuro.

2. Desarrollo

El desarrollo del sistema se realizó a lo largo de tres etapas: (1) definición de una metodología para la detección, clasificación y reconocimiento de objetos; (2) instalación y configuración de la plataforma tipo Raspberry Pi; y (3) ejecución en tiempo real de la implementación. Cada etapa que compone al proceso de desarrollo del sistema se describe a continuación:

2.1. Metodología de detección, clasificación y reconocimiento de objetos

Sin pérdida de generalidad, para probar la eficiencia y eficacia del sistema se seleccionaron cinco clases de objetos (ver Fig. 1). No obstante, es posible definir nuevas clases de objetos realizando un entrenamiento el cual consiste en la evaluación del promedio de la sumatoria del primer momento invariante de Hu para un conjunto determinado de imágenes en donde se encuentre el objeto que se pretende definir. Se ha escogido el primer momento de Hu por ser el más representativo en aplicaciones de reconocimiento de formas [1,6].

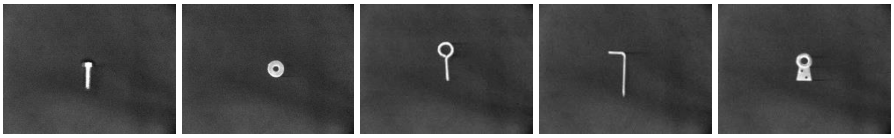


Fig. 1. Las cinco clases de objetos propuestos, de izquierda a derecha: tornillo, rondana, alcayata, armella y cola de milano.

La ejecución del sistema propuesto se realizó a partir de la captura de video en tiempo real por medio de una cámara de ocho megapíxeles conectada a la plataforma Raspberry Pi. El número de fotogramas por segundo procesados por la cámara es de 32, lo que equivale a 32 cuadros o imágenes procesadas por el sistema por cada segundo de video. Para disminuir el costo de procesamiento en la plataforma de la Raspberry Pi se estableció manualmente una resolución de imagen de 320×240 píxeles y un tiempo de retraso para preparar la cámara de 1 milisegundo de duración. Cada cuadro procesado por la cámara fue convertido a un arreglo multidimensional para acceder a la información de los píxeles que lo componen y de esta manera poder realizar el

reconocimiento de objetos. Las etapas que componen la metodología del sistema pueden observarse en el diagrama de la Fig. 2. Cada una de estas etapas se explican enseguida.

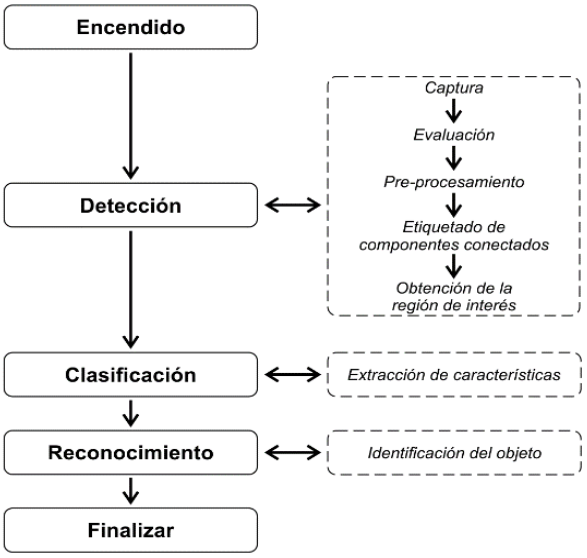


Fig. 2. Etapas de la metodología del sistema propuesto.

Encendido. Al encender la plataforma de la Raspberry Pi las siguientes tareas son realizadas de manera automática: (1) se inicia el sistema operativo, y (2) se cargan las dependencias utilizadas por el sistema. El usuario ejecuta manualmente el sistema a través de un archivo de procesamiento de lotes (comúnmente llamado archivo `script`), el cual se encuentra embebido en la plataforma de la Raspberry Pi. El sistema verifica que la cámara se encuentre presente y no esté siendo ocupada por ningún otro proceso, tras esto el sistema comienza a ejecutarse.

Detección. De acuerdo a la Fig. 2, esta etapa consta de los siguientes cinco pasos.

Captura: Al iniciar el programa, se obtiene el cuadro actual capturado por la cámara de la Raspberry Pi y se prepara para ser procesado por el sistema.

Evaluación: Se identifican los componentes candidatos que pertenecen a alguna clase de objeto. Para esto, el sistema calcula el valor promedio del color del cuadro procesado por la cámara de acuerdo a los valores presentes en los canales BGR (Blue, Green, Red) que utiliza por defecto la biblioteca de OpenCV. El valor promedio del color de la imagen viene dado por la sumatoria del valor promedio del color de cada fila de píxeles en la imagen. Debido a que no es necesario conocer el valor promedio de cada canal para determinar el ruido presente en la imagen, el valor obtenido de la sumatoria de las filas de píxeles se redondea para seleccionar el método de umbralado más conveniente para el tipo de imagen según el nivel de ruido presente.

Pre-procesamiento: De acuerdo al nivel de ruido presente en la imagen el sistema elige la técnica de umbralado adecuada para realizar la detección de componentes. En el caso de una imagen con un valor promedio superior a 90 se aplica un umbralado adaptativo, caso contrario en una imagen con un valor promedio igual o inferior a 90 en donde se aplica un umbralado manual. Ambos escenarios se muestran en la Fig. 3.

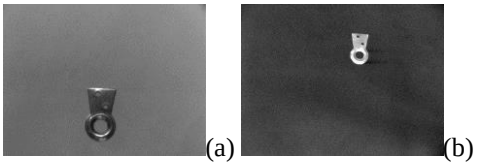


Fig. 3. (a) Imagen con valor promedio superior a 90; (b) Imagen con valor promedio igual o inferior a 90.

Etiquetado de componentes conectados: A continuación se realiza el etiquetado de los componentes conectados sobre la imagen umbralada. Mediante esta técnica, como se sabe, el sistema asigna de forma automática etiquetas a cada componente encontrado en el cuadro procesado por la cámara de la plataforma de la Raspberry Pi. El proceso del etiquetado de componentes conectados parte de la idea de analizar cada píxel dentro de una determinada fila en la imagen, repitiendo el proceso por cada una de las filas de píxeles que componen a la imagen, asignando etiquetas sucesivas de acuerdo al número de componentes conectados encontrados.

Obtención de la región de interés: Para cada componente conectado se define una región de interés a través de una caja delimitadora cuyas dimensiones toman como base a la altura y la anchura del componente, así como sus coordenadas en el eje X y en el eje Y. Cada región de interés generada representa un objeto potencial perteneciente a alguna de las clases descritas durante el entrenamiento. En la Fig. 4 se puede ver cómo se ha logrado realizar correctamente la separación de los cuatro objetos en la imagen.

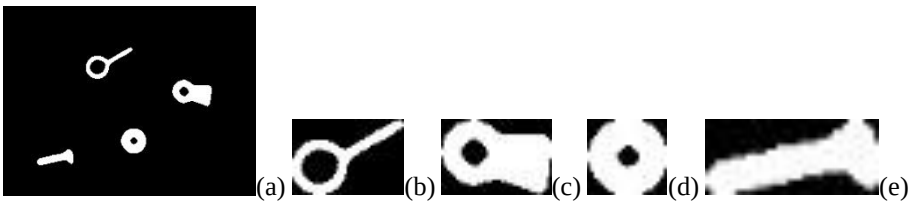


Fig. 4. (a) Imagen original con los componentes conectados; (b), (c), (d) y (e) regiones de interés generadas a partir del etiquetado de componentes conectados.

Clasificación. De acuerdo a la Fig. 2, esta etapa consta del siguiente paso.

Extracción de características: Las características de una clase de objeto son definidas a partir del cálculo de los momentos centrales normalizados de cada región de interés

obtenida en la etapa previa. Los momentos centrales normalizados, como es bien sabido, pueden ser calculados de la siguiente forma:

$$n_{pq} = \frac{\mu_{pq}}{\mu_{00}^{\gamma}}, \gamma = \frac{(p+q+2)}{2}, p+q = 2,3,\ldots \quad (1)$$

En este caso μ_{pq} representan los momentos del centroide, μ_{00} representa al momento central de orden cero y p, q son valores enteros no negativos. A partir de estos momentos centrales normalizados se realiza el cálculo del primer momento invariante de Hu el cual es utilizado para definir la clase a la que pertenece cada componente. El valor del primer momento invariante de Hu es obtenido de la siguiente manera:

$$\emptyset_1 = n_{20} + n_{02}. \quad (2)$$

En este caso n_{20} y n_{02} son momentos centrales normalizados de orden dos. El uso de los momentos invariantes de Hu en la etapa de clasificación es especialmente útil gracias a sus propiedades de invariancia ante traslaciones, rotaciones y cambios de escala [6]. Esto es comprobable a través de una matriz como la mostrada en la Tabla 1, donde se comparan los valores del primer momento invariante de Hu para 10 imágenes de cada objeto. Como puede observarse, los datos mostrados en la tabla reflejan poca dispersión entre los valores de cada clase, lo cual es un indicador, como es bien sabido, de que el proceso de clasificación se podrá realizar de manera robusta sin importar variaciones en los cambios de posición, las rotaciones y los cambios de escala de los objetos.

Reconocimiento: De acuerdo a la Fig. 2, esta etapa consta del siguiente paso:

Tabla 1. Los datos mostrados en la matriz de momentos reflejan poca dispersión para los 10 escenarios propuestos para cada clase de objeto.

$\emptyset_1$	Tornillo	Rondana	Armella	Alcayata	Cola de milano
Valor 1	2.8003	3.1431	2.6098	2.2586	3.0346
Valor 2	2.7835	3.1466	2.5967	2.2829	3.0320
Valor 3	2.7882	3.1517	2.5889	2.2923	3.0288
Valor 4	2.8043	3.1573	2.5833	2.2795	3.0281
Valor 5	2.7770	3.1464	2.6161	2.2829	3.0310
Valor 6	2.8149	3.1512	2.6073	2.2814	3.0412
Valor 7	2.7949	3.1525	2.6093	2.3052	3.0496
Valor 8	2.7802	3.1529	2.6029	2.2885	3.0434
Valor 9	2.7809	3.1493	2.6150	2.3056	3.0449
Valor 10	2.7872	3.1466	2.6079	2.3046	3.0374

Identificación del objeto: Con las características obtenidas se elige la clase de objeto a la que pertenece el componente mediante el uso de un clasificador de distancia euclidiana, el cual se encarga de calcular la distancia, en este caso, del valor del primer momento invariante de Hu del componente en relación con los valores representantes de las clases de objetos. Solo para recordar, esta distancia se define como:

$$distancia = \sqrt{(\emptyset_1 - representante)^2}. \quad (3)$$

Del conjunto de los valores de las distancias obtenidas se determina la clase de pertenencia para el componente mediante la selección de la distancia mínima, dada por la siguiente operación:

$$clase = argmin(distancia). \quad (4)$$

Finalizar. Si el sistema, por medio de la cámara de la plataforma de la Raspberry Pi, encuentra al menos un objeto que pertenezca a algunas de las clases definidas, la biblioteca de texto a voz `pyttsx` es instanciada pronunciando a través de un altavoz conectado a la Raspberry Pi el nombre de la etiqueta de la clase del objeto. Se usa un rango de velocidad promedio de 130 palabras por minuto por cada cuadro procesado en el que el objeto este presente.

2.2. Instalación y configuración de la plataforma Raspberry Pi

Existen varios modelos de plataformas tipo Raspberry Pi en el mercado, la diferencia entre estos modelos radica en las diversas configuraciones de hardware que presentan, siendo algunos modelos más avanzados gracias a sus especificaciones técnicas superiores permitiendo obtener una mayor eficiencia en su capacidad de procesamiento y administración de memoria. Una comparación detallada de los componentes técnicos de los distintos modelos se puede consultar en [9].

El modelo de la plataforma elegido para el desarrollo e implementación del sistema fue la Raspberry Pi 3 modelo B, sin embargo el sistema puede ser utilizado en otros modelos de la plataforma así como en otras plataformas embebidas que sean compatibles con las dependencias del sistema.

En lo que se refiere a la cámara de la plataforma, se utilizó la versión 2 del módulo de la cámara de la plataforma de la Raspberry Pi, la cual tiene una resolución de imagen de ocho megapíxeles y se conecta mediante el conector de la interfaz en serie de la cámara (CSI, por sus siglas en inglés) de la plataforma de la Raspberry Pi.

La implementación del sistema embebido en la plataforma tipo Raspberry Pi requirió de la integración de los componentes de hardware, los cuales son el módulo de la cámara y la tarjeta microSD con el sistema operativo instalado. El uso de un altavoz es opcional ya que su función dentro del sistema es con fines demostrativos únicamente.

En cuanto a la instalación y configuración del software en la plataforma, el sistema requiere de un sistema operativo para funcionar y ejecutar el archivo `script`. Para realizar la implementación descrita en este artículo, el sistema operativo utilizado fue `Raspbian`. No obstante, debido a que el sistema de reconocimiento de objetos fue diseñado para su uso en sistemas embebidos, este es multiplataforma, lo que significa que puede ser ejecutado en diferentes sistemas operativos sin necesidad de cambiar el código fuente. En lo que se refiere a las dependencias, el sistema requirió de la instalación de la biblioteca de visión por computadora OpenCV, la instalación de la interfaz `picamera` para la manipulación del módulo de la cámara de la Raspberry Pi y la instalación de la biblioteca de texto a voz `pyttsx`.

2.3. Ejecución en tiempo real de la implementación

Los resultados obtenidos son mostrados, de manera gráfica, en una ventana dividida en tres secciones como aparece en la Fig. 5: la sección "Original" muestra el cuadro actual procesado por la cámara, la sección "Umbralado" muestra los resultados de la etapa de binarizado del sistema para dicho cuadro y la sección "Resultado" permite visualizar los resultados de las etapas de clasificación y reconocimiento del sistema para el cuadro en cuestión.

Fig. 5. Resultados de la ejecución del sistema durante el procesamiento de video en tiempo real capturado por la cámara de la plataforma tipo Raspberry Pi.

3. Resultados experimentales

En esta sección se presentan los resultados de las pruebas de desempeño del sistema propuesto en distintos escenarios. A través de la matriz de confusión (Tabla 2) se muestran los resultados para una configuración de la cámara paralela al plano trabajo de los objetos a 10 centímetros de distancia entre la cámara y el espacio de trabajo. En este caso, como se puede ver los resultados siempre fueron los deseados y esto es respaldado por las propiedades de invariancia de los momentos de Hu.

Tabla 2. Matriz de confusión con los resultados del sistema para el escenario con cámara en posición paralela a los objetos a 10 centímetros de distancia.

Matriz de confusión Escenario 1		Valor predicho				
		Tornillo	Rondana	Armella	Alcayata	Cola de milano
Valor real	Tornillo	10	0	0	0	0
	Rondana	0	10	0	0	0
	Armella	0	0	10	0	0
	Alcayata	0	0	0	10	0
	Cola de milano	0	0	0	0	10

Por otro lado, en la matriz de confusión de la Tabla 3 se muestran los resultados para una configuración en la posición de la cámara con un ángulo de inclinación de 30° con respecto al plano de trabajo y a 15 centímetros de distancia. Nótese como en este caso ya se tienen algunos errores en la clasificación de algunos objetos.

Tabla 3. La matriz de confusión con los resultados del sistema para el escenario con la posición de la cámara a un ángulo de inclinación de 30° a 15 centímetros de distancia.

Matriz de confusión Escenario 2		Valor predicho				
		Tornillo	Rondana	Armella	Alcayata	Cola de milano
Valor real	Tornillo	9	0	1	0	0
	Rondana	0	10	0	0	0
	Armella	0	0	8	0	2
	Alcayata	0	0	0	10	0
	Cola de milano	0	1	0	0	9

Tabla 4. La matriz de confusión con los resultados del sistema para el escenario con posición de la cámara a un ángulo de inclinación de 45° y a 10 centímetros de distancia.

Matriz de confusión Escenario 3		Valor predicho				
		Tornillo	Rondana	Armella	Alcayata	Cola de milano
Valor real	Tornillo	8	0	2	0	0
	Rondana	0	9	0	0	1
	Armella	0	0	7	3	0
	Alcayata	0	0	1	9	0
	Cola de milano	0	2	0	0	8

En un tercer experimento, la cámara se situó a 10 centímetros del plano de trabajo a 45° de inclinación entre la cámara y dicho plano. La matriz de confusión de la Tabla 4 muestra los resultados obtenidos. Nótese como en esta situación el sistema tiende a presentar más errores. Es de suponer que a mayores inclinaciones y más alturas el sistema ofrecerá mayores errores.

El desempeño del clasificador es calculado con base a la siguiente fórmula:

$$d = 1 - \frac{errores}{150} x100 = n\%. \quad (5)$$

Por lo tanto, considerando los 13 errores obtenidos, el desempeño d del sistema propuesto para los 150 escenarios adoptados es de un 91.3%. En la Tabla 5 se muestran los tiempos y la cantidad de memoria requeridos por la plataforma tipo Raspberry Pi para procesar los objetos contenidos en un cuadro capturado por la cámara.

Tabla 5. Tiempo de procesamiento y cantidad de memoria requeridos según el número de objetos.

Número de objetos procesados (por cuadro)	Tiempo de procesamiento (en milisegundos)	Cantidad de memoria (en mebibytes)
0 objetos	38.2 ms	17.73 MiB
1 objeto	54.0 ms	17.92 MiB
2 objetos	54.9 ms	18.11 MiB
3 objetos	55.4 ms	18.30 MiB
5 objetos	58.1 ms	18.68 MiB
10 objetos	66.5 ms	19.63 MiB
15 objetos	75.4 ms	20.58 MiB

De esta tabla se puede observar que el número de objetos en la imagen efectivamente influye en los tiempos invertidos para el procesamiento completo de la imagen, de la misma manera ocurre en la cantidad de memoria utilizada en donde se puede observar una variación incremental de 0.19 MiB por cada objeto presente en la imagen.

4. Conclusiones y trabajo a futuro

En este artículo se describió un sistema eficiente para el reconocimiento de objetos cuasi-planos compatible con sistemas embebidos. Esta propuesta es aplicable a cualquier plataforma embebida cuyo sistema operativo sea compatible con las dependencias utilizadas por el sistema; solo los resultados en la plataforma de la Raspberry Pi 3 modelo B fueron presentados en este artículo.

Para medir el desempeño del sistema se propusieron cinco clases de objetos. No obstante el método permite definir hasta *n* objetos siempre y cuando los representantes de las clases sean lo suficientemente diferentes entre sí. Si la diferenciación entre los objetos adquiere un carácter no lineal, se deberá adoptar otro tipo de clasificador, por ejemplo, un arreglo de perceptrones en capas.

Una de las ventajas principales del sistema propuesto es su bajo consumo de procesamiento y memoria, lo que permite que varios cuadros capturados por la cámara conectada a la plataforma de la Raspberry Pi sean procesados simultáneamente, consiguiendo clasificar los objetos de manera correcta sin afectar el rendimiento del sistema.

Otra ventaja importante es la capacidad de reconocimiento en tiempo real de múltiples clases de objetos ubicados de forma dispersa de manera simultánea por cada cuadro procesado por la cámara, lo cual significa que el sistema puede ser implementando en líneas de producción industrial para el reconocimiento de múltiples objetos optimizando la toma de decisiones en los sistemas de producción en cadena.

De los experimentos realizados con cinco tipos de clases de objetos en una plataforma tipo Raspberry Pi se puede concluir que el desempeño del sistema es lo suficientemente bueno como para ser utilizado en dispositivos embebidos de características limitadas sin sacrificar la robustez del proceso de reconocimiento, obteniendo resultados similares a los que se obtendrían en un equipo de cómputo convencional.

Los trabajos a futuro incluyen: 1) medir el desempeño del sistema propuesto en problemas con más de cinco clases de objetos, 2) cambiar de clasificador, por ejemplo, una red neuronal artificial o una máquina vector soporte para poder afrontar problemas de carácter no lineal, e 3) implementar el sistema en robots móviles que puedan realizar determinadas acciones a partir de las clases de objetos reconocidos.

Reconocimientos. Julián Jerónimo agradece a la Academia Mexicana de Ciencias y a su programa "Verano de la Investigación Científica" por otorgarle la beca para realizar su estancia de investigación en el Centro de Investigación en Computación del IPN. H. Sossa agradece al Instituto Politécnico Nacional y al CONACYT por los apoyos

económicos, en el marco de los proyectos SIP 20170693, SIP 20180730 y 50 (Fronteras de la Ciencia).

Referencias

1. Arévalo, E., Zúñiga, A., Villegas, J., Avilés, C. Implementación de reconocimiento de objetos por color y forma en un robot móvil. Research in Computing Science, 91, pp. 21–31 (2015)
2. Ali, M., Dong, L., Liang, Y., Xu, Z., He, L. Feng, N.: A color image retrieval system based on weighted average. International Conference on Signal Processing, Communication and Computing, pp. 184–189 (2014)
3. Rosebrock, A.: Practical Python and OpenCV. 1ra ed. PyImageSearch, USA (2014)
4. Firdousi, R., Parveen S.: Local Thresholding Techniques in Image Binarization. International Journal of Engineering and Computer Science Vol: 3, pp. 4062–4065 (2014)
5. Burger, W., Burge, M.: Principles of Digital Image Processing, Undergraduate Topics in Computer Science. Springer-Verlag, London (2009)
6. Flusser, J.: Moment Invariants in Image Analysis, International Journal of Computer. Electrical, Automation, Control and Information Engineering Vol: 1, No: 11, pp. 196–201 (2006)
7. Li, D.: Analysis of Moment Invariants on Image Scaling and Rotation. In: Sobh T., Elleithy K. (eds.) Innovations in Computing Sciences and Software Engineering, pp. 415–419, Springer, Dordrecht (2010)
8. Jing, L., Bao-Liang, L.: An adaptive image Euclidean distance. Pattern Recognition 42, pp. 349–357 (2009)
9. Pajankar, A., Kakkar, A.: Raspberry Pi by Example. 1ra ed. Packt Publishing, UK (2016)

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Abril de 2018
Printing 500 / Edición 500 ejemplares