Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
Josef Kittler (Eds.)

LNCS 6256

Advances in Pattern Recognition

Second Mexican Conference on
Pattern Recognition, MCPR 2010
Puebla, Mexico, September 2010, Proceedings

Springer

Lecture Notes in Computer Science 6256

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

Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
Josef Kittler (Eds.)

Advances in Pattern Recognition

Second Mexican Conference on Pattern Recognition, MCPR 2010
Puebla, Mexico, September 27-29, 2010
Proceedings

Springer

Volume Editors

Jesús Ariel Carrasco-Ochoa
José Francisco Martínez-Trinidad
National Institute of Astrophysics, Optics and Electronics (INAOE)
Computer Science Department, Luis Enrique Erro No. 1
72840 Sta. Maria Tonantzintla, Puebla, Mexico
E-mail: {ariel; fmartine}@inaoep.mx

Josef Kittler
University of Surrey
Centre for Vision, Speech and Signal Processing
School of Electronics and Physical Sciences
Guildford GU2 7XH, United Kingdom
E-mail: j.kittler@surrey.ac.uk

Library of Congress Control Number: Applied for

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

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

ISSN 0302-9743
ISBN-10 3-642-15991-5 Springer Berlin Heidelberg New York
ISBN-13 978-3-642-15991-6 Springer Berlin Heidelberg New York

springer.com

Printed in Germany

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

Preface

The Mexican Conference on Pattern Recognition 2010 (MCPR 2010) was the second event organized by the Mexican Association for Computer Vision, Neurocomputing and Robotics (MACVNR). These conferences provide a forum for exchanging scientific results and experiences, sharing new knowledge, and increasing the cooperation between research groups in pattern recognition and related areas, in Mexico, as well as international cooperation.

MCPR 2010 was held in Puebla, Mexico, it was hosted and sponsored by the Computer Science Department of the National Institute of Astrophysics, Optics and Electronics (INAOE), jointly with the Center for Computing Research of the National Polytechnic Institute (CIC-IPN).

From 89 full papers submitted, 39(43.8%) were accepted for publication in these proceedings and for presentation at the conference. The contributions originated from 20 different countries. The review process was carried out by the Scientific Committee, composed of internationally recognized scientists, all experts in their respective fields.

The conference benefited from the contributions made by the invited speakers: Edwin Hancock from the Department of Computer Science, University of York (UK); Guozhu Dong from the Data Mining Research Lab, Department of Computer Science and Engineering, Wright State University (USA); and Ernesto Bribiesca Correa from the Department of Computer Science of the Center for Research in Applied Mathematics and Systems of the National Autonomous University of Mexico, IIMAS-UNAM (Mexico). We would like to express our sincere gratitude to the invited speakers.

We would also like to thank the members of the Organizing Committee for their enormous effort that resulted in these excellent conference proceedings. We trust that the papers in this volume will provide not only a record of the recent advances in this rapidly moving field, but will also stimulate future applied and theoretical research in the different areas of pattern recognition.

We hope this edition of the Mexican Conference on Pattern Recognition becomes a cornerstone of a series of many future events engaging the Mexican pattern recognition researchers and practitioners in discourse with the broader international pattern recognition community.

September 2010

José Francisco Martínez-Trinidad
Jesús Ariel Carrasco-Ochoa
Josef Kittler

Organization

MCPR 2010 was hosted and sponsored by the Computer Science Department of the National Institute of Astrophysics, Optics and Electronics (INAOE).

General Conference Co-chairs

Josef Kittler	Department of Electrical Engineering of Surrey University, UK
José Francisco Martínez-Trinidad	Computer Science Department, National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico
Jesús Ariel Carrasco-Ochoa	Computer Science Department, National Institute of Astrophysics, Optics and Electronics (INAOE), Mexico

Local Committee

María del Pilar Gómez Gil
Jesús Antonio González Bernal
Eduardo Morales Manzanares
Carlos Alberto Reyes García

Local Arrangements Committee

Carmen Meza Tlalpan
Gorgonio Cerón Benítez
Gabriela López Lucio

Scientific Committee

Alquézar Mancho, R.	Universitat Politécnica de Catalunya, Spain
Asano, A.	Hiroshima University, Japan
Bagdanov, A.	Universitat Autònoma de Barcelona, Spain
Bayro-Corrochano, E.	CINVESTAV-Guadalajara, Mexico
Benedi, J.M.	Universidad Politécnica de Valencia, Spain
Bigun, J.	Halmstad University, Sweden
Bonastre, J.F.	Université dÁvignon, France
Borges, D.L.	Universidade de Brasilia, Brazil
Bunke, H.	University of Bern, Switzerland
Caldas Pinto, J.R.	Instituto Superior Técnico, Portugal
Chetverikov, D.	Computer and Automation Research Institute, Hungary

Chollet, G.	ENST, France
Coello-Coello, C.A.	CINVESTAV, Mexico
Facon, J.	Pontifícia Universidade Católica do Paraná, Brazil
Ferri, F.J.	Universitat de València, Spain
Gelbukh, A.	CIC-IPN, Mexico
Graña, M.	University of the Basque Country, Spain
Grau, A.	Universitat Politécnica de Catalunya, Spain
Guzmán-Arenas, A.	CIC-IPN, Mexico
Haindl, M.	Institute of Information Theory and Automation, Czech Republic
Hanbury, A.	Vienna University of Technology, Austria
Hancock, E.R.	University of York, UK
Hernando, J.	Universitat Politécnica de Catalunya, Spain
Heutte, L.	Université de Rouen, France
Hlavac, V.	Czech Technical University, Czech Republic
Jiang, X.	University of Münster, Germany
Kampel, M.	Vienna University of Technology, Austria
Klette, R.	University of Auckland, New Zealand
Kober, V.	CICESE, Mexico
Koster, W.	Universiteit Leiden, The Netherlands
Lopez de Ipiña, K.	University of the Basque Country, Spain
Lorenzo-Ginori, J.V.	Universidad Central de Las Villas, Cuba
Martins-Gomes, H.	Universidade Federal de Campina Grande, Brazil
Mayol-Cuevas, W.	University of Bristol, UK
Mejail, M.	Universidad de Buenos Aires, Argentina
Morales, E.	INAOE, Mexico
Niemann, H.	University of Erlangen-Nuremberg, Germany
Pardo, A.	Universidad Católica del Uruguay, Uruguay
Pérez de la Blanca-Capilla, N.	Universidad de Granada, Spain
Petrou, M.	Imperial College, UK
Pina, P.	Instituto Superior Técnico, Portugal
Pistori, H.	Dom Bosco Catholic University, Brazil
Radeva, P.	Universitat Autònoma de Barcelona, Spain
Real, P.	University of Seville, Spain
Rodríguez, R.	ICIMAF, Cuba
Ross, A.	West Virginia University, USA
Rueda, L.	University of Windsor, Canada
Ruiz-Shulcloper, J.	CENATAV, Cuba
Sanches, J.	Universidade Tecnica de Lisboa, Portugal
Sánchez, J.S.	Universitat Jaume I, Spain
Sang-Woon, K.	Myongji University, Republic of Korea
San Martin, C.	University of La Frontera, Chile

Sanniti di Baja, G.	Istituto di Cibernetica, CNR, Italy
Sansone, C.	Università di Napoli, Italy
Santana, R.	University of the Basque Country, Spain
Shengrui, W.	University of Sherbrooke, Canada
Shirai, Y.	Ritsumeikan University, Japan
Sossa Azuela, J.H.	CIC-IPN, Mexico
Sousa-Santos, B.	Universidade de Aveiro, Portugal
Sucar, L.E.	INAOE, Mexico
Taboada Crispi, A.	Universidad Central de Las Villas, Cuba
Torres, M.I.	University of the Basque Country, Spain
Valev, V.	Prince Mohammad Bin Fahd University, Saudi Arabia

Additional Reviewers

Altamirano, L.
Batyrshin, I.
Carvalho, J.
Conti Pereira, M.
Gómez Gil, M.P.
González Bernal, J.A.
Hernandez-Sierra, G.
Huenupan, F.
Klette, G.
Ledeneva, Y.
Montes, M.
Mora, M.
Moreira, J.
Nolazco, J.A.
Olvera López, J.A.
Reyes García, C.A.
Romero Romero, M.E.
Sánchez Díaz, G.
Sánchez Soto E.
Silva, S.
Tomé, A.M.
Vega-Pons, S.
Wu, S.
Xiong, T.

Sponsoring Institutions

National Institute of Astrophysics, Optics and Electronics (INAOE)
Center for Computing Research of the National Polytechnic Institute (CIC-IPN)
Mexican Association for Computer Vision, Neurocomputing and Robotics (MACVNR)
International Association for Pattern Recognition (IAPR)

Table of Contents

Computer Vision and Robotics

Image Processing

Neural Networks and Signal Processing

Pattern Recognition and Data Mining

Natural Language and Document Processing

A Hierarchical Recursive Partial Active Basis Model

Pavel Herrera-Domínguez[1] and Leopoldo Altamirano-Robles[2]

[1] INAOE, Puebla
pavel13@inaoep.mx
[2] INAOE, Puebla
robles@inaoep.mx

Abstract. Recognition of occluded objects in computer vision is a very hard problem. In this work we propose an algorithm to construct a structure of a model using learned active basis models, then use it to do inference over the most probable detected parts of an object, to allow partial recognition using the standard sum-max-maps algorithm used for *active basis*. We tested our method and present some improvements on occluded face detection using our algorithm, we also present some experiments with other partially occluded objects.

1 Introduction

The problem of occlusion is a very hard problem in computer vision, because we need to fit some model to some image but we do not know all the possible ways the model can appear occluded, therefore we propose an algorithm to split a global model and then fit it by parts without losing the spatial information.

The objective of this work is to propose an algorithm to construct a model that gives the chance to detect occluded parts in a natural way, and to construct it with the less possible user supervision , in the way to avoid the time consuming labeling. In this work we assume that the object is centered and bounded, this can be resolved using local templates for active basis [8] or applying similar algorithms to the proposed by Zhu,L et al. [9].

2 Previous and Related Work

There are several works about the occlusion problem, most of them using discriminative models like Viola-Jones *adaboost* [4], some other approaches use grammars in the inference stage, like Wu et al. [6] that give a numerical study about the importance of each stage on the top-down and button-up inference. There are also some works for generative models, like the one from Baker et al. [1] that gives an extension for the AAM models(Cootes et al.[2]) using a more robust metric on the inference stage.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 1–10, 2010.

There are also works related to construct a hierarchical model like the proposed by Zhu,l et al [9] that gives an unsupervised algorithm to construct models using structures formed by edge-lets. Also a recent work for pedestrian detection that use several part-detectors and merge them to have better detection[5].

The difference of our work with others is that for example we do not try to detect parts independently like Wu et al.[6] and merge them. The difference with the original Sum-Max-Maps[8] used for active basis (the representation we are using) is that we have an intermediate stage merging the parts and deciding if they are present or not, this way we can have a *partial* recognition of the object that could be useful in some applications. In the case of the learning algorithm applied, the difference and contribution is that we use the algorithm proposed by Wu et al. [8], learning algorithm also spatial relations that help in the recognition step.

3 Active Basis

The active basis model is a deformable model which consist of a small number of Gabor wavelet elements $B_{x,y,s,\theta}$ at selected locations and orientations. Each element is given by $B_{x,y,s,\theta}(x',y') = G(\hat{x}/s, \hat{y}/s)$ where $G(x,y) = e^{-\frac{(\frac{x}{\sigma_x})^2+(\frac{y}{\sigma_y})^2}{2}} e^{ix}$ and $\hat{x} = (x'-x)cos\alpha - (y'-y)sin\alpha$, $\hat{y} = (x'-x)sin\alpha + (y'-y)cos\alpha$, s is the scale parameter and α is the orientation[7]. Using this elements we can represent the image as follows.

$$I_m = \sum_{i=0}^{n} c_{m,i} B_{m,i} + \epsilon \tag{1}$$

where ϵ is the residual image and

$$\begin{gathered}
B_{m,i} \approx B_i \\
B_i = B_{x_i,y_i,s,\alpha_i} \\
B_{m,i} = B_{x_{m,i},y_{m,i},s,\alpha_{m,i}} \\
x_{m,i} = x_i + d_{m,i} \sin \alpha_i \\
y_{m,i} = y_i + d_{m,i} \cos \alpha_i \\
\alpha_{m,i} = \alpha_i + \delta_{m,i} \\
d_{m,i} \in [-b_1, b_1], \delta_{m,i} \in [-b_2, b_2]
\end{gathered} \tag{2}$$

This means that B_i is allowed to shift and rotate in the intervals given, allowing small deformation in the object model.

The full details are explained in the original papers by Zhu et al. [7][8], here we give only a short description of how they are learned and how they are used in recognition and detection.

3.1 Learning Active Basis

The active basis model specifies the distribution of the image I as in equation (3).

$$p(I|B) = q(I)\frac{p(C)}{q(C)} = q(I)\frac{p(c_1, ..., c_n)}{qc_1, ..., c_n} \tag{3}$$

where $q(I)$ is the reference distribution. Assuming independence between the Gabor elements we have.

$$p(I|B) = q(I)\prod_{i=1}^{n}\frac{p(c_1)}{q(c_i)} \tag{4}$$

where $p(c_i)$ is parametrized as an exponential family model $p(c_i; \lambda) = \frac{1}{Z(\lambda)}e^{\lambda h(r_i)}q(c_i)$ where $r_i = | < I, B_i > |^2$ is the local energy of Gabor filter response and $h(r_i)$ is a sigmoid transformation function, and $q(c_i)$ is pooled from generic background images in an off-line stage. Replacing the parametrized distribution on equation(4) we have the probability of a single image given the model in equation (5).

$$p(I|B) = q(I)\prod_{i=1}^{n}\frac{1}{Z(\lambda)}e^{\lambda h(r_i)} \tag{5}$$

Now to learn B from a set of training images $I = \{I_1, ..., I_M\}$, we need to maximize the log-likelihood $\log\prod_{m=1}^{M}\frac{p(I_m|B_m)}{q(I_m)}$ over all images

$$\sum_{m=1}^{M} log(\frac{p(I_m|B_m)}{q(I_m)}) = \sum_{m=1}^{M} log(\frac{p(r_m)}{q(r_m)}), \tag{6}$$

when $M \rightarrow \infty$ we are maximizing the Kullback-Leibler divergence estimator between p and q. This way we learn B, this is for each B_i we have five parameters for each element $(x_i, y_i, \alpha_i, \lambda_i, \log Z_i)$, where (x_i, y_i, α_i) the parameters of the Gabor elements, and $(\lambda_i, \log Z_i)$ the parameters of the images responses distribution.

3.2 Using the Model

This way of learning B_i also give us a score function to find the model in a new image and sketch the object in the image. To find the model given by B we maximize the equation (8) this means to find the correct parameters Θ (center scale and position of the model, plus the parameters for each element of $\{B_i\}$)

$$argmax_{\Theta}\frac{P(I_m|B,\Theta)}{q(I_m)} = argmax_{\Theta}log(\frac{P(I_m|B,\Theta)}{q(I_m)}) \tag{7}$$

and

$$log(\frac{P(I_m|B,\Theta)}{q(I_m)}) = \sum_{i=0}^{n}(\lambda_i * h(r_{i,m})) - log(Z_i) \tag{8}$$

The full details of how to do this can be found in the original paper [8].

4 Formulation of the Occlusion Problem

Although in the literature exist very effective algorithms to learn models using *active basis* for deformable objects and use these models in recognition tasks, still there are not many advances in the solution of how to solve the occlusion problem, there are a solutions for example using grammars[6], using alpha, beta, gamma processes, but in this case the *active basis* models are used as simple detectors in the leaves of the grammar.

In this work we propose a different approach, let us first comment some key points to be considered.

1. There is no way to learn all possible forms that an object can be occluded.
2. The problem of occlusion seen from the active basis point of view can be interpreted as follows: there are some Gabor elements (B_i) not present in the image that contains the object to be recognized, so instead of having all B_i we will have only a subset of them.

The previous points give us the idea to consider the problem of occlusion as finding the most probable subset of basis that are present in the image.

5 Recursive Hierarchical Active Basis

In this work we propose an extension of what is proposed by Ying Nian et al.[8] who proposed to use part-templates to form a recursive model to deal with articulated objects, they give an inference algorithm named recursive sum-max maps. What we propose is to construct a hierarchical-graph with the active basis, having two kind of relations, inter-level that is relations of the active basis at the same level of detail, and relations intra-level, this is how more detailed active basis than the original one are related, the figure (1) describes the idea. R_i are the relations inter-level, and they are given by the indexes of the low detailed basis corresponding to the higher detail basis. The inter-level relations are spatial relations given as relative positions of the surrounding squares of each part.

5.1 Learning

We have already seen that we can learn a model B given a set of images of one object. Now we will describe how to learn the recursive structure. In algorithm (1) we can see the pseudo-code of how to learn the structure. The basic idea is to learn separate detailed parts of the images guiding the learning algorithm with the basis already learned in a low-detailed more general scale. This way we can construct a graph that contains the relations between the parts maintaining which elements correspond to each new detailed part, and at the same time we have spatial relations between the same level of detail basis. To learn a priori probabilities, that will be used in the inference algorithm, we use the percentage of the object represented by that sub-part learned.

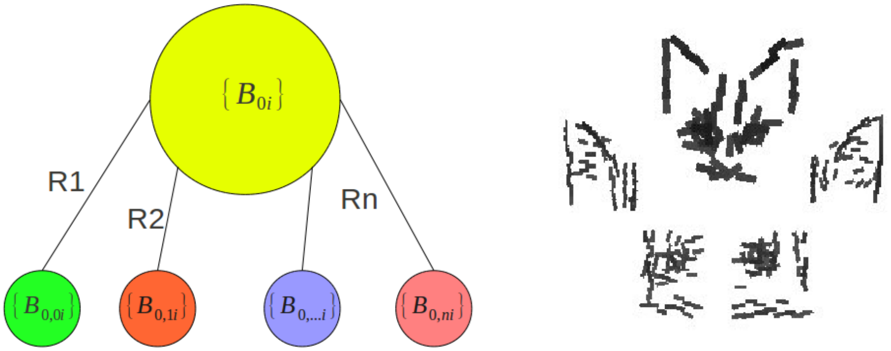


Fig. 1. Illustrative idea of the split model and its parts with the spatial relations to the global model

Algorithm 1. Learn-recursive-model

```
Require: $Itrain$
  $\{B_i\}$ ←learn from Itrain at scale Scale
  if $level = max\_level$ then
    $output \leftarrow \{B, \emptyset\}$
  else
    $Basis\_parts \leftarrow Split\{B_i\}$ using spatial relations
    for $part \in Basis\_Parts$ do
      for $img \in Itrain$ do
        $nTrain \leftarrow \{nItrain\} \cup subimage(img, part.subWindow)$
      end for
      $\{part.B_i\} \leftarrow$ Learn-recursive-model using nItrain
      $\{part.P(this|Parts)\} \leftarrow$ Probability of this part given that is a subpart
    end for
    $output \leftarrow \{B, Basis\_parts\}$
  end if
```

6 Using the Hierarchy

We modify the algorithm proposed by Wu et al. [8] to improve the recognition even when the object is partially occluded. The idea is based on the following simple observations.

1. If we need to maximize the log-likelihood, it is clear from the equation (8) that we should take the values which are greater than some threshold, that should represent the basis that are present as part of the object.
2. If we take the previous observation as a method to see the presence or not of and object, we need to merge them, because it is not enough taking them alone.

Algorithm 2. Computing SUM2MAP for incomplete basis

```
for $x = 1$ to W do
  for $y = 1$ to H do
    for $t = 1$ to NumberOfBasis do
      if $\lambda_t * MAX1MAP(x + B_{t,x}, y + B_{t,y}, B_{t,\theta}) - \log Z_t > 0$ then
        $H = H \cup \{t\}$
      end if
    end for
    $SUM2MAP(x, y) \leftarrow solve(H)$ this is Algorithm (3)
    $PresentBasis(x, y) \leftarrow H$
  end for
end for
```

Algorithm 3. Solve the inference problem button up using Dynamic Programming given the Hypothesis

```
Initialize $Table$ to $\epsilon$
$level \leftarrow lowest\_level$
for $t \in H$ do
  $Table[level][t] \leftarrow 1$
end for
$level \leftarrow level - 1$
while $level \geq 0$ do
  for $part \in Parts[level]$ do
    $prob \leftarrow 0$
    for $subpart \in part$ do
      $prob \leftarrow prob + Table[level + 1][subpart] * P(subpart|part)$
    end for
    $Table[level][part] \leftarrow prob$
    if $Table[level][part] < \epsilon_{part}$ then
      Retrieve info to H, this is when the part is not present just for sketching
    end if
  end for
  $level \leftarrow level - 1$
end while
$output \leftarrow Table[0][0]$
```

With this observations the SUM-MAX-MAPS algorithm [8] can be used to partially find an object, adding the relations learned in the training gives us an easy and natural way to merge the sub-parts.

In algorithm (2) it is shown how the algorithm sum-max-maps [8] is modified in the stage SUM2 to apply the idea proposed in this work. Here what we do is to take just the elements that are over some threshold and take them as the most simple hypothesis known, then merge them and compose the object to detect it.

6.1 Merging the Elements

Merging the found elements can be done by computing the probability as in equation (9).

$$P(Obj|\theta) = \sum_{SubPart \in Obj} P(SubPart|Obj)P(SubParts|\theta_{subPart}) \tag{9}$$

This can be accomplished by using dynamic programming, like is shown in algorithm (3), where each subproblem is to compute $P(Obj|SubParts)$, this means given the subparts estimate the *percentage* of the object *present*, then use Obj and $P(Obj|SubParts)$ as sub-part of a more general object, the leaves of the dynamic programming tree are the elements of Active Basis that satisfy the threshold mentioned in the MAX1 maps [8].

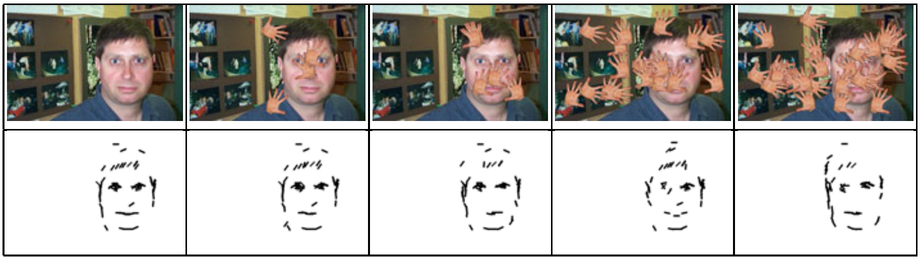
7 Experiments

We carried out some experiments with the inference algorithm and the structure learned in face detection task, to see its behavior with occlusion. The images were took from the Caltech 101 data base subcategory Faces[3]. We constructed several artificial images from this database generating some random occlusions in the image to know the level of occlusion it supports. The images contain random objects over all the image. We notice that if we only generate occlusion over the object of interest, the edges generated by these objects tend to form the face edges and help the matching score. To generate the occlusions we place the objects and count the number of pixels in the image that were occluded by the random objects until they satisfy a threshold that depends on the level assigned. Table (1) has examples of images under different levels of occlusion. To train the model we took the faces centered and bounded without any occlusion, the model had 100 elements for the model used with all the experiments.

The localization accuracy is measured by predicting object bounding boxes. For detection to be correct the area $A(rect)$ of overlap between a predicted bounding box B^c and a ground truth bounding box B^{gt} must be more than half the union of both areas:

$$B^c \text{ } is \text{ } correct \Leftrightarrow \frac{A(B^c \cup B^{gt})}{A(B^c \cup B^{gt})} \geq \frac{1}{2} \tag{10}$$

Table 1. Examples of the increasing level of occlusion and the sketches in one image of the data set. The size of all images is around 500 x 350 pixels.



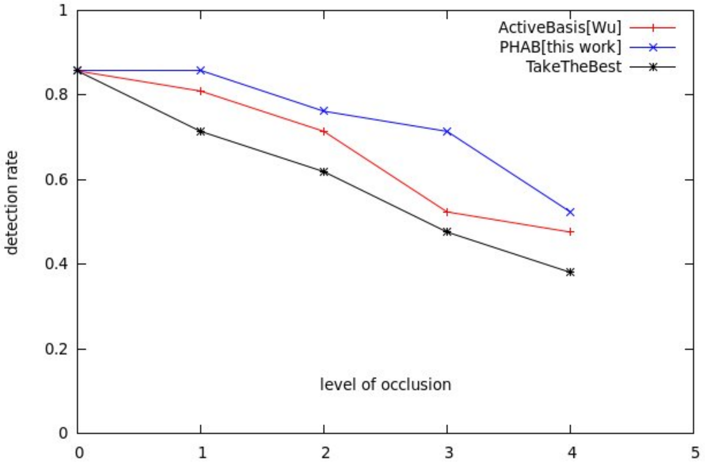


Fig. 2. Plotting level of occlusion in axis X vs detection rate in axis Y. The red plot (the curve in the middle) denotes the recognition rates using Sum-max-maps using the equation (8), the one in black (the curve in the bottom) plot was produced using the equation (8) but with the elements used in the blue one (the curve on the top). And finally the blue one is the proposed algorithm. Note how the proposed algorithm overcomes the original one.

So the localization rate is:

$$rate = \frac{positives}{positives + negatives} \tag{11}$$

where the positives are the hypothesis or bounding boxes that holds with equation (10).

The resulted model and the split sub-parts are shown in figure (3). The figure (2) shows the rates under several levels of occlusion; in the plot level 0 is the image without any occlusion. The second row of table (1) shows the partial sketches.

We made another experiment to see the performance on objects with *natural* occlusion. Using a set of cat faces we train the model showed in figure (1) and applied to a cat image taken from the Internet, the result is shown in figure (4). Also we trained a model for cars and tested on a image with some persons in the front, the results are presented in figure (4) also.

7.1 Implementation Details

We took the original code for active basis implemented in C++[1], then we modify and added the algorithm we propose. The parameters of *deformation* mentioned in equation (2) were the same for all the experiments, for training and testing:

[1] http://www.stat.ucla.edu/~ywu/AB/active_basis_cpp.html

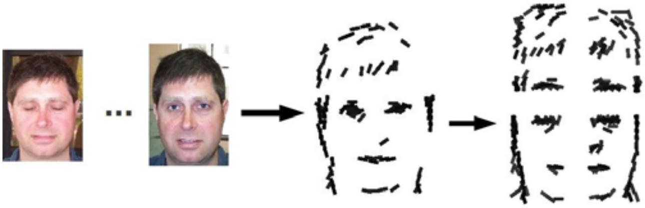


Fig. 3. Learned model and its parts, from a trainig set

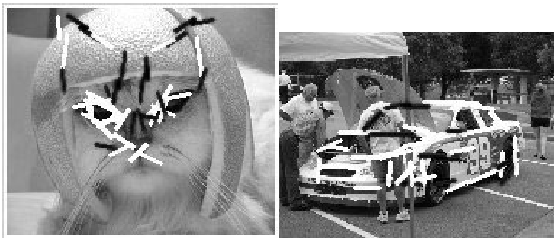


Fig. 4. Two diffetent examples to test the algorithm, the left one is a occluded cat an in the right a car with partial occlusion, the white lines denote the basis founded and the *black* ones the predicted ones

number of different orientations 15, angle deformation $\alpha_i = 3$ and the spatial deformation $d_i = 3$.

8 Conclusions

We have shown that is possible to detect and sketch the object even when there is partial occlusion. We have improved the detection rates compared with sum-max-maps[8], the detection can be increased even more if we use a better way to construct the model. We showed that by using the model separated on parts the detection rates can be increase under partial occlusion conditions.

Right now we are working splitting the model in the recognition stage, this modification is expected to work better on occlusion scenarios. It is worth to comment that algorithm behaves poor when the scale of the object changes considerably. As future work we will use these models to detect other kind of objects where natural occlusion occurs, like pedestrians, or cars on a parking lot.

Acknowledgment

The work reported in this paper has been supported by CONACyT scholarship 271666.

References

1. Baker, S., Matthews, I., Xiao, J., Gross, R., Kanade, T., Ishikawa, T.: Real-time non-rigid driver head tracking for driver mental state estimation. In: 11th World Congress on Intelligent Transportation Systems, Citeseer (2004)
2. Cootes, T.F., Edwards, G.J., Taylor, C.J.: Active appearance models. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(6), 681–685 (2001)
3. Fei-Fei, L., Fergus, R., Perona, P.: One-shot learning of object categories. IEEE Transactions on Pattern Analysis and Machine Intelligence 28(4), 594–611 (2006)
4. Viola, P., Jones, M.: Rapid Object Detection using a Boosted Cascade of Simple. In: Proc. IEEE CVPR 2001 (2001)
5. Wu, B., Nevatia, R.: Detection and segmentation of multiple, partially occluded objects by grouping, merging, assigning part detection responses. International Journal of Computer Vision 82(2), 185–204 (2009)
6. Wu, T., Zhu, S.C.: A Numerical Study of the Bottom-up and Top-down Inference Processes in And-Or Graphs. In Review (2009)
7. Wu, Y.N., Si, Z., Fleming, C., Zhu, S.C., Ucla, L.A.: Deformable Template As Active Basis. In: IEEE 11th International Conference on Computer Vision, ICCV 2007, pp. 1–8 (2007)
8. Wu, Y.N., Si, Z., Gong, H., Zhu, S.C.: Learning active basis model for object detection and recognition. International Journal of Computer Vision, 1–38 (2009)
9. Zhu, L., Lin, C., Huang, H., Chen, Y., Yuille, A.: Unsupervised structure learning: hierarchical recursive composition, suspicious coincidence and competitive exclusion. In: Forsyth, D., Torr, P., Zisserman, A. (eds.) ECCV 2008, Part II. LNCS, vol. 5303, pp. 759–773. Springer, Heidelberg (2008)

Facial Feature Model for Emotion Recognition Using Fuzzy Reasoning

Renan Contreras, Oleg Starostenko, Vicente Alarcon-Aquino,
and Leticia Flores-Pulido

CENTIA, Department of Computing, Electronics and Mechatronics
Universidad de las Américas Puebla, Cholula, 72820, México
{renan.contrerasgz,oleg.starostenko,
vicente.alarcon,leticia.florespo}@udlap.mx

Abstract. In this paper we present a fuzzy reasoning system that can measure and recognize the intensity of basic or non-prototypical facial expressions. The system inputs are the encoded facial deformations described either in terms of Ekman´s Action Units (AUs) or Facial Animation Parameters (FAPs) of MPEG-4 standard. The proposed fuzzy system uses a knowledge base implemented on knowledge acquisition and ontology editor Protégé. It allows the modeling of facial features obtained from geometric parameters coded by AUs - FAPs and also the definition of rules required for classification of measured expressions. This paper also presents the designed framework for fuzzyfication of input variables for fuzzy classifier based on statistical analysis of emotions expressed in video records of standard Cohn-Kanade's and Pantic´s MMI face databases. The proposed system has been tested in order to evaluate its capability for detection, classifying, and interpretation of facial expressions.

Keywords: Emotion recognition, facial features, knowledge-based framework, rules-based fuzzy classifier.

1 Introduction

The fuzzy systems and their combination with neural networks have been successfully used for pattern recognition and for image indexing and interpretation [1, 2]. In the area of facial expression recognition the application of a fuzzy reasoning remains marginal despite that some researchers have successfully used classifying systems which emulate the way as humans identify prototypical expression [3, 4, 5]. Usually the facial expression recognizing systems are based on two parts: a module for generation of feature vector corresponding to emotion expression in analyzed image (described by pixel position, colors, shapes, regions, etc.) and a classification module that recognize an emotion and describe its intensity.

There are a lot of techniques that have been used for facial features extraction. Some of them are based on Gabor Wavelets [5], Active Appearance and Geometric Models [6], Principal Components Analysis and Hierarchical Radial Basis Function Network [7], Optical Flow and Deformable Templates [8], Discrete Cosine Transform and Neural Networks [9], Multilevel Hidden Markov Models [10], Dynamic Bayesian

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 11–21, 2010.

networks [11], and others. Even though these approaches may extract and interpret the facial actions, there are not reports about how they may link standard facial actions with particular formal models or rules for automatic facial expression recognition.

In this paper we present the development of a fuzzy reasoning system, which is able not only to recognize facial expressions using standard Ekman´s AUs (Action Units), FAPs (Facial Animation Parameter) and FDPs (Facial Definition Parameter) of MPEG-4 standard, but also to measure their intensity [12]. This proposal permits to create novel models for precise facial feature detection as well as recognition and interpretation of basic and non-prototypical emotions. The proposed fuzzy system for facial expression recognition consists of two principal modules. The first one is a knowledge-based framework for modeling facial deformations by FAP and AU action units [13] developed according to well-known standards [14, 15]. The second module is used for recognizing facial expressions by fuzzy classifier in combination with Gaussian functions providing measurement of emotion intensity. To reduce relative subjectivity and lack of psychological meaning of emotional intensity levels the statistical analysis of facial actions in Cohn-Kanade´s and Pantic´s image databases has been implemented [16, 17]. This information has been incorporated into the knowledge-based framework enabling to model each facial deformation. The proposed approach has not been widely used in emotion classifiers, but we believe that this technique permits to develop knowledge-base frameworks for facial expression recognition because the analysis of semantics of facial actions may be implemented by using the rule-based descriptors and fuzzy reasoning classifier.

2 Knowledge-Based Framework

The knowledge-based framework allows measuring facial deformations in terms of distances between fiducial points modeled by FAPs and AUs and represented by rule-based descriptors used then in the process of fuzzyfication and interpretation of emotion intensity. The fiducial points represented by FDPs of MPEG-4 standard provides the automatic normalization of measured facial deformations making it invariant to scale of input images. The framework also allows modeling facial deformations defining a set of rules for indexing and quantification of expressions.

The proposed approach is able to detect and measure any type of facial expression however it has been tested using six basic expressions (happiness, sadness, disgust, surprise, anger, and fear). Fig. 1 shows the structure of knowledge-based framework that supports design of a fuzzy reasoning system. We exploit relationships between the measured facial deformations and their mathematical description, by the corresponding AUs and FAPs and rules required for identification of expressions. This knowledge-based framework has been implemented using ontology editor Protégé that provides extensible, flexible, and plug-and-play environment that allows fast prototyping and application development [18]. The proposed knowledge-based framework consists of two abstract super-classes: the *Face_Model* and the *Emotion_Model.* The *Face_Model* class defines different approaches for representation of face features.

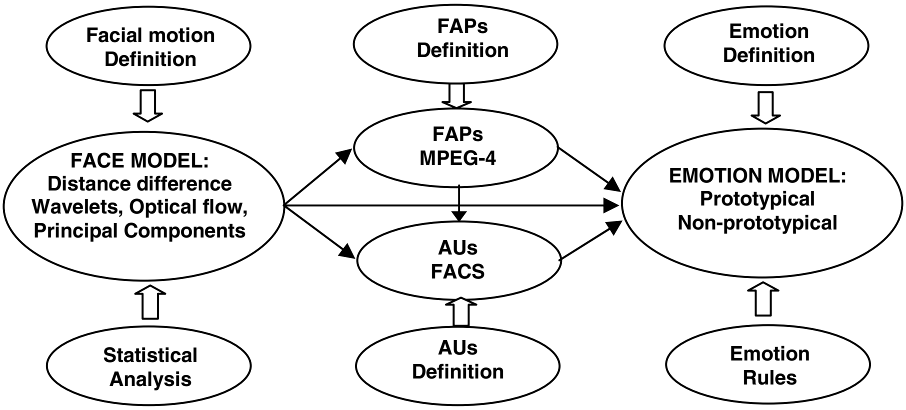
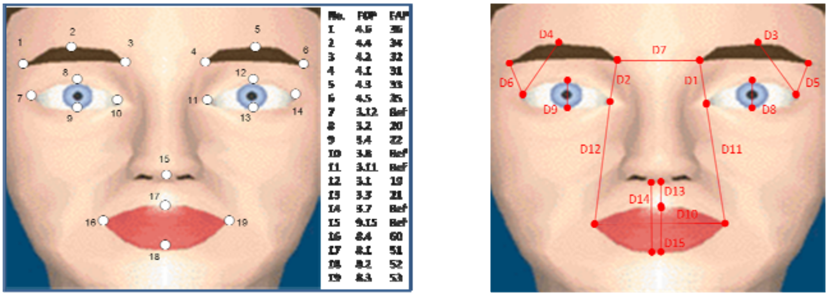


Fig. 1. Structure of emotion knowledge database based on FAPs and AUs

For the framework four specific classes based on AUs, FAPs, FDPs, and FDPs have been created. The *Emotion_Model* class permits to create the rule-based models for emotion indexing using classes of the *Face_Model*. The instances of the particular *Face_Model* class contain the basic facial actions (AUs, FAPs) that include action number, its name, description, direction of motion, involved facial muscles, part of a face where action is occurred, etc. Particularly, we propose emotion indexing based on measuring standard spatial variations of FDP positions implemented by our *Distance_Model*. The framework may be extended with new non-standard classes and models. The advantage of the proposed framework is that the classes and instances with attributes represent knowledge about emotions, and parameters of any model may be automatically converted to parameters of each other.

3 The Proposed Facial Model

In the proposed fuzzy reasoning system the facial model based on analysis of nineteen FDPs has been adopted. It describes all necessary facial actions defining either basic or non-prototypical emotions [13]. Fig. 2 (left) shows the selected FDPs with corresponding number of associated FAPs. Some FDPs are reference points which are remained static during facial deformation. Used FDPs define the *Distance_Class* that represent distances between fiducial reference points and particular FDPs (see Fig. 2). The facial action parameter FAP represents facial changes of emotional expression with respect to the neutral expression. The difference called *Dd* quantifies facial changes in terms of units defined by MPEG-4 standard. Table 1 shows the fifteen instances of the *DistanceClass* chosen for our model, geometric definitions of these distances, the measurement units, the relations with FAPs, and the actions, which they describe. Some reports of Plutchik [12], Pantic, [5], and Esau [4] suggest a geometrical model of face which includes not only distances but also angles between the lines connecting the standard FDPs. However this approach does not contribute significant precision and makes the processing too complex [13, 17].



No.	FDP	FAP
1	4.6	36
2	4.4	34
3	4.2	32
4	4.1	31
5	4.3	33
6	4.5	35
7	3.12	Ref
8	3.2	20
9	3.4	22
10	3.8	Ref
11	3.11	Ref
12	3.1	19
13	3.3	21
14	3.7	Ref
15	9.15	Ref
16	8.4	60
17	8.1	51
18	8.2	52
19	8.3	53

Fig. 2. FDPs used for recognizing facial expressions and definition of *Distance_Class* instances

Table 1. Description of instances in the *DistanceClass*

Dd	FDPs Difference	Units	FDP	Action Description
D1	d{3.11,4.1}	ENS	31	raise l i eyebrow
D2	d{3.8, 4.2}	ENS	32	raise r i eyebrow
D3	d{3.7, 4.3}	ENS	33	raise l m eyebrow
D4	d{3.12, 4.4}	ENS	34	raise r m eyebrow
D5	d{3.7, 4.5}	ENS	35	raise l o eyebrow
D6	d{3.12, 4.6}	ENS	36	raise r o eyebrow
D7	d{4.1, 4.2}	ES		squeeze l/r eyebrow
D8	d{3.3, 3.1}	IRISD	21-19	close t/b l eyelid
D9	d{3.4, 3.2}	IRISD	22-20	close t/b r eyelid
D10	d{8.3, 8.4}	MW	53-54	stretch l/r cornerlip
D11	d{3.11, 8.3}	ENS	59	raise l cornerlip o
D12	d{3.8, 8.4}	ENS	60	raise r cornerlip o
D13	d{9.15, 8.1}	MNS		lower t midlip
D14	d{9.15, 8.2}	MNS		raise b midlip
D15	d{8.1, 8.2}	MNS	51-52	raise b/t midlip

4 Fuzzyfication of Distance Instances

A fundamental process of fuzzy reasoning is fuzzyficacion of input variables and definition of the corresponding membership functions used for indexing facial deformations. The input variables are FAPs representing variation of distances between fiducial points that compose standard database of indexed facial expressions, particularly from Kanade´s and Pantic´s databases [16, 17]. Each database consists of approximately 500 records with expression of different emotions by 100 subjects in frontal position. Accompanying meta-data include annotation of FAC action units and emotion specified expressions. Recorded videos show a series of 23 facial muscle motions that are described by combination of action units (e.g. AU1+AU2 means inner and outer brows raised). Each record begins from a neutral or nearly neutral emotion (neutral face) finishing on a required target emotion. Table 2 shows the results of quantification of the distance differences (see Fig. 2) between fiducial points describing maximum and minimum values, mean, and standard deviation for

Table 2. AUs parameters determined for Kanade´s database

FACs	Distance	Maximum	Minimun	Mean	Deviation
AU1	D1	339.02		120.55	84.52
	D2	383.92		123.44	88.42
AU2	D5	190.23		72.16	60.02
	D6	172.25		35.08	67.66
AU4	D1		-264.20	-42.54	90.92
	D2		-243.63	-38.47	92.88
	D3		-176.41	-31.23	68.42
	D4		-125.68	-6.20	61.99
	D5		-120.53	-35.26	43.40
	D6		-137.58	-29.92	53.46
	D7		-216.24	-67.69	65.20
AU5	D8	429.14		129.51	221.11
	D9	474.65		136.61	243.04
AU7	D8		-677.76	-288.97	171.72
	D9		-511.21	-318.66	148.63
AU10	D13		-294.11	-171.46	85.75
AU12	D10	517.28		273.19	147.06
	D11		-267.11	-129.71	103.15
	D12		-268.58	-140.29	122.95
AU15	D11	438.04		116.17	125.59
	D12	526.43		118.10	152.28
AU16	D14	668.44		306.39	247.81
AU20	D10	345.04		208.07	116.20
	D15	528.24		282.48	144.23
AU25	D15	2248.76		676.64	577.28
AU27	D10		-230.91	-108.40	62.52
	D15	2248.76		1440.71	401.93

each one associated with the particular AU. Recall that the difference in distances is measured between a neutral face and one with any action expressing an emotion. The similar results have been obtained after analysis of emotion representation by AUs using either Kanade´s or Pantic´s database.

From Table 2 we already have quantitative definition of action units, which may be used for continuous interpretation of emotion. For measuring intensity of facial expression the Gaussian function has been used applying the following equation (1):

$$f(x,\sigma,c) = e^{\frac{-(x-c)^2}{2\sigma^2}} \tag{1}$$

The used parameters are determined by mentioned statistical analysis, where *c* defines the position of peak and *σ* controls the width of the bell shaped Gaussian curve.

The fuzzyfication process may be explained analyzing a behavior of any action unit, for example, AU12. According to the results of statistical analysis made for AU12 (see Table 2) the range of its distance variable, for example, D10 is between 0 and 517.20 MWs (Mouth Width). We have defined for each variable as for all ones

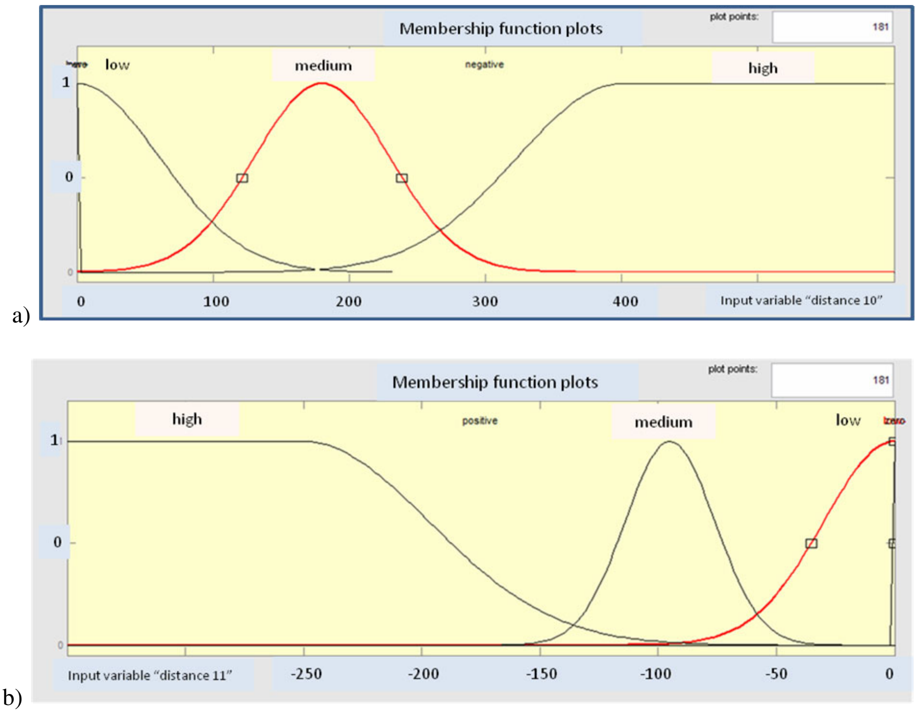


Fig. 3. Membership function plots and intensity partitions for variable a) D10 and b) D11

three levels of emotion intensity (*low+, medium+,* and *high+*) dividing in the corresponding proportion the range of distance variation. These intervals may be computed using the data from Table 2 such as center and width of medium section, the mean, and deviation of D10. Having already defined the middle section, then we compute the Gaussian functions for low and high sections. Additionally, a saturation level is included taking into account the maximum possible value of a facial deformation. Fig. 3 a) and b) show the final process of fuzzyfication for variables D10 and D11. The membership functions are obtained for each partition using Gaussian functions providing measurement of intensity of action unit in continuous manner.

4 Fuzzy Inference System

The proposed model for fuzzyfication of facial features has been tested on designed fuzzy inference system. The designed system (see Fig.4) consists of two modules: the first module measures value of AUs composing analyzed emotion; the second one recognizes and interprets the intensity of facial expressions. A set of rules defined for fuzzy logic that recognizes and measures intensity of AUs and corresponding emotion is shown in Tables 3 and 4.

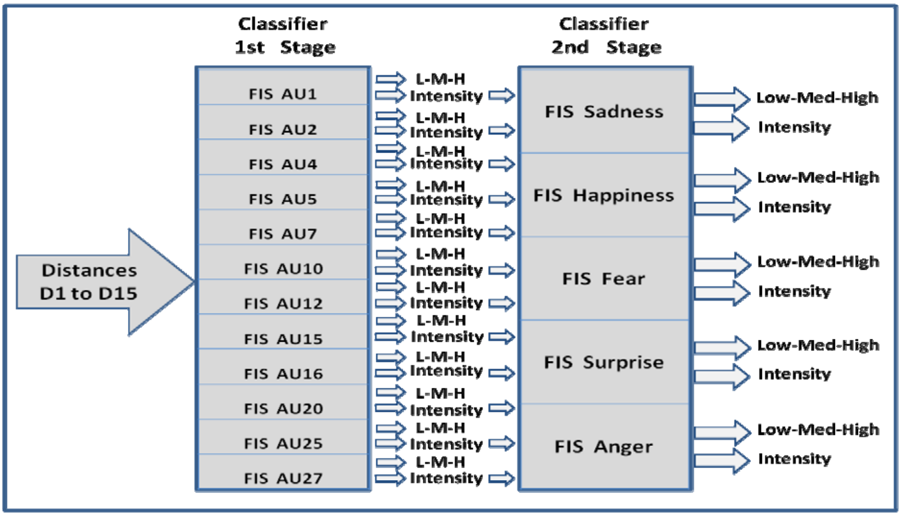


Fig. 4. Block diagram of fuzzy classifier of prototypical emotions

Table 3. Rules and Distance variables for recognizing AUs

Code	Description	Distance Diff.	Recognition Rules
AU1	Inner Brow Raiser	D1, D2	Both increase in same proportion
AU2	Outer Brow Raiser	D5, D6	Both increase in same proportion
AU4	Brow Lowerer	D3, D4, D7	D3&D4 increase, D7 decrease
AU5	Upper Lid Raiser	D8, D9	Both increase in same proportion
AU7	Lid Tightener	D8, D9	Both decrease in same proportion
AU10	Upper Lip Raiser	D13	D13 decrease
AU12	Lip Corner Puller	D10,D11,D12	D10 increase D11 & D12 decrease
AU15	Lip Corner Depressor	D11, D12	Both increase in same proportion
AU16	Lower Lip Depressor	D14	D14 increase
AU20	Lip stretcher	D10, D11, D12	D10, D11&D12 increase
AU25	Lips part	D15	D15 increase
AU27	Mouth Stretch	D10, D15	D10 decrease, D15 increase

Table 4. Rules and AUs for recognizing facial expressions

Emotion	AUs Used	Recognition Rules
Sadness	AU1, AU4, AU15	Increasing 3 actions increase expression intensity
Happiness	AU12, AU7	Presence of AU12 & AU7 but not AU7 (blinking). Increasing values increase expression intensity
Fear	AU1, AU2, AU4, AU5, AU20,AU27	Presence of the 6 actions but not AU7 (blinking). Increasing values increase expression intensity
Surprise	AU1, AU2 AU5, AU27	Presence of the 4th action but not AU5 (blinking). Increasing values increase expression intensity
Anger	AU4, AU7	Presence of AU4 & AU7 but not AU7 (blinking). Increasing values increase expression intensity

In Fig. 5 the user interface of designed fuzzy inference system is shown. In the right upper corner the reasoning process is visualized with intensity of analyzed action unit AU12. The intensity of the input values is tested by classifier applying three discrimination levels described by Gaussian functions: 1-st row in Fig. 5 presents low intensity for all input distances, 2-nd row presents medium and 3-rd - high intensity. The shaded areas correspond to magnitude of the membership functions that describe the contribution of each distance difference to particular emotion.

In some cases the displacement of symmetrical points on a face is different. Thus, it is also measured and shown in 4-th column. The intensity of output variables for the particular action unit presented in 5-th column is modeled by three grades described by the triangular functions instead of Gaussian. This approach is easy to implement and provides fast facial expression recognition without additional errors during interpretation. The proposed model of reasoning is flexible enough to allow its extension incorporating new features for recognition of non-prototypical emotions.

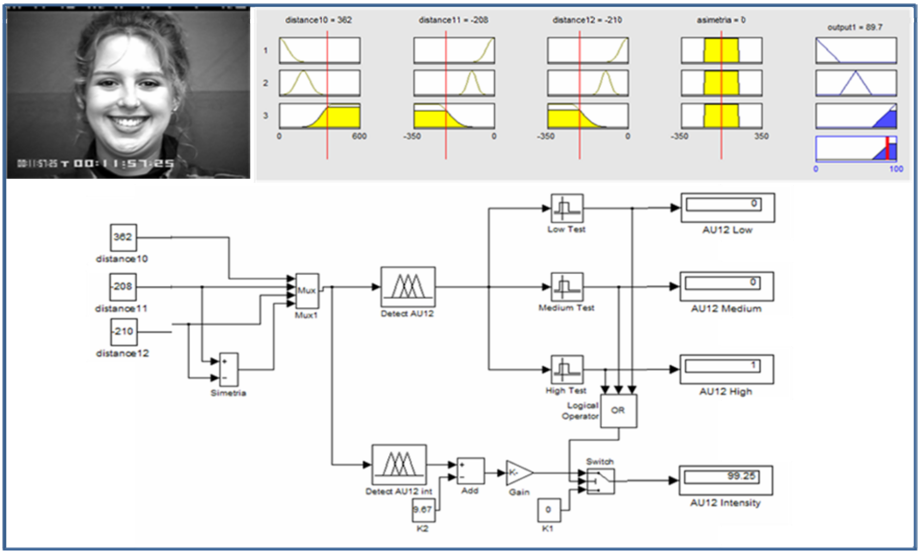


Fig. 5. Measurement of *AU12 Lip Corner Puller* representing happiness of high intensity

4 Obtained Results and Discussion

The test of system performance and efficiency of fuzzyfication model has been done using Kanade´s and Pantic’s databases. Tables 5 and 6 show the confusion matrices obtained for five basic prototypical emotions in case of medium and high intensity. Finally, with regard to correct evaluation of the expression reported by the system, Table 7 shows the comparison between the intensity of expression *Surprise* given by the classifier and reported by evaluation committee of ten persons participated in

Table 5. Confusion Matrix with expression of medium intensity

Emotion	Sadness	Surprise	Happiness	Anger	Fear
Sadness	81%	9.50%	0	0	9.50%
Surprise	0.30%	96%	0	0	3.70%
Happiness	0	0.20%	96%	1.90%	1.90%
Anger	0	4.50%	0.10%	92%	3.40%
Fear	6%	5.80%	0	0	88.20%

Table 6. Confusion Matrix for expressions of high intensity

Emotion	Sadness	Surprise	Happiness	Anger	Fear
Sadness	84%	8%	0	0	8%
Surprise	0.20%	96.40%	0	0	3.40%
Happiness	0	0	97.60%	1.20%	1.20%
Anger	0	1.70%	0	96.70%	1.60%
Fear	4.70%	5.70%	0	0	89.60%

Table 7. Usability test results for *Surprise* emotion

	Output	Evaluation	Status		Output	Evaluat.	Status
1	6.814	Low	OK	11	51.03	Medium	OK
2	50.33	Medium	OK	12	47.7	Medium-	OK
3	51.04	Low	FAIL	13	6.678	Low	OK
4	48.59	Medium	OK	14	50.2	Medium	OK
5	49.85	Medium	OK	15	17.95	Medium	FAIL
6	94.08	High	OK	16	95.12	High	OK
7	69.97	High	OK	17	94.05	High	OK
8	51.46	Medium	OK	18	49.29	Medium	OK
9	93.93	High	OK	19	93.21	High	OK
10	94.94	High	OK	20	93.41	High	OK
Correct assessment :					**90%**		

usability tests. The obtained results indicate a correct assessment of the intensity about 90% for *Surprise* emotion. For other expressions such as joy, sadness, anger, and fear the percentage of corresponding correct recognition is about 80, 85, 77, and 75% respectively. Comparing the results of facial expression recognition with other well-known systems the proposed approach gives average value about 79% against 72% of Esau [4]. The high degree of recognition mainly depends on the number of AUs or FAPs used for description of emotion. The recognition of non-prototypical emotions lies in the range about of 40-60%. This low level of recognition is because of complexity in selection of AUs for representation of non-prototypical emotion and due to subjectivity of its perception by each person. The proposed framework opens new possibility for design of systems for emotion detection and intensity recognition.

5 Conclusions

In this paper we presented a model for fuzzyfication of facial features used for recognition of basic or non-prototypical emotions. For quantification of emotions and their intensities a statistical analysis of Kanade's and Pantic's face database has been made. Two-stage fuzzy inference using Gaussian and triangular functions is applied providing measurement of facial expression intensity. In the preliminary experiments the basic emotion recognition achieves up to 70-90% that depends on complexity in selection of AUs for representation of particular emotion and subjectivity of its perception by each person. The designed knowledge-base framework is general enough to create the diverse instances of emotions, as well as it provides quite exact quantitative description of measured facial actions. This permits simple and formal definition of relationship between emotions, facial actions, and their descriptors. The proposed framework also allows the postulation of rules for prototypical or non-prototypical facial expression recognition using any type of classifiers.

Acknowledgments. This research is sponsored by Mexican National Council of Science and Technology, CONACyT, Projects: #109115 and #109417.

References

1. Young-Joong, K., Myo-Taeg, L.: Near-Optimal Fuzzy Systems Using Polar Clustering: Application. In: Khosla, R., Howlett, R.J., Jain, L.C. (eds.) KES 2005. LNCS (LNAI), vol. 3684, pp. 518–524. Springer, Heidelberg (2005)
2. Yamakawa, T.: Stabilization of an inverted pendulum by a high-speed fuzzy logic controller hardware system, J. Fuzzy Sets and Sys. 32(2), 161–180 (1989)
3. Mufti, M., Khanam, A.: Fuzzy Rule Based Facial Expression Recognition, Computational Intelligence for Modeling, Control and Automation (2006)
4. Esau, N., Wetzel, E. L.: Real-Time Facial Expression Recognition Using a Fuzzy Emotion Model. In: IEEE Fuzzy Systems Conf., pp. 1–6 (2007)
5. Pantic, M.: An Expert System for Multiple Emotional Classification of Facial Expressions. In: 11th IEEE Int. Conf. on Tools with Artif. Intel., p. 113 (1999)
6. Akamatsu, L.S.: Coding facial expressions with Gabor wavelets. McGraw Hill, N.Y. (1998)
7. Kyoung, S.C., Yong-Guk, K., Yang-Bok, L.: Real-Time Expression Recog. Using Active Appearance Model. In: Int. Conf. Comp. Intelligence and Security, China, pp. 1–8 (2006)
8. Lin, D.T.: Facial Expression Classification Using PCA and Hierarchical Radial Basis Function Network. J. Inf. Science and Eng. 22, 1033–1046 (2006)
9. Black, M.J.: Recognizing Facial Expressions in Image Sequences Using Local Parameterized Models of Image Motion. J. of Comp. Vision 25(1), 23–48 (1998)
10. Kharat, G.U., Dudul, S.V.: Neural Network Classifier for Human Emotion Recognition. In: 1-st Int. Conf. on Emerging Trends in Eng. and Techn., Iran, pp. 1–6 (2008)
11. Cohen, I., Garg, A., Huang, T.S.: Emotion Recognition from Facial Expressions using Multilevel HMM. Neural Information Processing Systems, London (2000)
12. Plutchik, R.: The nature of emotions. J. American Scientist 89, 344 (2001)
13. Contreras, R., Starostenko, O.: A Knowledge-base Framework for Analysis of Facial Expressions. In: 10th Conf. on Pat. Recog. and Inf. Proces., Belarus, pp. 251–256 (2009)

14. Ekman, P., Friesen, W.: Facial Action Coding System (FACS). Consulting Psychologists Press, Palo Alto (1978)
15. ISO/IEC14496-2:2001(E), International Standard, Information technology - Coding of audio-visual objects - Part 2, 2nd Ed. (2001)
16. Kanade T., Cohn J.: Comprehensive database for facial expression analysis. In: 4-th IEEE Conf. on Autom. Face and Gesture Recog. France, pp. 46–53 (2000)
17. Pantic, M., Valstar, M.F., Rademaker, R.: Web-based Database for Facial Expression Analysis. In: IEEE Conf. Multmedia and Expo., Netherlands, pp. 1–6 (2005)
18. Ontology editor Protégé (2009), `http://protege.stanford.edu`

Face Recognition Using Simplicial Complexes

Chengming Zou[1] and Edwin R. Hancock[2]

[1] School of Computer Science, Wuhan University of Technology,
Wuhan, Hubei, 430070, China
zoucm@hotmail.com

[2] Department of Computer Science, The University of York, York, YO10 5DD, UK
erh@cs.york.ac.uk

Abstract. The paper presents a novel method for 3D facial shape recognition. Our inputs are 3D facial shapes which are reconstructed from point clouds, and then filtered using PCA. The resulting data are represented by simplicial complexes. This representation can capture topological and geometric information at a specified resolution with a small number of control points. We calculate the Gromov-Hausdorff distance between simplicial complexes, and this measures how far each pair of faces are from being isometric. Finally, we demonstrate our method in an application to point clouds collected from laser range scanner.

Keywords: Simplicial Complexes, Gromov-Hausdorff Distance, 3D Facial Shape, Recognition.

1 Introduction

Face recognition is one of the most significant applications of image understanding. Although rapid progress has been made in this area during the last few years, the general task of recognition remains unsolved. In general, face appearance is influenced not only by intrinsic factors such as shape and texture (albedo), but also by extrinsic ones such as illumination and viewpoint. As a result there have been attempts to compute facial shape invariants that can be used to recognise a face. It has been shown that statistical models based on the distribution of surface normals can offer a powerful means of representing and recognising facial shape. For instance, Smith and Hancock [1] project the surface normals into a tangent space to construct a statistical model using principal geodesic analysis. This work has recently been extended to gender recognition [2], but has proved too cumbersome for expression recognition. Bronstein, Bronstein and Kimmel [3], on the other hand, develop a spherical embedding, that allows faces to be represented in a manner that is invariant to expression. Parameterizing the distribution of surface normals, Kazhdan et al.[4] use the fact that the spherical harmonics of a given frequency form a subspace which is a rotationally invariant. This method can be applied to the extended Gaussian image (EGI) to create a rotationally invariant shape descriptor.

Recently, it has been shown that topological methods provide a robust approach for shape comparison and 3D object recognition, which can also be used

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 22–29, 2010.

for face recognition. Singh, Mmoli and Carlsson[5] use the "Mapper" system for extracting simple descriptions of high dimensional data sets in the form of simplicial complexes. They implement 3D object recognition by comparing the distance between two simplicial complexes. Mmoli and Sapiro[6] present a theoretical and computational framework for isometry invariant recognition of point cloud data based on Gromov-Hausdorff distances. Our aim in this paper is to explore whether such topological descriptions can be used for view and expression invariant face recognition. We commence by converting facial range-data into a simplicial complex and then use the Gromov-Hausdorff distance as a similarity measure.

The outline of the paper is as follows. We commence in section 2 by describing preprocessing the raw data. Section 3 details how to represent the facial shape via simplicial complexes. In section 4 we describe how to recognise face by calculate the Gromov-Hausdorff distance. Experiments are presented in section 5. Finally,section 6 concludes the paper and offer directions for future investigation.

2 Preprocessing

Our raw data is in the form of point clouds collected using a Cyberware 3030 laser range scanner, which is capable of digitising the surface of objects with a resolution of up to 1024 × 1440 points. We use PCA to remove noise from the point cloud data. Let $p_i(x_i, y_i, z_i)$ denote the i^{th} point of a point cloud which has N points. Let $\bar{p} = \frac{1}{N}\sum_{i=1}^{N} p_i$ denote the mean coordinate. The data matrix $A = [(p_1 - \bar{p})\ (p_2 - \bar{p}) \cdots (p_N - \bar{p})]$, is constructed by subtracting the mean coordinate. Then the covariance matrix of the data is $C = AA^T$, which is a 3×3 matrix. Performing an eigen-decomposition on C, yields the eigenvectors v_1, v_2, v_3 and their corresponding eigenvalues $\lambda_1, \lambda_2 \lambda_3$, where $\lambda_1 > \lambda_2 > \lambda_3$. The first eigenvector gives us the leading principal component of the point cloud, and the last eigenvector the least important variance component, which we consider as noise in the point cloud. We therefore use v_3 to remove the noise component $NoiseComponent = v_3 \times A^T$, denoted as $P^{'} = \{p_1^{'}, p_2^{'}, \cdots, p_N^{'}\}, p_i^{'} = (x_i^{'}, y_i^{'}, z_i^{'}), i = 1, 2, \cdots, N$ from the raw data. Then we can find the minimum or maximum value (extrema) of $z_i^{'}, i = 1, 2, \cdots, N$ and its corresponding location, denoted as $z_k^{'}$ and $p_k^{'}$ respectively. From the extremum location can construct the neighbourhood NB of $p_k^{'}$, which contains points that are both close to $p_k^{'}$ and have z coordinates with the same sign as $z_k^{'}$. Removing those points in the neighbourhood NB will implement remove the noisy points from the point cloud.

3 Facial Shape Representation Using Simplicial Complexes

An obvious way to convert a point cloud residing in a metric space into a global object is to use the set of points in the cloud as the vertices of a combinatorial graph whose edges are determined by proximity (i.e. whether vertices are within some specified distance of one-another) [7]. Such a graph, while capturing connectivity information, ignores a wealth of higher order relational features which go beyond defining clusters. We can interpret these features by thinking of the graph as a scaffold for a higher-dimensional object. So to go from a point cloud to a simplicial complex is to embed the point cloud in a piecewise space constructed from simplicies identified (combinatorially) with the faces of the graph. The freedom of choice as to how to fill in the higher dimensional simplicies of the proximity graph allows for different global representations. A natural method for so doing is using $\check{C}$ech theory [7]. To do so we make use of the Rips complex, which is the abstract simplicial complex whose k-simplices correspond to unordered $(k+1)$-tuples of the point cloud which are within a pairwise distance ϵ. Although the Rips complex has more simplices than alternative representations, it is less expensive from a computational point of view. The reason is that the Rips complex is a flag complex, i.e. it is maximal among all simplicial complexes with the given 1-skeleton. Thus, the combinatorics of the 1-skeleton completely determines the complex.

Motivated by the $\check{C}$ech theory, we commence our construction of the simplicial complex representation by firstly locating a covering for the given metric space. Given a point cloud, it is not straightforward to directly locate such a covering. However, for a closed interval $Z = [a, b] \subseteq R$, we may easily locate a covering. Now suppose that we are given a metric space possessing a continuous map $f : X \to Z$ onto Z, and that Z possesses a covering $\mathcal{U} = \{U_\alpha\}_{\alpha \in A}$, again for some finite indexing set A. Since f is continuous, the set $f^{-1}(U_\alpha)$ also forms an open covering of X. For each α, we can now consider the decomposition of $f^{-1}(U_\alpha)$ into its path connected components. So we write $f^{-1}(U_\alpha) = \bigcup_{i=1}^{j_\alpha} V(\alpha, i)$, where j_α is the number of connected components in $f^{-1}(U_\alpha)$. That is to say, we have located a covering $f^{-1}(U_\alpha)$ for X, and we refer to f as a filter.

Our aim in this paper is to represent facial shape using simplicial complexes. In other words, we reduce high dimensional data sets into simplicial complexes with a significant reduction in the number of points needed to represent the topological and geometric information at a specified resolution. To this end, we subsample from the facial shape to reduce the number of points. We then cluster the remaining points using the available geodesic distance matrix and the self organising map (SOM). Then we can construct a simplicial complex by adding a $n-1$-simplex to it whenever n clusters have non-empty intersection.

4 Dissimilarity Calculation

In this section we show how simplicial complexes can be used for facial shape representation and recognition. From the above discussion, the process of face recognition is equivalent to searching for isometric simplicial complexes using the Gromov-Hausdorff. From [8], every facial shape has a "Point fingerprint", which is the set of iso-depth contours around the nose tip. Since faces are of different size, we do not need to measure the whole facial shape. Instead we need only measure the point fingerprint region. If two point fingerprints are isometric, then the corresponding facial shapes correspond to the same individual with the same or different expression. So to recognise a face, we need only to calculate a sub graph corresponding to the point fingerprint region. We refer to this sub graph as the fingerprint graph.

Consider two fingerprint graphs $G_1(V_1, E_1)$ and $G_2(V_2, E_2)$. From the above preprocessing steps, the two fingerprint graphs form a sub-space of the spherical metric space. As a result the Gromov-Hausdorff distance between G_1 and G_2 is

$$d_{\mathcal{GH}}(G_1, G_2) = \inf max\{\sup_{u\in V_1} \inf_{v\in V_2} d(u,v), \sup_{v\in V_2} \inf_{u\in V_1} d(u,v)\} \tag{1}$$

The Gromov-Hausdorff distance $d_{\mathcal{GH}}(G_1, G_2)$ measures the dissimilarity of two facial shapes. By setting a distance threshold τ, when $d_{\mathcal{GH}}(G_1, G_2) \leq \tau$ then two facial shapes are deemed to be isometric or the same facial shape with the same or different expression.

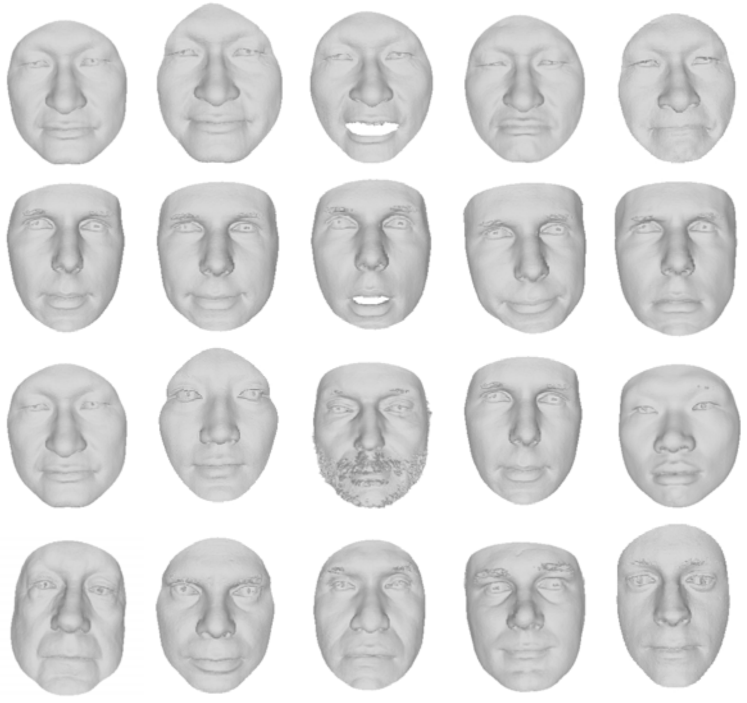


Fig. 1. Facial shape after preprocessing, first row shows facial shape A with its five expressions, the second row shows facial shape D with its five expressions, the third row and last row are facial shapes A,B,C,D,E,F,G,H,I,J with their neutral expression.

5 Experiments

We have performed preliminary experiments on several 3D range data-sets of faces. The experiments are performed by using VS2005 and MatLab. We have performed three sets of experiments, aimed at evaluating the performance of the three steps of the method. Each experiment is performed on ten individuals, labeled with the letters A-J, who present five facial expressions (neutral, smile, laugh, sad, surprised). In each case we commence with point clouds containing 46149 vertices. After subsampling, 5829 vertices remain. A sample of the experimental results is shown in Figure 1. It is clear that the reconstruction is accurate and that our method can effectively filter noise from the data.

Experimental results with simplicial complex representations are shown in Figure 2. From Figure 2, it is clear that the facial shapes of the different subjects give rise to distinct simplicial complex representations. Moreover, the different expressions from the same individual give rise to similar simplicial complexes.

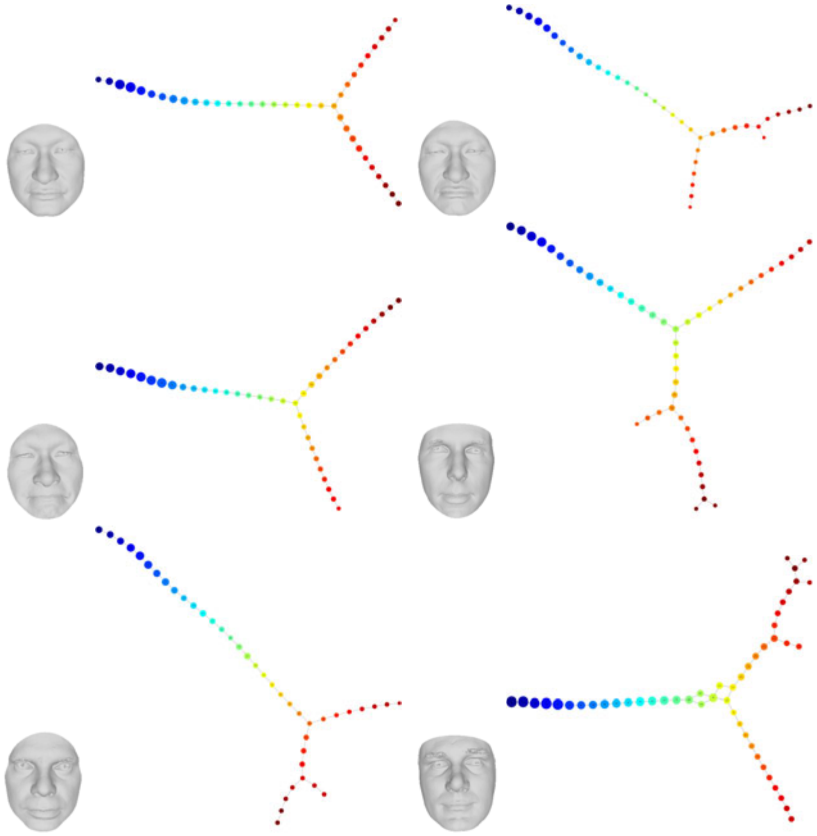
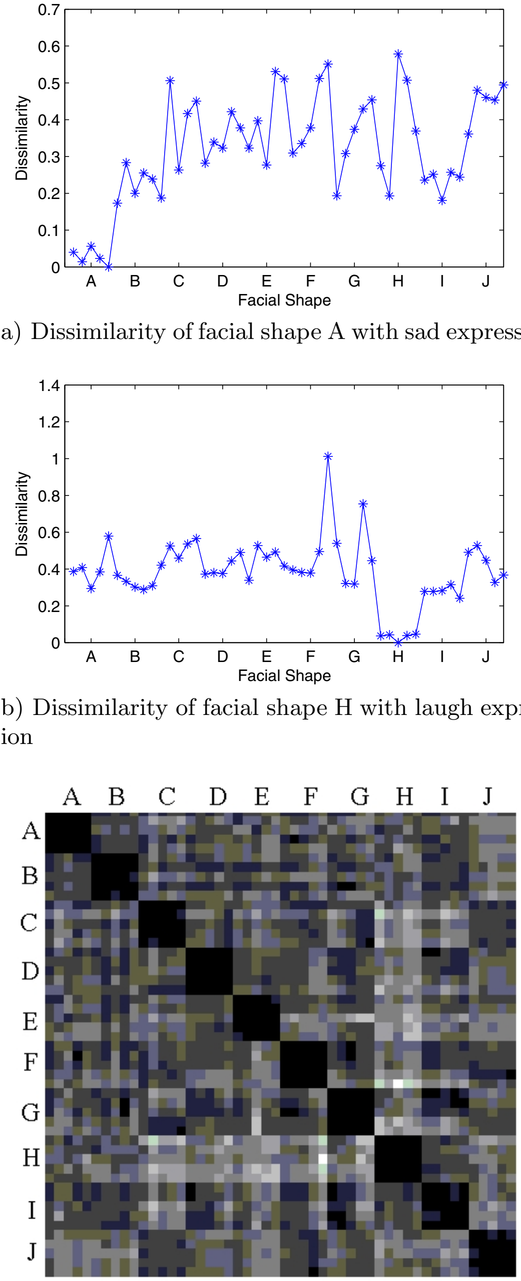


Fig. 2. Some facial shapes and their simplicial complex representation visualized by GraphViz



(a) Dissimilarity of facial shape A with sad expression

(b) Dissimilarity of facial shape H with laugh expression

(c) The dissimilarity matrix

Fig. 3. Facial shape dissimilarity, all the dissimilarities have been divided by the maximum value 100

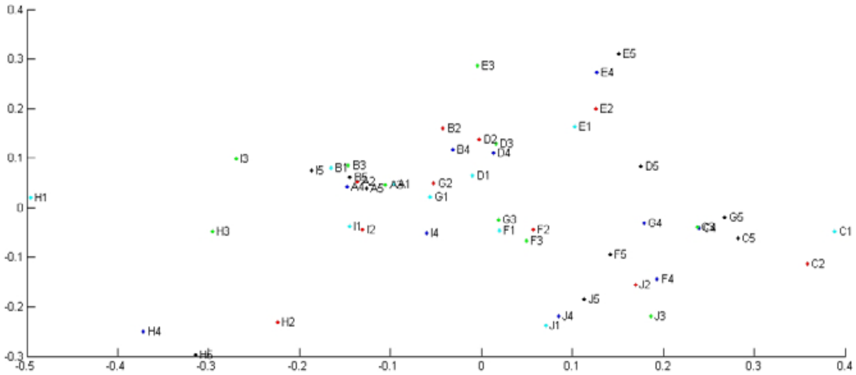


Fig. 4. The dissimilarity multidimensional scaling

That is say the simplicial complex is an effective means of expression invariant face recognition. Moreover the representation is based on just 5829 points, and this represents a data reduction factor of 15%.

In Figure 3 we explore the properties of the dissimilarity measure. Figs 3a and 3b show the distribution of distances for two expressions (face A with a sad expression and face H with a laugh expression). In both the cases the distances for the same subject are the smallest. Figure 3c shows the distance matrix for the complete set of similarity data. Here the different expressions for the same subject appear as distinct blocks.

To take this study one step further, in Figure 4 we embed the different expressions of the different faces into a 2 dimensional space using multidimensional scaling. Each face is denoted by a alphanumeric symbol (e.g. G2) to denote the subject and the expression. From the figure it is clear that the different expressions of the same subject project into the same region. In other words the Gromov-Hausdorff distance can be used to distinguish subjects in an expression invariant manner.

6 Conclusions

We have introduced a theoretical and computational framework for expression invariant facial shape representation and recognition. Experimental results have shown that our method for 3D facial shape representation and recognition is feasible and effective. Much work remains to be done in the future, especially in recognising the facial shape recovered from the photos, and verifying the validity of our method by doing more detailed experiments.

References

1. Smith, W.A.P., Hancock, E.R.: Facial Shape-from-shading and Recognition Using Principal Geodesic Analysis and Robust Statistics. Int. Journal Computer Vision 76, 71–91 (2008)

2. Wu, J., Smith, W.A.P., Hancock, E.R.: Weighted Principal Geodesic Analysis for Facial Gender Classification. In: Rueda, L., Mery, D., Kittler, J. (eds.) CIARP 2007. LNCS, vol. 4756, pp. 331–339. Springer, Heidelberg (2007)
3. Bronstein, A., Bronstein, M., Kimmel, R.: Expression invariant face recognition via spherical embedding. Image Processing, ICIP 3, 756–759 (2005)
4. Kazhdan, M., Funkhouser, T., Rusinkiewicz, S.: Rotation Invariant Spherical harmonic representation of 3D shape descriptors. In: Proceedings of the 2003 Eurographics, vol. 43, pp. 126–164 (2003)
5. Singh, G., Mmoli, F., Carlsson, G.: Topological Methods for the Analysis of High Dimensional Data Sets and 3D Object Recognition. In: Eurographics Symposium on Point-Based Graphics (2007)
6. Mmoli, F., Sapiro, G.: A Theoretical and Computational Framework for Isometry Invariant Recognition of Point Cloud Data. Foundations of Computational Mathematics 5, 313–347 (2005)
7. Ghrist, R.: Barcodes: the persistent topology of data. In: AMS Current Events Bulletin, 2007 Joint Mathematics Meetings, New orleans, LA, January 7-8 (2007)
8. Sun, Y., Abidi, M.: Surface matching by 3d point's fingerprint. In: International Conference on Computer Vision, ICCV 2001, pp. 263–269 (July 2001)

Skeleton Simplification by Key Points Identification

Gabriel Rojas-Albarracín[1], Carlos A. Carbajal[1],
Antonio Fernández-Caballero[1,2], and María T. López[1,2]

[1] Instituto de Investigación en Informática de Albacete, 02071-Albacete, Spain
[2] Universidad de Castilla-La Mancha, Departamento de Sistemas Informáticos,
02071-Albacete, Spain
caballer@dsi.uclm.es

Abstract. The current skeletonisation algorithms, based on thinning, extract the morphological features of an object in an image but the skeletonized objects are coarsely presented. This paper proposes an algorithm which goes beyond that approach by changing the coarse line segments into perfect "straight" line segments, obtaining points, angles, line segment size and proportions. Our technique is applied in the post-processing phase of the skeleton, which improves it no matter which skeletonisation technique is used, as long as the structure is made with one-pixel width continuous line segments. This proposal is a first step towards human activity recognition through the analysis of human poses represented by their skeletons.

Keywords: Thinning, skeletonisation, image post-processing.

1 Introduction

Pattern recognition has been and continues to be one of the main lines of research in Artificial Intelligence, especially in the areas of Natural Language processing (voice recognition) and Computer Vision (face and human emotion recognition, handwriting recognition, document classification and many more) through biometric parameters. Some of these techniques are based on the principles found in Psychology (visual intelligence), Biology (human anatomical features), Mathematics, Physics and Statistics. From a human perspective, an object can be recognized by looking straight at it or by looking at a simplified image of it. In the field of Artificial Vision, one of the ways to improve the process of object recognition is through the skeletonisation of the image or of the points of interest to be identified in the object, so that from that or those points of interest, recognition can take place. This step reduces the amount of data to be processed, thus reducing the time spent in object recognition.

The algorithm proposed in this paper is a complement to the post-processing phase of the skeletonized image, in which the skeleton is perfected through the elimination of isolated pixels and the substitution into straight line segments. At the same time, they provide points, angles, line segment size and proportions,

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 30–39, 2010.

which are valid and feasible results for image analysis. The algorithm was tested using two well-known skeletonisation algorithms in different images obtained from different sources by applying a pre-processing. This proposal is an initial step towards human activity recognition through the analysis of human poses represented by their skeletons. Aside from this initial section, the rest of the paper is made up of 4 additional sections. In section 2, there is an outline of some important skeletonisation concepts and the algorithms used in the tests. The details of the proposed algorithm are described in section 3. Section 4 shows the results obtained in the tests carried out. Finally, in section 5, we conclude with observations and recommendations for future works.

2 Skeletonisation

In an image, skeletons are very useful for the recognition of elongated objects or patterns that have a certain shape, such as characters, polygons, chromosomal patterns, etc. Skeletons provide an extraction of topological and geometrical features of the object, so that when it is stored and processed, certain structural information about the original object is considered. Skeletonisation can be seen as a data compression process. The concept of skeleton was introduced by Blum in 1967 [6] in his analogy of middle axis detection with a grass fire. Since then, his definition has been used as a model for skeletonisation. A great number of techniques to obtain skeletons from discrete objects have been developed in the fields of Computer Vision and Pattern Recognition. Said techniques can be grouped into four different classes [7]: topological reduction [19] [28], distance transformation [9] [12] [20], curve evolution [16] [25] [27] and computational geometry [2] [22] [21] methods.

The skeletonisation technique based on topological reduction is frequently used to get the skeleton from a shape or object through thinning. Thinning is the reduction process of a digital image made up of certain number of pixels into a simplified version based on single-pixel-width line segments, so that the elimination of said point will not affect image connectivity and will respect the local end-point property in such a way that the topological properties of the object are preserved. In other words, after the pixels have been removed, the pattern must be recognized. The thinned version of a shape is called a skeleton. Fig. 1 shows different types of matrices (rectangular, hexagonal and triangular) used for pixel analysis [11] [10]. Likewise, sequential [15] [23] [18] and parallel [30] [14] [8] implementations have been published. In sequential algorithms, the pixels are eliminated in every iteration in a fixed sequence. The exclusion of a point p in iteration n depends on all operations executed until then. On the other hand, in parallel algorithms, the elimination of iteration n depends solely on the pixels of iteration $n-1$. Therefore, all pixels can be analyzed independently in parallel to each iteration.

Generally a rectangular matrix topology is used to generate a topological reduction. On the one hand, topological reduction can guarantee connected skeletons. However, most reduction algorithms can not always guarantee perfectly thinned shapes, since there will be cases where an array of pixels cannot be

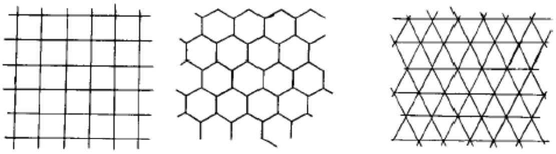


Fig. 1. Matrices used for pixel analysis. (a) Rectangular matrix. (b) Hexagonal matrix. (c) Triangular matrix.

more thoroughly eroded. Moreover, these methods are seriously affected when objects undergo rotation. Nonetheless, skeletons produced with most techniques are sensitive to noise or to a variation of boundary, which often generates redundant skeleton branches that can alter the topology of the skeleton. To offset this problem, many skeletonisation algorithms include pruning methods, which appear as application-dependent [3]. Krinidis and Chatzis [17] have recently worked in an algorithm, without the use of any pruning methods, which does not generate spurious branches.

Aslan et al. [1] presented a different skeletal representation which deals with the problem of shape recognition with local deformations (see Fig. 2). Said algorithm relies on the stable features of the shape, instead of on the secondary inaccurately measured details. Therefore, the generated skeleton works with disconnected branches. The new representation does not suffer the common instability of the traditional connected skeletons, thus producing descriptions that are sensitive to any combination of changes in scale, position, orientation and articulation, as well as invariant ones. This way, the skeletons produced are similar, even when image boundaries undergo deformation or when there is a change in scale or rotation. From these data (location of disconnection and its length or branch), we can define primitives that can attain shape recognition through trees or skeletal graphs, where shape dissimilarity is calculated through distance correction.

The most important challenge for skeletal similarity is probably the fact that, on the one hand, the topological structure of trees or skeletal graphs of similar objects can be completely different (as a consequence of not taking into account the context). On the other hand, skeletal graphs of different objects can have

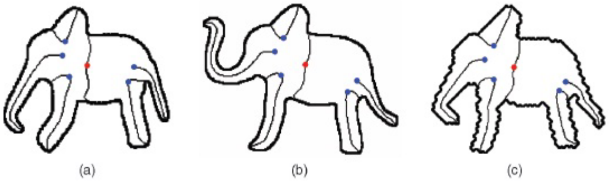


Fig. 2. Disconnected skeletons for the elephant in images with different rotation and borders. Notice that the branch and the location of the disconnection (indicated by a point) are similar [1].

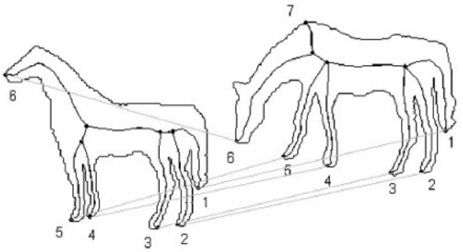


Fig. 3. The corresponding end nodes between the two skeleton graphs are linked with lines [4]

similar topology. To tackle these problems, Baseski, Erdem and Tari [5] exhibit an approach in which the contextual effects are considered relevant information for the calculation of skeletal similarity without being directly related to the geometric properties of the compared form. Likewise, Bai and Latecki's [4] main idea is to match skeletal graphs comparing geodesic paths between the skeleton's end-points without thinking about the topological structure of the graphic (see Fig. 3c).

Rizvandi, Pizurica and Philip [24], in their method for the detection of worms, decompose the skeleton into branches through the elimination of junction pixels (pixels with more than two neighbors), then calculate the angles for all branches and compare the angles of neighboring branches. Neighboring branches with an angle difference of less than a threshold are connected. Thus, a series of points (final, connecting and junction) and branches (final and connective) are defined.

3 Key Point Identification Algorithm

Our algorithm simplifies the skeletons previously obtained through any reduction technique in which skeletal thickness has a maximum width of one pixel. For instance, in Fig. 4a we show one input sample of the "LEMS 99 Silhouette Database", and in Fig. 4b you have the output of the Hilditch skeletonisation algorithm. Our proposal is part of the post-processing phase, which is applicable to skeletons. Notice that when an image from a two-dimensional object is thinned, the resulting skeleton has an irregular shape, based on arcs and curves. We expect to take that image and simplify it into points, obtaining angles, line segment sizes and proportions. For verification purposes, once the skeleton is simplified, we trace said points into perfect "straight" line segments by changing the coarse line segments of the original skeleton. We decompose an image into line segments. To do this, we define:

- End-point: A pixel (point) of the skeleton with only one neighbor.
- Intersecting-point: A pixel (point) in which two or more line segments cross or intersect.

These are key points within the structure of the skeleton. Our tests reveal a decrease in the amount of information necessary to represent a skeletonized

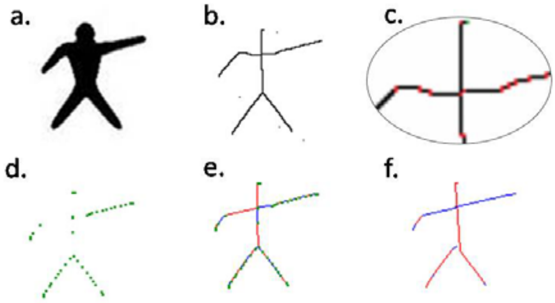


Fig. 4. (a) A sample from the "LEMS 99 Silhouette Database". (b) Result of the Hilditch skeletonisation algorithm. (c) Result of step "Finding out straight line segments". (d) Result of step "Finding out key points". (e) Result of step "Joining key points". (f) Result of step "Joining resulting line segments".

image, thus allowing us to center image analysis on said points. It is necessary to start from a previously obtained skeleton before running our processing algorithms. Any already well-known thinning algorithm can be used. Starting from that thinned image, we reduce the skeleton to points (expressed as pixel coordinates), and later we proceed to reconstruct the skeleton. Our algorithm consists of 4 significant steps, namely, finding out straight line segments, finding out key points, joining key points and joining resulting line segments.

3.1 Finding Out Straight Line Segments

The image is decomposed into straight line segments l, made up of consecutive pixels aligned in the same direction, containing a minimum of 2 pixels to represent a line segment. The slope for each line segment is yielded by function $m(l)$. For this, we have defined 4 directions: horizontal, vertical, and diagonal line segments slanting to the right and to the left. Then, two consecutive points $(x1, y1)$ and $(x2, y2)$ are aligned if:

- Horizontal (0°): $x2 = (x1 + 1)$ and $y2 = y1$
- Vertical (90°): $x2 = x1$ and $y2 = (y1 + 1)$
- Diagonal slanting to the right (45°): $x2 = (x1 + 1)$ and $y2 = (y1 + 1)$
- Diagonal slanting to the left (135°): $x2 = (x1 - 1)$ and $y2 = (y1 + 1)$

Through an iterative process, we search for continuous pixels all in the same direction, obtaining this way line segments longer or equal to 2 pixels. All the line segments found are stored with their starting (x_s, y_s) and final (x_f, y_f) coordinates in a set as shown in equation 1. In Fig. 4c, we show the output of this first step of the algorithm. Notice that all the disconnected points have been eliminated.

$$L = \{l^1[p^1(x_s, y_s), p^1(x_f, y_f)], ..., l^{max_l}[p^{max_l}(x_s, y_s), p^{max_l}(x_f, y_f)]\} \tag{1}$$

3.2 Finding Out Key Points

In this step, two types of key points, $p_k(x,y)$, are detected: end-points, $p_e(x,y)$, and intersecting-points, $p_\cap(x,y)$. End-points are those that have only one neighboring skeleton pixel. That is to say, the points where the continuity of the skeleton ends. This way $p_e(x,y)$ is an end-point if the 8-connectivity of pixel (x,y) is equal to 1 neighbor. All end-points are stored in set $P_e = \{p_e^1(x,y), ..., p_e^{max_e}(x,y)\}$. Also, the line segments related to all end-points are stored as sets $R_e(p_e^\kappa) = \{l^i, ..., l^j\}$.

The other group of key points is that of the intersections. This group represents the points where two or more line segments with different slopes cross. Let l' be a line segment yielded by points $p_1'(x_1', y_1'), p_2'(x_2', y_2')$, and l'' a line segment yielded by points $p_1''(x_1'', y_1''), p_2''(x_2'', y_2'')$. $p_\cap(x,y)$ would be an intersection point for l' and l'' if,

$$m(l') \neq m(l'') \tag{2}$$

obtaining x and y coordinates as:

$$x = \frac{(y_1'' - y_1') + (m(l') \times x_1') - (m(l'') \times x_1'')}{m(l') - m(l'')} \tag{3}$$

$$y = m(l') \times (x - x_1'') + y_1' \tag{4}$$

if, and only if, point $p_\cap(x,y)$ coincides with:

$$\begin{aligned} x_1' \leq x \leq x_2' \\ y_1' \leq y \leq y_2' \\ x_1'' \leq x \leq x_2'' \\ y_1'' \leq y \leq y_2'' \end{aligned} \tag{5}$$

All the intersecting are stored in another set $P_\cap = \{p_\cap^1(x,y), ..., p_\cap^{max_\cap}(x,y)\}$. Also, the line segments related to all intersecting-points are stored as sets $R_\cap(p_\cap^\mu)$ $= \{l^k, ..., l^n\}$. Fig. 4d shows the 40 key points detected for our running example at this step.

3.3 Joining Key Points

In this step, we get the most significant result when the two key points are joined (end-points and intersecting-points), generating new line segments (l^u) that allow us to represent the original structure with a lot less information. To do this, a line segment is created between each pair of points if, and only if, there is a path between them and there are no key points between them. The process to join two key points is:

```
$L' = \emptyset$
$\forall p_k(x, y) \in P_k$
  $\forall l^i \in R_k(p_k(x, y))$
    $\forall l^j \in L, l^j \neq l^i$
    $if \;\; \exists l^j | (p^i(x_f, y_f) \in l^i) \; and \; (p^j(x_s, y_s) \in l^j) \; are \; 8 - connected \; then$
      $if \;\; l^j(x_s, y_s) \in R_k \; then$
        $L' = L' \cup \{l^u[p^i(x_s, y_s), p^j(x_f, y_f)]\}$
      $else$
        $l^i = \{l^u[p^i(x_s, y_s), p^j(x_f, y_f)]\}$
    $else$
      $L' = L' \;\cup l^i$
```

To start, we define a set (L') to store the new line segments. Then, each line segments associated to some key point is compared to each line segment in the figure (L). That comparison allows to determinate if the line segments are 8-connected. In this case, a new line segment is created from their union. Finally, the line segment is stored in L'. Fig. 4e shows the way the key points have been joined trough the algorithm at this step.

3.4 Joining Resulting Line Segments

This is a polishing step where we detect line segments that can join and become one, from among the line segments generated in the previous step, and similarly, two line segments will join if both are 8-connected and have the same slope, resulting one line segment made up of the two most distant points, from among the four points that characterize both line segments. Fig. 4f shows the 14 resulting line segments.

4 Data and Results

The shapes from the 99−silhoutte database [24] were used by Goh [13], Sebastian, Klein, and Kimia [26], among others, in their experiments. In our case, the simplification algorithm was tested in 10 images (Fig. 5a), in which the skeletonisation algorithms previously described (Fig. 5b and e) were applied and from which the resulting images were obtained. Finally, the process results in a list of coordinates that make up the resulting line segments. Therefore, we can obtain line segments' sizes, proportions between line segments and their respective angles of inclination, and specially a skeleton with the morphology of the original structure but with less information (Fig. 5c and f). In the same way, Fig. 5 in columns (c) and (g) shows the reduction rate obtained by our technique. The reduction is calculated by dividing the number of lines needed to represent the skeleton after our process by the number of the original lines.

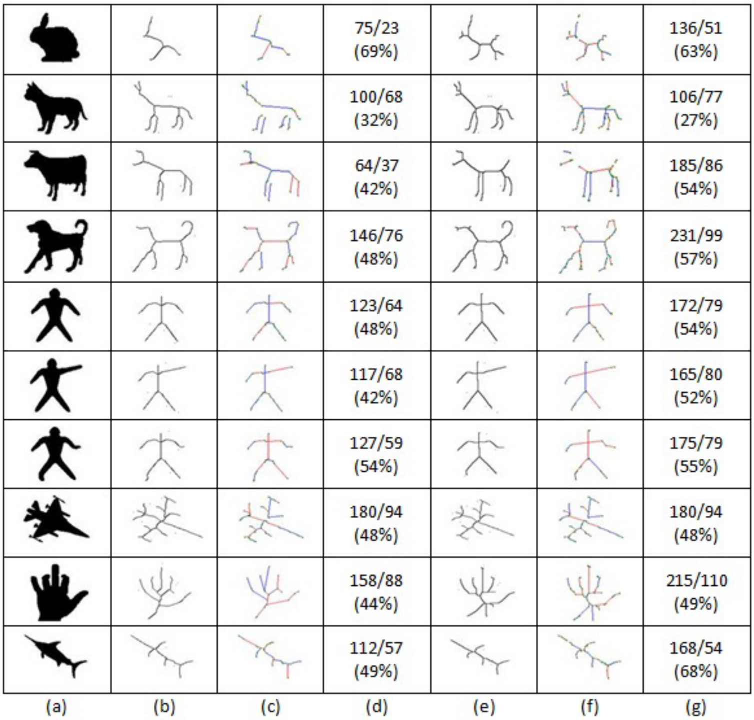


Fig. 5. (a) Shapes used in the experiments. (b) Results of the Zhang-Suen skeletonisation. (c) Final results of our algorithms on Zhang-Suen skeletonisation. (d) Reduction rate compared to the Zhang-Suen algorithm. (e) Results of the Hilditch skeletonisation. (f) Final results of our algorithms on Hilditch skeletonisation. (g) Reduction rate compared to the Hilditch algorithm.

5 Conclusions

The use of skeletons and their subsequent polishing allows for data compression, reducing the need for storing, as well as improving the quality of the information stored, since it dismisses irrelevant data generated by common skeletonisation algorithms, such as isolated pixels. Many methods to skeletonized images have been developed. Goh [13] recently proposed an image comparison method using skeletons. This paper proposes a method that allows us to simplify, not only data from a skeleton, but also its subsequent analysis, resulting in a series of related coordinates, which represent the skeletonized image in a reliable way without the computational cost of analyzing a complete image.

We have shown that the skeleton represented in the simplified image can be reconstructed, with a high degree of accuracy, based on the points (coordinates) generated in the simplification process. The problems that have come up are previous to algorithm skeletonisation, specifically to the generation of line segments that are more than one pixel wide. In future works, the effectiveness of the method will be considered when the skeletonized image undergoes alterations related to rotation, position and scale. The refinement method will also extend to three-dimensional images, keeping the method as simple as possible. Lastly, remember that the algorithms proposed are a first step towards human activity recognition through the analysis of human poses represented by their skeletons.

Acknowledgements

This work was partially supported by Spanish Ministerio de Ciencia e Innovación under projects TIN2007-67586-C02-02 and TIN2010-20845-C03-01, and by Junta de Comunidades de Castilla-La Mancha under projects PII2I09-0069-0994, PII2I09-0071-3947 and PEII09-0054-9581. The authors are grateful for the public use of the *LEMS 99 Silhouette Database*.

References

1. Aslan, C., Erdem, A., Erdem, E., Tari, S.: Disconnected skeleton: Shape at its absolute scale. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(12), 2188–2203 (2008)
2. Aurenhammer, F.: Voronoi diagrams: A survey of a fundamental geometric data structure. ACM Computing Surveys 23(3), 345–405 (1991)
3. Bai, X., Latecki, L.J., Liu, W.Y.: Skeleton pruning by contour partitioning with discrete curve evolution. IEEE Transactions on Pattern Analysis and Machine Intelligence 29(3), 449–462 (2007)
4. Bai, X., Latecki, L.J.: Path similarity skeleton graph matching. IEEE Transactions on Pattern Analysis and Machine Intelligence 30(7), 1282–1292 (2008)
5. Baseski, E., Erdem, A., Tari, S.: Dissimilarity between two skeletal trees in a context. Pattern Recognition 42(3), 370–385 (2008)
6. Blum, H.: A transformation for extracting new descriptors of shape. In: Models for the Perception of Speech and Visual Form, pp. 153–171. MIT Press, Cambridge (1967)
7. Bouix, S., Siddiqi, K.: Optics, mechanics, and Hamilton-Jacobi skeletons. Advances in Imaging and Electron Physics 135, 1–39 (2005)
8. Chin, R.T., Wan, H.K., Stover, D.L.: A one-pass thinning algorithm and its parallel implementation. Computer Vision, Graphics, and Image Processing 40(1), 30–40 (1987)
9. Danielsson, P.: Euclidean distance mapping. Computer Vision, Graphics, and Image Processing 14, 227–248 (1980)
10. Davies, E.R., Plummer, A.P.N.: Thinning algorithms: A critique and a new methodology. Pattern Recognition 14, 53–63 (1981)
11. Deutsch, E.S.: Thinning algorithms on rectangular, hexagonal, and triangular arrays. Communications of the ACM 15(9), 827–837 (1972)
12. Ge, Y., Fitzpatrick, J.M.: On the generation of skeletons from discrete euclidean distance maps. IEEE Transactions on Pattern Analysis and Machine Intelligence 18(11), 1055–1066 (1996)
13. Goh, W.B.: Strategies for shape matching using skeletons. Computer Vision and Image Undertanding 110(3), 326–345 (2008)

14. Hall, R.W.: Fast parallel thinning algorithms: Parallel speed and connectivity preservation. Communications of the ACM 32(1), 124–131 (1989)
15. Hilditch, C.: Linear skeletons from square cupboards. Machine Intelligence 4, 403–420 (1969)
16. Kimia, B.B., Tannenbaum, A.R., Zucker, S.W.: Shapes, shocks, and deformations I: The components of two-dimensional shape and the reaction-diffusion space. International Journal of Computer Vision 15(3), 189–224 (1995)
17. Krinidis, S., Chatzis, V.: A skeleton family generator via physics-based deformable models. IEEE Transactions on Image Processing 18(1), 1–11 (2008)
18. Kwok, P.C.K.: A thinning algorithm by contour generation. Communications of the ACM 31(11), 1314–1324 (1988)
19. Lam, L., Lee, S.W., Suen, C.Y.: Thinning methodologies: A comprehensive survey. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(9), 869–885 (1992)
20. Leymarie, F., Levine, M.D.: Simulating the grassfire transform using an active contour model. IEEE Transactions on Pattern Analysis and Machine Intelligence 14(1), 56–75 (1992)
21. Ogniewicz, R.L., Ilg, M.: Voronoi skeletons: Theory and applications. In: Proc. Conference on Computer Vision and Pattern Recognition, pp. 63–69 (1992)
22. Ogniewicz, R.L., Kübler, O.: Hierarchic voronoi skeletons. Pattern Recognition 28(3), 343–359 (1995)
23. Pavlidis, T.: A thinning algorithm for discrete binary images. Computer Vision, Graphics, and Image Processing 13, 142–157 (1980)
24. Rizvandi, N.B., Pizurica, A., Philips, W.: Automatic individual detection and separation of multiple overlapped nematode worms using skeleton analysis. In: Campilho, A., Kamel, M.S. (eds.) ICIAR 2008. LNCS, vol. 5112, pp. 817–826. Springer, Heidelberg (2008)
25. Scott, G.L., Turner, S.C., Zisserman, A.: Using a mixed wave/diffusion process to elicit the symmetry set. Image and Vision Computing 7(1), 63–70 (1989)
26. Sebastian, T.B., Klein, P.N., Kimia, B.B.: Recognition of shapes by editing shock graphs. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(5), 550–571 (2004)
27. Tari, S., Shah, J., Pien, H.: Extraction of shape skeletons from gray-scale images. Computer Vision and Image Understanding 66(2), 133–146 (1997)
28. Xie, W., Thompson, R.P., Perucchio, R.: A topology-preserving parallel 3D thinning algorithm for extracting the curve skeleton. Pattern Recognition 36(7), 1529–1544 (2003)
29. Xu, W., Wang, C.X.: A fast thinning algorithm implemented on a sequential computer. IEEE Systems, Man, and Cybernetics 17(5), 847–851 (1987)
30. Zhang, T.Y., Suen, C.Y.: A fast parallel algorithm for thinning digital patterns. Communications of the ACM 27(3), 236–239 (1984)

Learning and Fast Object Recognition in Robot Skill Acquisition: A New Method

I. Lopez-Juarez[1], R. Rios-Cabrera[1], M. Peña-Cabrera[2], and R. Osorio-Comparan[2]

[1] Centro de Investigación y de Estudios Avanzados del I.P.N. (CINVESTAV)
Ramos Arizpe. CP 25900. Coah. México
{ismael.lopez,reyes.rios}@cinvestav.edu.mx
[2] Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas,
Universidad Nacional Autónoma de México (UNAM)
Apdo. Postal 20-726, México DF, México
{mario,roman}@leibniz.iimas.unam.mx

Abstract. Invariant object recognition aims at recognising an object independently of its position, scale and orientation. This is important in robot skill acquisition during grasping operations especially when working in unstructured environments. In this paper we present an approach to aid the learning of manipulative skills on-line. We introduce and approach based on an ANN for object learning and recognition using a descriptive vector built on recurrent patterns. Experimental learning results using a fast camera are presented. Some simple parts (i.e. circular, squared and radiused-square) were used for comparing different connectionist models (Backpropagation, Perceptron and FuzzyARTMAP) and to select the appropriate model. Later during experiments, complex figures were learned using the chosen FuzzyARTMAP algorithm showing a 93.8% overall efficiency and 100% recognition rate with not so complex parts. Recognition times were lower than 1 ms, which clearly indicates the suitability of the approach to be implemented in robotic real-world operations.

Keywords: ART Theory, Artificial Neural Networks, Invariant Object Recognition, Machine Vision, Robotics.

1 Introduction

Grasping and assembly operations using industrial robots is currently based on the accuracy of the robot and the precise knowledge of the environment, i.e. information about the geometry of assembly parts and their localization in the workspace. Techniques are sought to provide self-adaptation for robots. This document reports a neural-based methodology for invariant object recognition applied to self-adapting industrial robots which can perform assembly tasks. New objects can also be learned quickly if certain clues are given to the learner, since the methodology uses only two on-line patterns for learning complex objects. The architecture is firstly trained with clues representing different objects that the robot is likely to encounter (and with others that represent complex objects) within the working space to form its initial

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 40–49, 2010.

knowledge base. The main idea suggests that it is possible to get fast and reliable information from a simple but focused analysis of what an object might show. The very important aspects of the scene (we have called "clues"), can be used later to retrieve memorized aspects of the object without having to recall detailed features. By using neural networks it is possible to learn manipulative skills which can be used by an industrial manipulator. In someway we humans do that process once an object has been seen and learned for the first time.

The article describes a robust method for very fast learning, perimeter and centroid calculations, object functions and pose estimation. The remainder of this paper is structured as follows. Section 2 reviews related work and state our contribution to the field of self-adaptive industrial robots for assembly and object recognition. In Section 3, the analysis of three ANN's is presented for selecting the appropriate connectionist model while in section 4 the methodology is explained. Experimental results from several object learning and recognition tasks are given in section 5. Finally, conclusions and future work is described in section 6.

2 Background and Related Work

2.1 Related Work

Many authors considered only constraint motion control during assembly; however, to complete the autonomy of the assembly system a machine vision system has also to be considered. Hoska, introduced the concept of "Robotic Fixtureless Assembly" (RFA), that eliminates the need of using complex and rigid fixtures, which involves new technical challenges, but allows potential solutions [1]. Ngyyuen and Mills, [2] have studied RFA of flexible parts with a dynamic model of two robots with a proposed algorithm, which does not require measurements of the part deflections. The goal of RFA is to replace these fixtures with sensor-guided robots which can work within RFA workcells. Using ANNs, an integrated intelligent vision-guided system can be achieved as it is shown by Langley, [3]. Many authors had come with descriptor vectors and image transformations, used as general methods for computer vision applications in order to extract invariant features from shapes. Aguado, et al. [4] developed a methodology for including invariance in general form of the Hough transform, Chin-Hsiung, et al. [5] designed a technique for computing shape moments based on the quadtree representation of images. Best and McKay [6] described a method for registration of 3D shapes in minutes. Bribiesca, [7] developed a new chain code for shapes composed of regular cells, which has recently evolved even to represent 3D paths and knots. But the methods require a multiple-pattern input for training. Some authors use multiple cameras/views to extract information, to perform invariant object recognition and determine object's position and motion, Gonzalez-Galvan, et al. [8] developed a procedure for precision measure in 3D rigid-body positioning using camera-space manipulation for robot operation. Dickmanns [9], Kollnig and Nagel, [10] have shown solutions to facilitate the use of vision for real world-interaction. Applications of guided vision used for assembly are well illustrated by

Bone and Capson, [11] which developed a vision-guide fixtureless assembly system using a 2D computer vision for robust grasping and a 3D computer vision to align parts prior to mating.

A previous method proposed by Peña-Cabrera [12] reported 100% of object recognition; however, there were used simple geometry (i.e. circular, squared and radiused-square pegs) and 72 patterns for training each of the three objects. The testbed was formed basically by a 6 DOF KUKA KR15 robot, master computer and a JR3 F/T sensor attached to the robot's wrist to form a feedback loop while assembling components. The F/T sensor was used only for alignment during part mating after the assembly component was recognised by the vision system (the reader is referred to [13] for details). The vision system was composed by a CCD Pulnix 6710 camera with 640x480 pixel resolution. POSE information was provided by the vision system to the master computer to generate the robot motion commands for component grasping. Once the assembly component was held by the robot, then the vision system also determined the assembly block location for assembly. Figure 1 shows the assembly components used for recognition and learning of robot manipulative tasks (i.e. assembly operations).

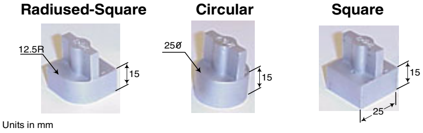


Fig. 1. Assembly Components

2.2 Original Work

The research presented in this paper is focused on learning and object recognition of simple and complex 2D objects in order to enable industrial robots to learn manipulative tasks (i.e. assembly operation). In this area moment invariants are still popular descriptors for image regions and boundary segments, but computation of moments of a two dimensional (2D) image involves a significant amount of multiplications and additions in a direct method. The computation of moments can be simplified since it contains only the information about the shape of the image as proposed by Chen, [14]. This paper proposes recurrent pattern vector descriptors, using collections of 2D images to obtain a very fast feature data of an object by using image projections and mirror images. Fast learning is achieved on-line considering only two patterns of simple –geometrical- and complex objects to achieve invariance to rotation, translation and scale. The fast algorithm allows calculation of a boundary object function (BOF) and centroid which defines object information, and also considers variance normalized grey-colour intensity properties, which in conjunction with an ANN forms a system called SIRIO which recognizes, learns and performs pose estimation of assembly components in the order of milliseconds what constitutes a practical tool for real-world robot applications.

3 Neural Networks Evaluation

The next subsections show the result in experimental comparisons among BP (Backpropagation), P (Perceptron), and FAM (Fuzzy ARTMAP) in order to select an appropriate model for the vision system. For training and testing purposes, the shapes of the patterns shown in figure 1 were used. Figure 2 shows the patterns.

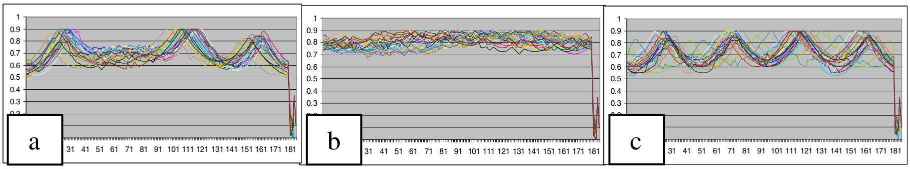


Fig. 2. a) radiused-square; b) Circle, c) square. Total=216 patterns (72 from each shape)

All ANN's were programmed in Visual C++ .NET using a Pentium D PC @ 2.8 GHz with 512 MB RAM.

3.1 Backpropagation

BP is a stochastic steepest descent learning rule used to train single or multiple layer nonlinear networks. The algorithm overcomes some limitations of the perceptron rule by providing a framework for computing the weights of hidden layer neurons, but also it takes more time for training/testing. The configuration was: layer_In=185, layer_hidden=200, layer out=4, the weights were selected layer 1,2: Random weight [-2.0,2.0], and layer 3, Random weight [-1.0,1.0]. The learning rate α=0.7 and maximum error allowed was 0.12. (Patterns used are in figure 2). Other experiments were done using, topology 185-300-4, learn rate= 0.85, 185-250-4 but they showed less efficiency.

Since this ANN depends on weights randomly selected, five experiments where done, and the average of the experiments is considered for comparison purposes. The figure 3a, shows the performance of the ANN in learning. Since there were only 3 objects to recognize, the classification starts with a 64.16% error (the error means the % of patterns that are not recognized from the universe of 216). The best case in 675 epochs reached 0% error in 57.593 s (training), in the worse case 1332 epochs in 116.375 s.

3.2 Perceptron

It is a feedforward network with one or more outputs that learn the position of a separating hyperplane in pattern space (a layer for nonlinearly separable pattern pairs). The first layer has fixed weights and the second ones change according to the error in the output. If a neuron shows no error, its weights are not modified. This architecture was considered, because it can reach always a no-linear patterns classification properly with the enough number of neurons in layer 1. In [15] it was demonstrated that

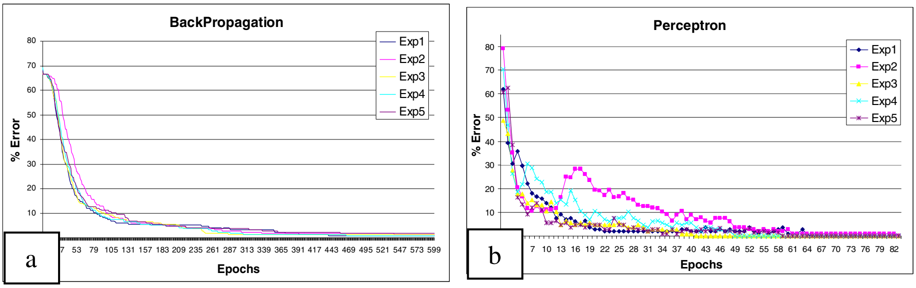


Fig. 3. a) Backpropagation performance (epochs are showed only until 600), b) Perceptron

one hidden layer is sufficient for performing any arbitrary transformation, using enough nodes. This model was considered for comparison also because of its high speed in training/testing compared with BP. Five experiments were done. The configuration used was: Inputs: 185, Outputs: 4, Layer1: 450 neurons (10 C/N=Connections per neuron), Random weights [-2.0, 2.0], Layer2: 4 neurons, Threshold: 0.0 (both layers, signum threshold device), α= 0.85. Other experiments were done using 450 (6, 8, 10, 12 C/N)-4, 350 (6, 8, 10, 12 C/N)-4, showing very similar results. The behaviour starts with an average error of 63.8%. (See figure 3b) This ANN showed a much better performance than the BP, in the best case reached 0% error in 41 epochs and a training time: 0.8112 s, the worse case it took 83 epochs, with 1.734 s in training.

3.3 Fuzzy ARTMAP

Fuzzy ARTMAP developed by Carpenter. G. A, [16] is one of the architectures based on ART (Adaptive Resonance Theory) in which supervised learning is carried out. This ANN creates several neurons according to the number of patterns and the differences among them. It has several advantages compared with traditional neural networks such as Backpropagation, since it does not suffer catastrophic lose of knowledge. The configuration for this architecture was, 2 epochs, rho_map=0.8; β= 1.0; ρ_a = 0.7; ρ_b = 1.0; aF1Size= 185; bF1Size=4. For all experiments α was set to 0.1. Five experiments were carried out, but since this ANN does not depend on random values, the same results were obtained in epochs/%error.

This ANN starts from an error of 100%, but reached fast 0% error in one epoch. The best training time was 0.172 s and the worst 0.188 s. See figure 4a. For all cases, the generated Knowledge Base (internal representation of the network) showed the same behaviour: For the circle one neuron was generated, and for the square and radiused-square, 2 neurons. This is because it was configured to a maximum data compression. Figure 4b, is a 100 epochs experiment with the same patterns, varying parameters β, ρ_a, α and rho_map. This experiment was done for testing the stability of the network and for choosing the best parameters for the proposed methodology. The graphic shows that the combinations of the parameters in the middle part of the graph do not reach stability and the network continues creating new neurons. Generally, β and ρ_a determine the stability. The property of FAM encoding critical features is a key

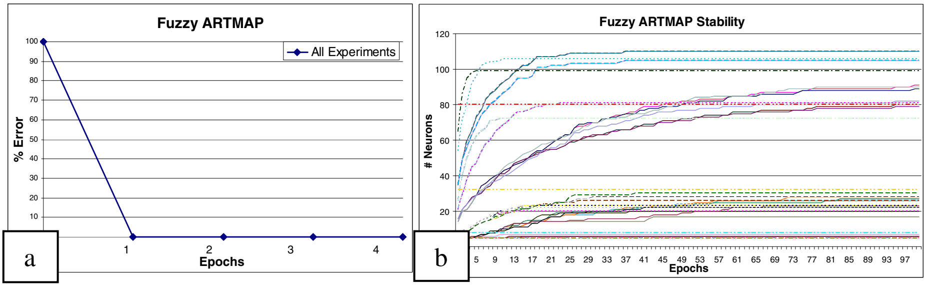


Fig. 4. a) FuzzyARTMAP, b) FAM Stability-sensibility parameters

to code stability. This learning strategy makes the difference of ART networks and MLP's (Multi Layer Perceptrons), which typically encode the current input, rather than a matched pattern, and hence, employ slow learning across many input trials to avoid catastrophic forgetting. [17].

3.4 Evaluation

Table 1 shows the results of the experiments. Time average is showed in figure 5.

Table 1. Results of the ANN experiments. Time is given in seconds and it is training/testing all patterns.

ANN	Exp1 Train	Exp1 Test	Exp2 Train	Exp2 Test	Exp3 Train	Exp3 Test	Exp4 Train	Exp4 Test	Exp5 Train	Exp5 Test	Average Train	Average Test
BACKPROPAGATION	79.625	0.047	116.375	0.047	68.734	0.047	57.593	0.047	74.657	0.047	79.3968	0.047
PERCEPTRON	1.391	0.031	1.734	0.031	0.8112	0.016	1.109	0.047	1.203	0.047	1.24964	0.0344
FUZZY ARTMAP	0.188	0.015	0.187	0.015	0.187	0.016	0.172	0.016	0.172	0.016	0.1812	0.0156

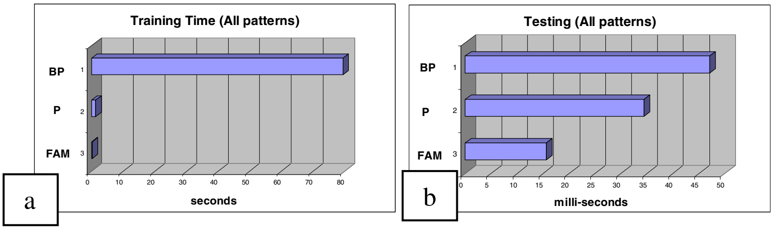


Fig. 5. a) ANN's Training time - b) ANN's Testing time. **BP** showed a Train-Test time of 367.577ms-0.217 ms per pattern, the **P,** 5.78ms-0.159 ms per pattern, and **FAM** 0.838 ms -- 0.0722 ms per pattern.

FAM was chosen because of its incremental knowledge capabilities and stability, but mostly because of the fast recognition and geometrical classification responses.

4 Object Recognition Methodology

4.1 First Approach

The Boundary Object Function (BOF) is the function that describes a specific shape (distances perimeter-centroid). In [12] a vector called CFD&POSE was proposed, eq (1), where***:*** $\boldsymbol{D_i}$ distances centroid-perimeter, $\boldsymbol{X_C}$***,*** $\boldsymbol{Y_C}$***,*** centroid, $\boldsymbol{\phi}$, orientation, $\boldsymbol{Z}$ is the object's height (for this experiments, 3D objects were used), ***ID*** is a code number related to the geometry of the components.

$$[CFD\,\&\,POSE] = [D_1, D_2, D_3, D_n, X_c, Y_c, \phi, Z, ID]^T \tag{1}$$

This method was tested with 3 shapes for on-line recognition, the results showed training/testing time of few ms and a 100% classification. However, the methodology needed several patterns for training, then adding a new object would not be fast.

The hypothesis of the new SIRIO is: *With only two patterns from a regular or irregular object, it is possible to learn and recognise the object on-line achieving invariance to: rotation, location and scaling of a 2D image representation of an object.* The basic idea is to imitate the human learning/recognition ability, since humans need only to observe the object once or twice irrespective of its size/orientation/position in order to remember it later.

4.2 Patterns Generation –Proposed Approach

In order to subtract only parameters from the object, we take the BOF and grey-scale characteristics: the average grey colour and the object-histogram. We suppose a robotic fixtureless scenario in which there is no specific guide or line to find the orientation of a piece. We can approach a line through the piece and determine the orientation angle, but we can obtain only a value 0-180° (in both cases 0° to 180° or 180° to 360°). This behaviour will generate two different patterns for the same object that can not be rotated and rearranged automatically for the lack of the real angle. In the conveyor belt, it is also possible for the robot to find an object that is the mirror image of the trained one (the other object's side). In order to classify those mirror images also in the ANN, it is necessary to reproduce those in the inverse order during the training stage in such a way to have a new pattern to classify (2 on-line images and 2 mirror images) see Fig 6a. With these initial patterns, it is possible now to rearrange all other not trained patterns with a rotation function based on the found orientation and to generate very similar patterns. For managing the size invariance, normalization is applied. The grey-scale intensity average is an important characteristic of an object, and it is included in the object-properties vector as well as its histogram. The histogram no matter the size, orientation or light conditions (within a working range), has implicit information about the object and behaves similarly. It is proposed the vector of eq. (2), where **I** is the grey-scale intensity average value redundantly copied N times in the vector, and **H** is the object-histogram considering only K points of its behaviour and **P** is another possible invariant property. The figure 6b shows the generation of a vector with the properties of a wrench. The descriptor vector uses 180 BOF data, 10 Intensity data, and 30 data coming from function histogram, all data is normalized to [0,1].

$$[BOF + I_{1...n} + H_{1...K} + P] \qquad (2)$$

For the first experiments, the whole pattern sets were made of [BOF + I], and the histogram was not used in order to test part of the vector only. Later other experiments where done, using the H, for the group of similar figures.

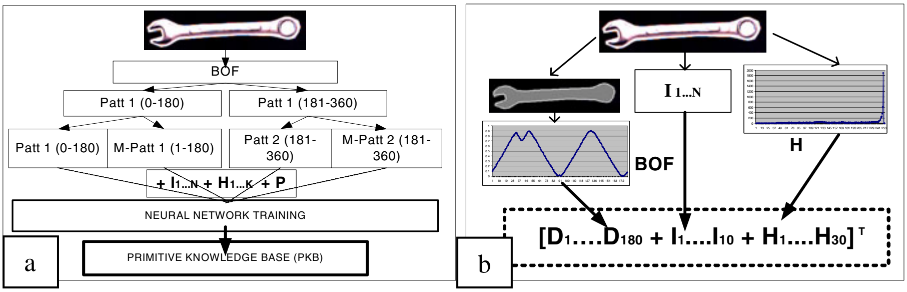


Fig. 6. a) PKB Base generation using two simple patterns. b) Descriptor vector generation

Figure 7 shows the universe of images used for the experiments and some of the patterns generated.

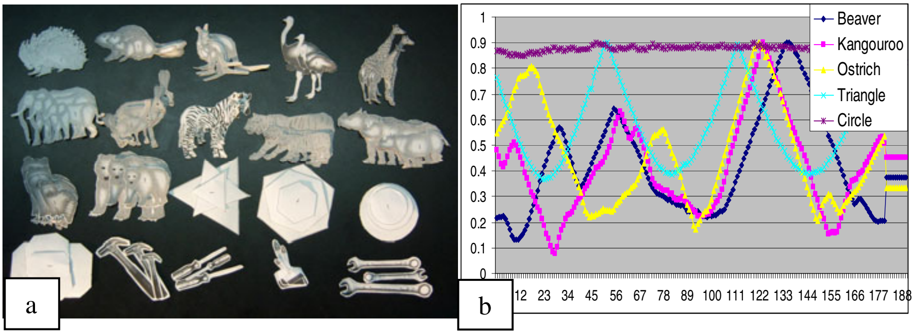


Fig. 7. a) 20 different figures: small, medium and big (total = 60), b) Patterns for 5 figures

One of the advantages of the algorithm is that while finding the perimeter and marking the object, the centroid, mass, histogram, etc, are calculated at the same time. Once this is completed, then the descriptor vectors are generated. All data is normalized to the range [0, 1], and the number of points fixed to vector size.

5 Experiments and Results

Algorithms were coded in Visual C++ .NET using a Pentium D PC @ 2.8 GHz with 512 MB RAM. For taking the images in real-time, a high speed, 100 fps, IEEE 1394 Basler f602c 640x480 resolution camera, was used. From the universe of 20 different

Table 2. For geometric pieces 100% recognition was achieved. An overall average recognition of 93.8% was achieved. Only for 6 out of 20 objects the percentage was not 100% (similar objects). If we consider only the No Mirror part (side 1) for testing, the achievement is 96.3%. Training time was 1.95 ms per pattern and 0.2 ms for testing. FAM parameters were: 2 epochs, rho_map=0.92; β= 1.0; ρa = 0.9; ρb = 1.0. Err=Error, T=total, NM= No Mirror, N=Neurons.

#	Object	Code ANN	Err Side 1	Err Side 2	T. Err	% OK	%OK NM	Generated N
1	Porcupine	1 0 0 0 0 0	0	0	0	100.0	100.0	{0, 1, 2, 3}
2	Beaver	1 0 0 0 0 1	0	0	0	100.0	100.0	{4, 5, 6, 7}
3	Kangouroo	1 0 0 0 1 0	0	0	0	100.0	100.0	{8, 9, 10, 11}
4	Ostrich	1 0 0 0 1 1	0	0	0	100.0	100.0	{12, 13, 14, 15}
5	Giraffe	1 0 0 1 0 0	1	3	4	83.3	91.7	{16, 17, 18, 19}
6	Elephant	1 0 0 1 0 1	2	4	6	75.0	83.3	{20, 21, 22, 23}
7	Hare	1 0 0 1 1 0	0	1	1	95.8	100.0	{24, 25, 26, 27}
8	Zebra	1 0 0 1 1 1	1	3	4	83.3	91.7	{28, 29, 30, 31}
9	Tiger	1 0 1 0 0 0	1	2	3	87.5	91.7	{32, 33, 34, 35}
10	Rhinoceros	1 0 1 0 0 1	2	4	6	75.0	83.3	{36, 37, 38, 39}
11	Raccoon	1 0 1 0 1 0	0	0	0	100.0	100.0	{40, 41, 42, 43}
12	Bear	1 0 1 0 1 1	2	4	6	75.0	83.3	{44, 45, 46, 47}
13	Triangle	1 0 1 1 0 0	0	0	0	100.0	100.0	{48, 49, 50}
14	Hexagon	1 0 1 1 0 1	0	0	0	100.0	100.0	{51}
15	circle	1 0 1 1 1 0	0	0	0	100.0	100.0	{52}
16	Parabola	1 0 1 1 1 1	0	0	0	100.0	100.0	{53, 54}
17	Hammer	1 1 0 0 0 0	0	0	0	100.0	100.0	{55, 56, 57, 58}
18	Drill	1 1 0 0 0 1	0	0	0	100.0	100.0	{59, 60, 61}
19	Brush	1 1 0 0 1 0	0	0	0	100.0	100.0	{62, 63, 64, 65}
20	Wrench	1 1 0 0 1 1	0	0	0	100.0	100.0	{66, 67, 68}
				Total	30	93.8	96.3	

objects (three sizes for each one) as observed in figure 7a, 20 different middle size were selected for training purposes, placing each object in two different angles (values within 1° to 180° and 181° to 360° range). Two patterns were generated, as well as their mirror images (40 images were captured on-line and 40 mirror images, for the 20 different figures) for creating the PKB in the ANN. For the first experiment only [BOF + I] was used in order to evaluate de performance. After training, the experiments consisted of testing the 60 pieces, even though they where not trained, with different location, and orientation. For testing: Size 1, 2, 3, Angle 45, 135, 225, 315, Sides: 1, 2, Total=480 patterns were used. The results are given in Table 2.

Most of the errors in table 2 were detected in the animals with similar bodies, (Tiger, Bear, Zebra, etc), this is because the ANN confused in some conditions this animals among them. The solution of this problem could be adding more patterns, but then the hypothesis will not be fulfilled. The other option is to add the object histogram to complete the descriptor vector. For this group of animals, the complete vector was generated, new experiments were done and the results were successfully achieved. We can conclude that if we have very similar objects it is necessary to use the complete vector and for simple or medium complex figures with the basic form is enough. We can see in objects 14 and 15 that the ANN created only one neuron for recognizing all patterns of its group, because of the object simplicity. 68 neurons were created for recognizing the universe of 20 objects, 3 sizes, all angles.

For sending information to the robot once in the manufacturing cell, a secondary vector is considered, (based on the recognized object and a data base related to it) in eq. (3), where $\mathbf{C_x}$, $\mathbf{C_y}$ is the POSE of the object, $\mathbf{\Phi}$ is the orientation angle, **ID** is the identified object, and **OI** is Object Information related to the object grasping taken from the data base.

$$[C_x\, C_y + \Phi + ID + OI] \tag{3}$$

6 Conclusions and Future Work

An invariant method using 2D image object representation suitable for part grasping during robot operations was presented. The method proposes to use of only two image patterns from a regular or irregular object instead of using multiple patterns for training. From the given results it was demonstrated that it is possible to learn and recognise the objects on-line achieving invariance to: rotation, location and scaling.

The method showed potential fast recognition and accuracy in the classification of simple and complex 2D objects. The superiority of FAM for this task was also experimentally demonstrated compared to other connectionist models. For future work it is intended to prove the algorithms in the robotic testbed using varying light conditions.

References

1. Hoska, D.: Fixturless assembly manufacturing. Manufacturing Eng. 100, 49–54 (1988)
2. Ngyuen, W., Mills, J.K.: Multirobot control for flexible fixturless assembly of flexible sheet metal autobody parts. In: IEEE Int. Conf. Robotics & Automation, pp. 2340–2345 (1996)
3. Langley, C.S., Eleuterio, D.: GMT, A memory efficient neural network for robotic pose estimation. In: Proc. of IEEE Int. Symp.on Computational Intelligence in Robotics and Automation, vol. 1, pp. 418–423. IEEE CIRA (2003)
4. Alberto, S., Aguado, E., Montiel, M.: Invariant characterization of the Hough Transform for pose estimation of arbitrary shapes. Pattern Recognition 35, 1083–1097 (2002)
5. Wu, C.-H., et al.: A new computation of shape moments via quadtree decomposition. Pattern Recognition 34, 1319–1330 (2001)
6. Best, P.J., McKay, N.D.: A Method for Registration of 3-D Shapes. IEEE Trans. on Pattern Analysis and Machine Intelligence 14(2) (1992)
7. Bribiesca, E.: A new Chain Code. Pattern Recognition 32, 235–251 (1999)
8. Gonzalez-Galvan, E.J., et al.: Application of Precision-Enhancing Measure in 3D Rigid-Body Positioning using Camera-Space Manipulation. The International Journal of Robotics Research 16(2), 240–257 (1997)
9. Dickmanns, E.: Vehicles capable of dynamic vision: a new breed of technical beings? Artifical Intelligence 103, 49–76 (1998)
10. Kollnig, H., Nagel, H.: 3d pose estimation by directly matching polyhedral models to gray value gradients. Int. Journal of Comp.Vision 23(3), 282–302 (1997)
11. Bone, G.M., Capson, D.: Vision-guided fixturless assembly of automotive components. Robotics and Computer Integrated Manufacturing 19, 79–87 (2003)
12. Peña-Cabrera, M., Lopez-Juarez, I., Rios-Cabrera, R., Corona-Castuera, J.: Machine vision approach for robotic assembly. Journal of Assembly Automation 25(3), 204–216 (2005)
13. Corona-Castuera, J., Lopez-Juarez, I.: Behaviour-based approach for skill acquisition during assembly operations, starting from scratch. Robotica 24(6), 657–671 (2006)
14. Chen, K.: Efficient parallel algorithms for computation of two-dimensional image moments. Pattern Recognition 23, 109–119 (1990)
15. Cybenko, G.: Approximation by superposition of a Sigmoidal Function. Mathematics of Control, Signals, and Systems 2, 303–314 (1989)
16. Carpenter Gail, A., et al.: Fuzzy ARTMAP: A Neural Network Architecture for Incremental Supervised Learning of Analog Multidimensional Maps. IEEE Transactions on Neural Networks 3(5) (1992)
17. Carpenter, G.A., Grossberg, S.: Adaptive Resonance Theory. In: The Handbook of Brain theory and neural networks, 2nd edn. MIT Press, Cambridge (2002)

Light Source Intensity Adjustment for Enhanced Feature Extraction

Francisco J. Castro-Martínez*, Mario Castelán, and Ismael López-Juárez

Centro de Investigación y de Estudios Avanzados del I.P.N.
Robotics and Advanced Manufacturing Group,
Ramos Arizpe, Coahuila, 25900, México
{francisco.castro,mario.castelan,ismael.lopez}@cinvestav.edu.mx

Abstract. We explore the automatic adjustment of an artificial light source intensity for the purposes of image-based feature extraction and recognition. Two histogram-based criteria are proposed to achieve this adjustment: a two-class separation measure for 2D features and a Gaussian distribution measure for 2.5D features. To this end, the light source intensity is varied within a fixed interval as a camera captures one image for each intensity variation. The image that best satisfies the criteria for feature extraction is tested on a neural-network based recognition system. The network considers information related to both 2D (contour) and 2.5D shape (local surface curvature) of different objects. Experimental tests performed during different times of the day confirm that the proposed adjustment delivers improved feature extraction, extending the recognition capabilities of the system and adding robustness against changes in ambient light.

Keywords: Object recognition, neural networks, feature extraction.

1 Introduction

It is well known that illumination is a highly important factor in computer vision tasks. Basically, if knowledge is to be obtained from visual data, little can be inferred from a scene whose objects are illuminated with a too intense or too low light source. The impact of illumination in computer vision has been explored for many purposes. The Photometric Stereo Method (PSM) [12] may be the best (and probably oldest) example to get some benefit from changes in light source direction in order to obtain 3D information of the surface observed by the camera. More recently, stereopsis has also exploited the idea of illumination variations in order to redefine the correspondence problem under the Light Transport Constancy (LTC) constraint [11], with encouraging results. Variations in intensity of the light source have also proved to be useful in obtaining the 3D

* The authors wish to thank CONACyT for the financial support given to the project through grant No. 61373 and also for the scholarship granted to Francisco J. Castro-Martínez during his Master studies.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 50–60, 2010.

surface of objects, as shown in [10], where Light Fall-off Stereo (LFS) is employed to estimate 3D surface as a light source increasingly departs from the illuminated object. Researchers on the field of robotics have also shown interest in the automatic positioning of light sources so as to maximize luminance and contrast in the image for the purposes of tracking [8].

Studies have demonstrated that the appearance of convex surfaces such as human faces is more dependable on changes in direction and intensity of the light source than on changes in facial pose. [9]. In general, many computer vision problems make assumptions on (generally Lambertian) reflectance properties of objects as well as good illumination conditions, i.e., a negligible effect of ambient light and inter-reflections. In reality, even when the reflectance of a material can be approximated with a Lambertian reflectance model, the intensity of the light source plays a key role in revealing appropriate levels of reflectance for each pixel, therefore facilitating the task of feature extraction from pixel values.

In this paper, we do not address the problem of 3D shape recovery nor automatic light source positioning, rather, we focus on the adjustment of the light source intensity for the task of 2D and 2.5D feature extraction. To this end, two main criteria based on gray level histogram are used. In a first criterion, we draw on the ideas of Otsu's method [13] in order to decide whether a light source intensity is suitable for background segmentation. The second criterion searches for the best Gaussian distribution of a gray level histogram and is related to the construction of shape-index histograms for local surface orientation estimation [5]. This improved-from-illumination feature extraction benefits recognition rates in a neural network system.

The organization of the paper is as follows: in Section 2, the object recognition system is roughly described. Section 3 provides an explanation of the two main criteria used for feature extraction and light source intensity regulation. Experiments on the behavior of these criteria as well as recognition performance are depicted in Section 4. Finally, conclusions are given in Section 5.

2 The Object Recognition System

The object recognition system consists of a neural network which has been trained with feature vectors reflecting 2D (contour) and 2.5D (local surface curvature) shape of the objects. Roughly, the main concepts related to this recognition system are explained below.

The *FuzzyARTMAP neural network* is based on the Adaptive Resonance Theory (ART) which was developed by Stephen Grossberg and Gail Carpenter at Boston University. In Fuzzy ARTMAP there are two modules ART_a and ART_b and an inter-ART module "map field" that controls the learning of an associative map from ART_a recognition categories to ART_b recognition categories [1]. The map field module also controls the match tracking of ART_a vigilance parameter. A mismatch between Map field and ART_a category activated by input $\mathbf{a}$ and ART_b category activated by input $\mathbf{b}$ increases ART_a vigilance by the minimum amount needed for the system to search for, and if necessary, learn a new ART_a

category whose prediction matches the ART_b category. The search initiated by the inter-ART reset can shift attention to a novel cluster of features that can be incorporated through learning into a new ART_a recognition category, which can then be linked to a new ART prediction via associative learning at the Map field. The algorithm uses a preprocessing step, called complement coding which is designed to avoid category proliferation. Similar to ART-1, a vigilance parameter measures the difference allowed between the input data and the stored pattern. Therefore this parameter is determinant to affect the selectivity or granularity of the network prediction. For learning, the FuzzyARTMAP has 4 important factors: Vigilance in the input module (ρ_a), vigilance in the output module (ρ_b), vigilance in the Map field (ρ_{ab}) and learning rate (β). These were the considered factors in this research with values of $\rho_a = 0.93$, $\rho_b = 1$, $\rho_{ab} = 0.95$ and $\beta = 1$.

The method obtains the object's contour using metric properties such as the perimeter, area and centroid information in order to form the so-called *Boundary Object Function (BOF)*[2]. The BOF is a 2D descriptor vector that contains the euclidean distance between the object's contour and its centroid. The vector is formed by 180 elements obtained from the measurement of the distance between a contour point and the centroid every two degrees. The BOF and its starting point is easily determined for geometrical figures such as circles, but in complex shapes the procedure is more involved. The starting point is important since this is also a reference for the Neural Network Pattern Recognition system.

The Shape-From-Shading method (SFS) [3] consists primarily of obtaining the orientation of the surface due to local variations in brightness that is reflected by the object, in other words, the intensities of the greyscale image is taken as a topographic surface. The surface normal representation has also been called the 2.5D sketch by Marr [6]. SFS is known as an ill-posed problem, causing ambiguity between what has a concave and convex surface, which is due, among other things, to poorly illuminated surface patches [4]. In this paper, we make use of the particular SFS method proposed in [5], the surface normal is rotated in accordance with a local surface curvature measure known as the *Shape Index* (SI) [7]. This measure provides an idea of how patches over the surface correspond to degrees of concavity and convexity. The SFS method in [5] has proved to be useful for object recognition when using SI histograms. In this work, SI-histograms are also used to build feature vectors encoding 2.5D information for the neural network.

3 Improved Image Feature Extraction

In this section we describe the methods used to extract relevant information for 2D and 2.5D shape classification. Both methods are based on intensity histogram analysis. The distribution of the intensities in the images is then helpful to determine the goodness of each image for the purposes of 2D and 2.5D feature extraction. We commence with the description of the *two-class separation criterion* for BOF. The idea here is to provide the BOF with a feature extraction method able to deliver correct information, i.e., a good segmentation between object and background that allows the classifier to sharply determine the different 2D shapes (contour) of the objects. To this end, we borrow ideas from Otsu's method [13] to automatically perform histogram shape-based image thresholding, reducing a graylevel image to a binary image. The algorithm assumes that the image to be thresholded contains two classes of pixels (e.g. foreground and background) then calculates the optimum threshold separating those two classes so that their combined intra-class variance is minimal. Let us define the global and intra-class variances as

$$\sigma_G^2 = \sum_{i=0}^{L-1}(i - m_G)^2 P_i \quad \text{and} \quad \sigma_B^2 = P_1(m_1 - m_G)^2 + P_2(m_2 - m_G)^2, \tag{1}$$

respectively, where L is the number of gray levels (tipically 256), m_G is the average gray level in the image and P_i is the probability of the i_{th} gray level in the image. The probabilities P_1 and P_2 of the two potential classes C_1 and C_2 are defined respectively as $P_1(k) = \sum_{i=0}^{k} P_i$ and $P_2(k) = 1 - P_1(k)$ with $0 < k < L - 1$. The average probabilities m_1 and m_2 of C_1 and C_2 are defined respectively as $m_1(k) = \frac{1}{P_1(k)} \sum_{i=0}^{k} iP_i$ and $m_2(k) = \frac{1}{P_2(k)} \sum_{i=k+1}^{L-1} iP_i$.

Note that the optimal classes C_1 and C_2 are separated by the k_{th} gray level. Therefore, the optimal threshold in Otsu's method is the gray level value that makes σ_B^2 maximal. The relationship between global and intra-class variances is given by $\eta(k) = \sigma_B^2(k)/\sigma_G^2$. This means that the intra-class variability must be evaluated for every gray level as $\sigma_B^2(k) = (m_G P_1(k) - m(k))^2 / P_1(k)(1 - P_1(k))$, with $m(k)$ being the cumulative average probability up to the k_{th} gray level. The optimal threshold is then defined as k^* satisfying

$$\sigma_B^2(k^*) = max(\sigma_B^2(k)). \tag{2}$$

For the total number of changes in light source intensity, the criterion

$$\eta_b(n) = \frac{\sigma_B^2(k^*, n)}{\sigma_G^2(n)}, 1 < n < N, \tag{3}$$

is recorded for every n_{th} intensity variation, where N is the number of variations. The n_{th}^* image maximizing the criterion is then selected using the equation $\eta_b(n^*) = max(\eta_b(n))$. Finally, the optimal threshold k^* is used to segment the selected image. Once the contour of the object is at hand, the BOF neural network feature vector can be built.

In the following, the *Gaussian distribution criterion* for SFS is described. As far as 2.5D shape is concerned, shape index histograms are constructed for each image in order to represent local surface shape for the different objects. The aim of this representation is to allow the recognition system to determine whether the surface of the observed object corresponds to a pyramidal or curved one (see Fig. 1 (left)). As this data is calculated from surface normal estimations obtained through a geometrical SFS method [5], it is important to guarantee that the light source illuminating the objects minimizes unwanted artifacts in the image. Typically, these artifacts appear due to specular reflections and as a consequence of overly illuminated patches on the surface of the object. Other undesired effects may be caused by the opposite situation, i.e., by poorly illuminating surface patches. In any case, a light source intensity too high or too low leads to pixel values located near the limits of the gray level range. As a result, surface orientation may be wrongly estimated due to the ambiguity of such gray level values. An appropriate distribution of gray levels among the objects in the observed image is therefore sought in order to provide a solution to reducing the number of ambiguously valued pixels. To overcome this problem, we propose selecting the image whose gray level histogram best fits a Gaussian distribution with mean μ and standard deviation σ. The criterion can be defined as

$$\eta_s = \sum_{x=1}^{L} \left|\left| P(x) - \left(\frac{1}{\sigma\sqrt{2\pi^2}} e^{\frac{-(x-\mu)^2}{2\sigma^2}} \right) \right|\right|^2, \tag{4}$$

where $P(x)$ is the probability of the gray level x to occur in the image. The criterion $\eta_s(n), 1 < n < N$, where N is the total number of changes in light source intensity is recorded for each intensity variation. Finally, the n^*_{th} image best satisfying the criterion is selected as the optimal image for building the SFS neural network feature vector, as shown by the equation $\eta_s(n^*) = min(\eta_s(n))$.

4 Experiments

Eight different hand-made pieces were used for experimental tests. The different pieces attempt to emulate common objects in a manufacturing environment. The pieces can be observed in Figure 1 (left). As shown in the figure, the combination of features in the set consists of four 2D shapes (square, triangle, cross, star) and two 2.5D shapes (curved, pyramidal). The acquisition platform is shown in Figure 1 (right). The platform is conformed by a halogen lamp connected to a dimmer in order to regulate light source intensity. An extra halogen lamp was used for experiments considering a fixed light source intensity. A camera was synchronized to capture an image along each intensity variation. The gain of the camera was set to zero and the auto-adjustment feature was off in order to isolate the effect of illumination only to the influence of the lamps. The tests

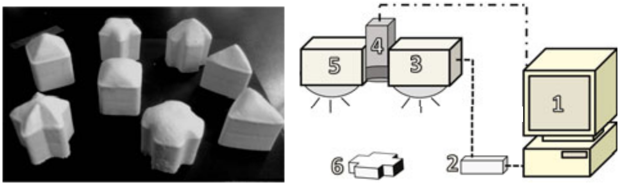


Fig. 1. The image acquisition platform. The eight different pieces used for the experiments (left). The acquisition platform (right) shows a diagram of the different devices (labeled as numbers) used for capturing images: one fixed(5), one variable(3) light source, one camera(4), a dimmer(2), a computer(1) and one piece(6).

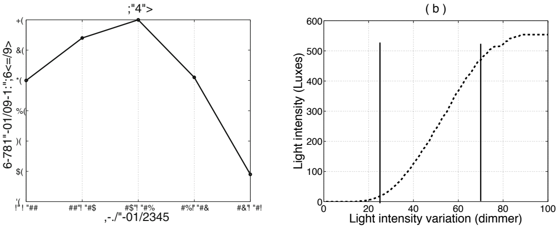


Fig. 2. Behavior of ambient light and artificial light source intensities. In (a), the intensity of ambient light is shown, in luxes, as a function of daytime interval. In (b), the intensity of the artificial light source, in luxes, is shown as a function of the resistance of the dimmer. The linear response of the light is bounded between the two vertical lines in the diagram.

were carried out during five intervals of time in one day. The duration of each interval was of two hours, starting at 9 and finishing at 19 hrs. A light meter was used to measure the luminance of both ambient light (i.e., the amount of light received around the scene area, coming from bulbs in the working area as well as from natural external light) and the artificial light source (the halogen lamps).

This information is shown in Figure 2. The left diagram of the figure depicts a plot of the ambient light intensity behavior during the five time intervals. Note that the peak intensity was reached at the sunniest time (13-15 hrs. interval), while a considerably lower intensity was reached near the evening (17-19 hrs. interval). The right diagram of the figure depicts a plot of the lamp intensity against resistance of the dimmer, for the 9-11 hrs. interval. In the plot, the ambient light is subtracted. Note that only the linear response, bounded between two vertical lines, was taken into account for the experiments. Therefore, only ten equally spaced dimmer values (from 25 to 70) were synchronized with the camera.

4.1 Optimality Analysis for the Criteria η_b and η_s

The image acquisition procedure involved in the experiments is described in this section. One single piece was placed on three main locations: the center, the top-left and bottom-right corners of the viewing plane of the camera, over a dark card. At the center, the piece was arbitrarily rotated on its own axis twice for a total number of four events. For each event, an image was taken 10 times from the synchronized-with-dimmer camera. From the ten image set, the corresponding criteria η_b and η_s were recorded. In the next figures, results on a total number of 1600 images = 8 pieces × 4 events × 10 variations in intensity light × 5 intervals of time are shown. We start the analysis in Figure 3, where the evolution of the 2D shape criterion η_b is shown as a function of light intensity variation. From left to right, results over the 2D shapes of square, triangle, cross and star are presented. For each shape, the average of all the corresponding light source intensity variations during the five intervals of time are depicted.

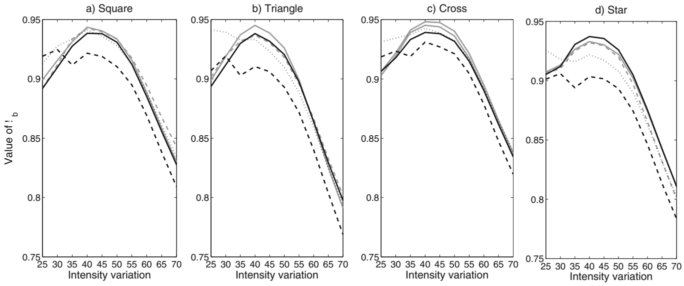


Fig. 3. Evolution of η_b as a function of dimmer values and for the different time intervals. Results on the 2D shapes are labeled accordingly on each diagram. The different intervals of time are represented using the following lines: 9 - 11 hrs., solid gray; 11 - 13 hrs., dashed gray; 13 - 15 hrs., solid black; 15 - 17 hrs., dotted black and 17 - 19 hrs., dashed black.

The different intervals are represented using the following lines: 9 - 11 hrs., solid gray; 11 - 13 hrs., dashed gray; 13 - 15 hrs., solid black; 15 - 17 hrs., dotted black and 17 - 19 hrs., dashed black. From the figure, it is noticeable that the highest value of η_b, and therefore the optimal criterion for binary segmentation happens near the dimmer resistance area of 40 - 50, with a relatively steady behavior for the first three intervals of time. A less predictable behavior is nonetheless shown during the last two intervals of time (dashed and dotted black lines), which may be explained by the rapid descent of ambient light during those hours of the day. Note how, for all the cases, the value of η_b tends to descend once a maximum value has been reached.

The evolution of the 2.5D shape criterion η_s is shown in Figure 4. Results during different intervals of time are presented in a similar way as in Figure 3. Here, the diagrams are organized in accordance with the curved and pyramidal surface features of the pieces. Although the overall behavior appears to increase once a minimum has been reached, this minimum seems to require a bigger amount of light intensity for the last two intervals of time. Such effect may be again explained as a consequence of the lack of ambient light towards the evening. Note how, in the figure, the different shapes of the pieces seem to be responsible of most of the variability in both the minimum value of η_s and the required light source intensity. This fact suggests that the adjustment of the lightning requirements of the scene depends on the particular 2D and 2.5D features of the piece rather than using a fixed illumination for every object and for all times of the day.

4.2 Recognition

In order to show the importance of a dynamic selection of the criteria η_b and η_s, we compare the performance of a fixed light source intensity with an empirical fixed threshold for segmentation against a variable light source intensity with optimal citeria η_b and η_s as well as optimal threshold k^* (see Eq. 2). The fixed light

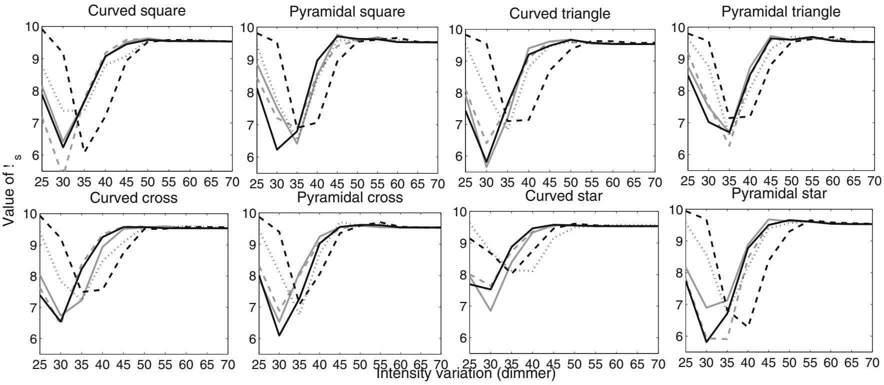


Fig. 4. Evolution of η_s as a function of dimmer values and for the different time intervals. Results on the 2.5D shapes are labeled accordingly on each diagram. The different intervals of time are represented as in Figure 3.

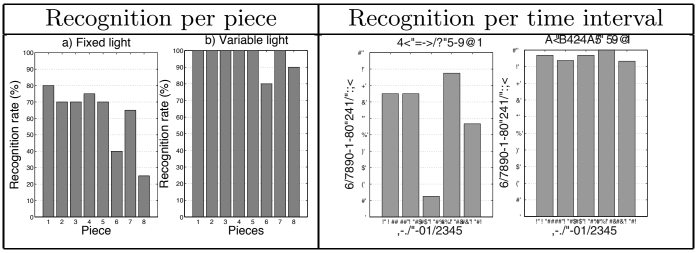


Fig. 5. Recognition rates under fixed and variable light source intensities. The two panels contain bar diagrams with results concerning fixed and variable light source intensity. For the fixed light, a fixed threshold for BOF segmentation was also used. Recognition rates per piece (left) and per time interval (right) are shown.

was registered to emit around 100 luxes (without ambient light) at the start of the experiments, which can be comparable to using the dimmer value of 40 on the variable light source. The recognition experiment is roughly explained as follows. For each event, ten intensity variations are synchronized with a camera shot. The image with optimal criterion η_b and threshold k^* is selected, its contour extracted, and its BOF feature vector built. Once the optimal segmentation is at hand, the background is removed from all of the ten images and the optimal criterion η_s sought in order to select the best image for building the SFS feature vector. Finally, a single image is captured using the fixed light source and a fixed threshold. The image is segmented and its BOF feature vector is generated. Background removal is then performed and the SFS feature vector built. A piece is successfully recognized when both 2D and 2.5D features are correctly determined. For example, the system should be able to determine whether the piece is a curved square or a pyramidal cross, among other possibilities in the set.

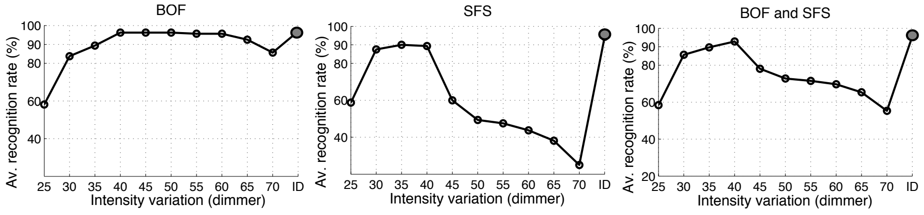


Fig. 6. Average recognition rates for all time intervals per light source intensity variation. From left to right, results concerning BOF, SFS and BOF+SFS are shown as a function of light source intensity variation, respectively. The recognition rate from optimal values η_b and η_s is shown as a gray circle at the end of the line plot.

Recognition rates per piece and per time interval are shown in Figure 5 (left) and Figure 5 (right), respectively. In the figure, the different pieces are labeled as follows: 1-2, square; 3-4, triangle; 5-6, cross; and 7-8, star. Even and odd numbers correspond to curved and pyramidal shapes, respectively. The figure reveals that both fixed and variable approaches struggle to recognize more complex shapes such as crosses and stars, with a particular emphasis in pyramidal shapes. Nonetheless, the advantage of varying light source intensity over a fixed light is clearly demonstrated. As far as the time interval analysis is concerned, it is evident that the fixed light intensity approach encounters more difficulty during the 13 - 15 hrs. period, which may be caused by the peak in ambient light during that interval of time. On the contrary, recognition rates for the variable light intensity approach remain relatively steady along the five intervals of time. This fact justifies the need of an automatic adjustment of both light source intensity and optimal threshold to the particular features of the observed piece and the particular illumination conditions of the different hours of the day.

To conclude the recognition analysis, Figure 6 presents a panorama of the average recognition rate as a function of light source intensity variation (value of dimmer). The figure is divided into three plots, where recognition on BOF (left), SFS (center) and BOF + SFS (right) are shown separately. For this figure, results using the fixed light source and fixed threshold are not included. Instead, we focused on showing the recognition rate per each light source intensity variation and using an optimal threshold k^* for segmentation, i.e., as if none of the criteria η_b and η_s had been calculated for image selection. However, at the end of each line plot, results on recognition with optimal criteria η_b and η_s are depicted with a gray circle. Note how for the BOF-based recognition, a stable high recognition rate is achieved from 40 to 60 dimmer values, which suggests that a good separation of object from background may be achieved from images obtained within a range of light source intensities. However, for results related to SFS-based recognition, a smaller region (from 30 to 40 dimmer values) shows a lesser recognition rate, which suggests that the extraction of accurate local surface curvature from images may be too sensitive to light source intensity fluctuations. Predictably, for the BOF + SFS case, only the dimmer value of 40 provides a recognition rate slightly higher than 90%, reducing the trustable region to a single dimmer value. This observation suggests that fixing the light source intensity at a dimmer value of 40 may be sufficient for obtaining good recognition results. Nonetheless, although this fixed value of 40 shows a relatively steady recognition rate, ambient light is a factor which cannot be controlled. In other words, what may be useful in a sunny day, may not be so on a darker day. Interestingly, the advantage of using the optimal criteria η_b and η_s is revealed as the highest and most stable of all recognition rates. This justifies again the necessity of an automatic adjustment of the light source intensity if robustness against uncontrolled changes in ambient light is to be provided in recognition systems based on image feature extraction.

5 Conclusions

We have addressed the problem of compensating illumination requirements for the purposes of 2D and 2.5D shape feature extraction. To this end, we have proposed to adapt two histogram-based criteria to the particular needs of the observed scenario during different times of the day. The proposed criteria have shown to improve stability and robustness to ambient light changes. Particularly, this enhanced feature extraction from the automatic adjustment of light source intensity has extended the capabilities of a neural network based recognition system. As future work, we plan to investigate the inclusion of additional light sources as well as the adaptation of other criteria for different feature extraction tasks, i.e., the optimal illumination conditions for stereo matching and other 2D feature extraction such as corners and contours for a more complex class of scenes.

References

1. Carpenter, G.A., Grossberg, S., Markuzon, N., Reyno1ds, J.H., Rosen, D.B.: FuzzyARTMAP: A neural network architecture for incremental learning of analog multidimensional maps. IEEE Trans. on Neural Networks 3(5), 698–713 (1992)
2. Peña-Cabrera, M., Lopez-Juarez, I., Rios-Cabrera, R., Corona-Castuera, J.: Machine Vision Approach for Robotic Assembly. Assembly Automation 25(3), 204–216 (2005)
3. Horn, B.K.P.: Shape from Shading: A Method for Obtaining the Shape of a Smooth Opaque Object from One View. PhD thesis, MIT (1970)
4. Brooks, M.: Two results concerning ambiguity in shape from shading. In: AAAI-83, pp. 36–39 (1983)
5. Worthington, P.L., Hancock, E.R.: Object Recognition Using Shape-from-Shading. IEEE Trans. on Pattern Analysis and Machine Intelligence 23(5), 535–542 (2001)
6. Marr, D., Nishihara, H.K.: Representation and Recognition of the Spatial Organization of Three Dimensional Shapes. Proc. Royal Society of London, B. 200, 269–294 (1978)
7. Koenderink, J., Van Doorn, A.: Surface shape and curvature scale. Image and Vision Computing 10, 557–565 (1992)
8. Collewet, C.: Modeling complex luminance variations for target tracking. In: Proc. IEEE Int. Conf. on Computer Vision and Pattern Recognition, pp. 1–7 (2008)

9. Moses, Y., Adini, Y., Ullman, S.: Face Recognition: the Problem of Compensating for Changes in Illumination Direction. In: Proc. European Conference on Computer Vision, pp. 286–296 (1994)
10. Liao, M., Wang, L., Yang, R.: Gong. M, Light Fall-off Stereo. In: IEEE Conference on Computer Vision and Pattern Recognition (2007)
11. Wang, L., Yang, R., Davis, J.E.: BRDF Invariant Stereo Using Light Transport Con- stancy. IEEE Trans. on Pattern Analysis and Machine Intelligence 29(9), 1616–1626 (2007)
12. Woodham, R.J.: Photometric method for determining surface orientation from multiple images. Optical Enginnering 19(1), 139–144 (1980)
13. Otsu, N.: A threshold selection method from gray-level histograms. IEEE Trans. Sys., Man., Cyber. 9, 62–66 (1979)

Fringe-Pattern Demodulation Using a Parametric Method Based on Differential Evolution

J.F. Jimenez[1], F.J. Cuevas[2], J.H. Sossa[1], and L.E. Gomez[1]

[1] Centro de Investigación en Computación-IPN, Unidad Profesional Adolfo-López Mateos, Av. Juan de Dios Bátiz s/n and M. Othón de Mendizábal, Zacatenco, México, DF. 07738, Mexico

[2] Centro de Investigaciones en Óptica A.C. Loma del Bosque #115, Col. Lomas del Campestre C.P. 37150, León Gto. México

jfvielma@cio.mx, hsossa@cic.ipn.mx, fjcuevas@cio.mx, sgomezb08@sagitario.cic.ipn.mx

Abstract. A parametric method to carry out fringe pattern demodulation by means of Differential Evolution is presented. The phase is approximated by the parametric estimation of an nth-grade polynomial so that no further unwrapping is required. On the other hand, a different parametric function can be chosen according to the prior knowledge of the phase behavior. A differential evolution is codified with the parameters of the function that estimates the phase. The differential evolution evolves until a fitness average threshold is obtained. The method can demodulate noisy fringe patterns and even a one-image closed-fringe pattern successfully.

Keywords: Phase retrieval; Fringe analysis; Optical metrology; Differential Evolution.

1 Introduction

In optical metrology, a fringe pattern (interferogram) can be represented using the following mathematical expression:

$$I(x,y) = a(x,y) + b(x,y) \times \cos(\omega_x x + \omega_y y + \phi(x,y) + n(x,y)) \qquad (1)$$

where *x, y* are integer values representing indexes of the pixel location in the fringe image, *a(x,y)* is the background illumination, *b(x,y)* is the amplitude modulation and is $\phi(x,y)$ the phase term related to the physical quantity being measured. ω_x and ω_y are the angular carrier frequency in directions *x* and *y*. The term $n(x,y)$ is an additive phase noise. The purpose of any interferometric technique is to determine the phase term, which is related to the physical quantity, being measured. One way to calculate the phase term $\phi(x,y)$ is by using the phase-shifting technique (PST) [1–5], which needs at least three phase-shifted interferograms. The phase shift among interferograms must be known and experimentally controlled. This technique can be used when mechanical conditions are met throughout the interferometric experiment.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 61–70, 2010.

On the other hand, when the stability conditions mentioned are not covered, there are many techniques to estimate the phase term from a single fringe pattern, such as: the Fourier method [6,7], the Synchronous method [8] and the phase locked loop method (PLL) [9], among others. However, these techniques work well only if the analyzed interferogram has a carrier frequency, a narrow bandwidth and the signal has low noise. Moreover, these methods fail for phase calculation of a closed-fringe pattern. Additionally, the Fourier and Synchronous methods estimate the phase wrapped because of the arctangent function used in the phase calculation, so an additional unwrapping process is required. The unwrapping process is difficult when the fringe pattern includes high amplitude noise, which causes differences greater than 2π radians between adjacent pixels [10–12].

Recently, regularization [13–15] and neural networks techniques [16,17] have been used to work with fringe patterns, which contain a narrow bandwidth and noise.

In this work, we propose a technique to determine the phase $\phi(x, y)$, from a fringe pattern with a narrow bandwidth and/or noise, by parametric estimation of a global non-linear function instead of local planes in each site *(x,y)* as it was proposed in [13,19]. Differential Evolution (DE) algorithm is a new heuristic approach mainly having three advantages; Finding the true global minimum regardless of the initial parameter values, fast convergence, and using few control parameters. DE algorithm is a population based algorithm like genetic algorithms using similar operators; crossover, mutation and selection. When a noisy closed fringe pattern is demodulated, neither a low-pass filter nor a thresholoding operator is required. On the other hand, regularization techniques need both of them.

2 DE Applied to Phase Recovery

The standard Differential Evolution (DE) algorithm, belonging to the family of Evolutionary Algorithms, was described by Storn and Price [20],[21]. It is based on evolution of a population of vectors, which encode potential solutions to the problem and traverse the fitness landscape by means of genetic operators that are supposed to bias their evolution towards better solutions. DE is a relatively new optimisation technique compared with other more established Evolutionary Algorithms, such as Genetic Algorithms, Evolutionary Strategy, and Genetic Programming [22].

DE is an optimization algorithm that creates new candidate solutions by combining the parent vector and several other vectors of the same population. A candidate replaces the parent only if it has better fitness [22],[23]. DE uses genetic operators, referred to as mutation, crossover and selection. The role of the genetic operators is to ensure that there is sufficient pressure to obtain even better solutions from good ones (exploitation) and to cover sufficiently the solution space to maximize the probability of discovering the global optimum (exploration).

During the initialization of the algorithm, a population of NP vectors, where NP is the number of vectors, each of dimension D (Which is the number of decision variables in the optimization problem), is randomly generated over the feasible search space.

The fringe demodulation problem is difficult to solve when the level of noise affecting the fringe pattern is elevated, since many solutions are possible even for a single noiseless fringe pattern. Besides, the complexity of the problem is increased when a carrier frequency does not exist (closed fringes are presented).

Given that for a closed fringe interferogram there are multiple phase functions for the same pattern, the problem is stated as an ill-posed problem in the Hadamard sense, since a unique solution cannot be obtained [23]. It is clear that image of a fringe pattern $I(x, y)$ will not change if $\phi(x, y)$ in Eq. (1) is replaced with another phase function $\tilde{\phi}(x, y)$ given by

$$\tilde{\phi}(x, y) = \begin{cases} -\phi(x, y) + 2\pi & (x, y) \in R, \\ \phi(x, y) & (x, y) \notin R \end{cases} \tag{2}$$

where R is an arbitrary region and k is an integer. In this work, a DE is presented to carry out the optimization process, where a parametric estimation of a non-linear function is proposed to fit the phase of a fringe pattern. Then, DE technique fits a global non-linear function instead of a local plane to each pixel just like it is made in regularization techniques [13,19]. The fitting function is chosen depending on the prior knowledge of the demodulation problem as object shape, carrier frequency, pupil size, etc. When no prior information about the shape of $\phi(x, y)$ is known, a polynomial fitting is recommended. In this paper, authors have used a polynomial fitting to show how the method works.

The purpose in any application of DE is to evolve a population of size NP (which codifies NP possible solutions to the problem) using mutation, crossover and selection of each vector, with the goal of optimizing a fitness function adequate to the problem to solve.

In this work, the fitness function U, which is used to evaluate the pth vector a^p in the population, is given by [18]:

$$\begin{aligned} U\left(a^p\right) = \alpha - \sum_{y=1}^{R-1} \sum_{x=1}^{C-1} \Big\{ & \left(I_N(x, y) - \cos\left(\omega_x x + \omega_y y + f\left(a^p, x, y\right)\right)\right)^2 \\ & + \lambda\Big[\left(f\left(a^p, x, y\right) - f\left(a^p, x-1, y\right)\right)^2 \\ & + \left(f\left(a^p, x, y\right) - f\left(a^p, x, y-1\right)\right)^2\Big]\Big\} m(x, y), \end{aligned} \tag{3}$$

where x, y are integer values representing indexes of the pixel location in the fringe image. Superindex p is an integer index value between 1 and NP, which indicates the number of vectors in the population. $I_N(x, y)$ is the normalized version of the detected irradiance at point (x, y). The data were normalized in the range $[-1,1]$. ω_x and ω_y are the angular carrier frequencies in directions x and y. The Function $f(\cdot)$ is the selected fitting function to carry out the phase approximation. $R \times C$ is the image resolution where fringe intensity values are known and λ is a smoothness weight factor (it should be clear for the reader that a higher value of parameter λ implies a smoother function to be fitted). The binary mask $m(x, y)$ is a field which defines the valid area in the fringe pattern. The parameter a can be set to the maximum value of the second term (in negative sum term) at Eq. (3) in the first vector population, which is given by

$$\alpha = \max_{p} \left\{ \sum_{y=1}^{R-1} \sum_{x=1}^{C-1} \{ \left(I_N(x,y) - \cos\left(\omega_x x + \omega_y y + f\left(a^p, x, y\right)\right)\right)^2 \right. \\ + \lambda \left[\left(f\left(a^p,x,y\right) - f\left(a^p, x-1, y\right)\right)^2 \right. \\ \left. + \left(f\left(a^p,x,y\right) - f\left(a^p,x,y-1\right)\right)^2 \right] \} m(x,y), \tag{4}$$

parameter α is used to convert the proposal from minimal to maximal optimization since a fitness function in a DE is considered to be a nonnegative figure of merit and profit [20].

The first term (in negative sum term) at Eq. (3) attempts to keep the local fringe model close to the observed irradiances in least-squares sense. The second term (in negative sum term) at Eq. (3) is a local discrete difference, which enforces the assumption of smoothness and continuity of the detected phase.

At the beginning of a DE, a set of random solutions are codified in a vector population of size NP. Each vector a is formed by the parameter function vector (possible solution) and chained string such as:

$$a = \left[a_0 \middle| a_1 \middle| a_2 \middle| \ldots \middle| a_n \right] \tag{5}$$

Each dimension a_i is a random real number in a defined search range $\left(\min\left(a_i\right), \max\left(a_i\right)\right)$ (the user defined maximum and minimum of a_i). These values can be initialized using prior knowledge (e.g. in the polynomial case, components x and y are related to the interferogram tilt so if a closed fringe is presented, then these values are near 0). Every dimension is generated as:

$$a_i = random\left(\min\left(a_i\right), \max\left(a_i\right)\right) \tag{6}$$

Therefore, the population of DE consists of NP D-dimensional parameter vectors $X_{i,G}$, where $i = 1, 2, \ldots, NP$, for each generation G.

2.1 Mutation

In the mutation step, a difference between two randomly selected vectors from the population is calculated. This difference is multiplied by a fixed weighting factor, F, and it is added to a third randomly selected vector from the population, generating the mutant vector, $V[1-3]$.

For each target vector $x_{i,G}$, a mutant vector is produced by;

$$v_{i,G+1} = x_{i,G} + K \bullet \left(x_{r1,G} - x_{i,G} \right) + F \bullet \left(x_{r2,G} - x_{r3,G} \right) \tag{7}$$

where $i, r_1, r_2, r_3 \in \{1, 2, ..., NP\}$ are randomly chosen and must be different from each other. In Equation (7), F is the scaling factor which has an effect on the difference vector $(x_{r^2,G} - x_{r^3,G})$, K is the combination factor.

2.2 Crossover

After mutation, the crossover is performed between the vector (X) and the mutant vector (V) (Figure 1), using the scheme in (8) to yield the trial vector (U). The crossover probability is determined by the crossover constant (CR), and its purpose is to bring in diversity into the original population [24].

The parent vector is mixed with the mutated vector to produce a trial vector $u_{ji,G+1}$

$$u_{ji,G+1} = \begin{cases} v_{ji,G+1} & if \left(rnd_j \leq CR\right) or \; j = rn_i, \\ \\ q_{ji,G} & if \left(rnd_j > CR\right) or \; j \neq rn_i, \end{cases} \tag{8}$$

where $j = 1, 2, ..., D$; $rnd_j \in [0,1]$ is the random number; CR is crossover constant $\in [0,1]$ and $rn_i \in (1, 2, ..., D)$ is the randomly chosen index, which ensures that $u_{i,G+1}$ gets at least one parameter from $v_{i,G+1}$ [19].

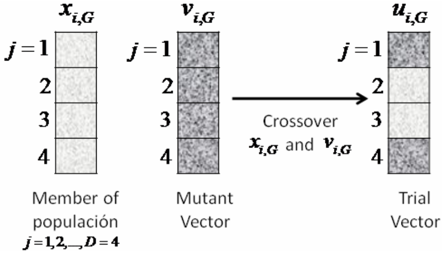


Fig. 1. Illustration of the crossover process for *D*=4

There are different variants that can be used in mutation and crossover, and they are referred to as DE/x/y/z, where x specifies the vector to be mutated which currently can be "rand" (a randomly chosen population vector) or "best" (vector of the lowest cost from the current population); y is the number of difference vectors used and z denotes the crossover scheme [21].

2.3 Selection

In the last step, called selection, the new vectors (U) replace their predecessors if they are closer to the target vector. All solutions in the population have the same

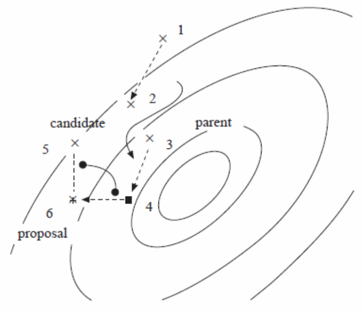


Fig. 2. Obtaining a new proposal in DE

chance of being selected as parents without dependence of their fitness value. The child produced after the mutation and crossover operations is evaluated. Then, the performance of the child vector and its parent is compared and the better one is selected. If the parent is still better, it is retained in the population.

Figure 2 shows DE's process in detail: the difference between two population members (1,2) is added to a third population member (3). The result (4) is subject to the crossover with the candidate for replacement (5) to obtain a proposal (6). The proposal is evaluated and replaces the candidate if it is found to be better.

DE has shown to be effective on a large range of classical optimization problems, and it showed to be more efficient than techniques such as Simulated Annealing and Genetic Algorithms [23],[24]. However, its capability of finding the global optimum is very sensitive to the choice of the control variable F and CR [25]. Consistently with related studies [23],[24],[25], the paper highlights an undesirable behaviour of the algorithm, i.e., the DE does not find the global optimum (value to reach - VTR) when 100% of the population is trapped in a basin of attraction of a local optimum.

2.4 DE Convergence

The DE convergence mainly depends on the population size. It should be clear that if we increase the population size, more vectors will search the global optimum and a best solution will be found in a minor number of iterations, although the processing time can be increased [25].

To stop the DE process, different convergence measures can be employed. In this paper, we have used a relative comparison between the fitness function value of the best vectors in the population and value a , which is the maximum possible value to get in Eq. (3). Then, we can establish a relative evaluation of uncertainty to stop the DE as:

$$\left| \frac{\alpha - U\left(a^*\right)}{\alpha} \right| \leq \varepsilon, \tag{9}$$

where $U\left(a^*\right)$is the fitness function value of the best vectors in the population in the current iteration, and ε is the relative error tolerance. Additionally, we can stop the process in a specified number of iterations, if Eq. (9) is not satisfied.

3 Experiment

The parametric method using a DE was applied to calculate phase from shadow moiré closed fringe pattern. We used a population size equal to 100, F is calculated by values of "F_lower" and "F_higher", in the ranges $[0.1, 0.9]$. In each vector, the coded coefficients of a fourth degree polynomial were included. The following polynomial was coded in each vector:

$$\begin{aligned} p_4(x,y) = a_0 + a_1 x + a_2 y + a_3 x^2 + a_4 xy + a_5 y^2 + a_6 x^3 + a_7 x^2 y + a_9 xy^2 \\ + a_9 y^3 + a_{10} x^4 + a_{11} x^3 y + a_{12} x^2 y^2 + a_{13} xy^3 + a_{14} y^4 \end{aligned} \tag{10}$$

so that 15 coefficients were configured in each vector inside population to be evolved.

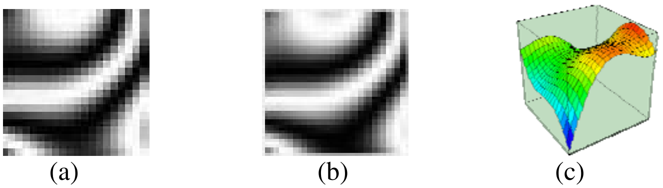
A low contrasted noisy closed fringe pattern was generated in the computer using the following expression:

$$I(x,y) = 127 + 63\cos(P_4(x,y) + \eta(x,y)), \tag{11}$$

where

$$\begin{aligned} p_4(x,y) = -0.7316x - 0.2801y + 0.0065x^2 + 0.00036xy - 0.0372y^2 \\ + 0.00212x^3 + 0.000272x^2 y + 0.001xy^2 - 0.002y^3 \\ + 0.000012x^4 + 0.00015x^3 y + 0.00023x^2 y^2 + 0.00011xy^3 \\ + 0.000086y^4 \end{aligned} \tag{12}$$

and $\eta(x,y)$ is the uniform additive noise in the range $[-2radians, 2radians]$. Additionally, the fringe pattern was generated with a low resolution of 60×60. In this case, we use a parameter search range of $[-1,1]$. The population of vectors was evolved until the number of iterations and relative error tolerance ε was 0.05 in Eq. (9). This condition was achieved in 77s on a AMD Turion X2-2.4 GHz computer. The fringe pattern and the contour phase field of the computer generated interferogram are shown in Fig. 3.



(a) (b) (c)

Fig. 3. (a) Original fringe pattern, (b) phase field obtained by using DE technique and (c) phase obtained in 3D

The DE technique was used to recover the phase from the fringe pattern. The fringe pattern and the phase estimated by DE is shown in Fig. 3. Tests are shown on Table 1, the best vectors for the testers are shown on Table 2, and worst vectors for the testers are shown on Table 3.

Table 1. Table of parameters of "F_lower" and "F_higher"

	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
0.1	2.30E-02	4.84E-01	7.79E-01	6.29E-02	1.64E-01	2.24E-02	2.63E-01	3.77E-01	7.58E-01
0.2	3.60E-02	4.80E-02	7.19E-01	9.19E-01	1.21E+00	1.52E-02	**1.28E-03**	**1.40E-04**	**9.63E-03**
0.3	1.99E-02	**4.37E-02**	**1.08E-03**	1.32E+00	1.83E-03	**1.12E-04**	1.35E+00	8.28E-03	2.24E+00
0.4	7.00E-01	1.04E+00	1.51E+00	3.62E-01	**1.58E-04**	9.44E-04	1.61E+00	1.92E-03	2.90E+00
0.5	**1.03E-03**	2.81E-01	1.47E+00	1.67E+00	3.54E-04	1.76E+00	2.12E+00	1.92E+00	1.94E+00
0.6	7.88E-01	2.02E-01	1.44E+00	1.35E+00	1.46E+00	2.23E+00	1.80E+00	2.72E+00	3.14E+00
0.7	3.15E-01	1.19E+00	1.95E+00	1.17E+00	1.88E+00	2.31E+00	2.11E+00	3.29E+00	2.87E+00
0.8	9.20E-01	1.74E+00	1.31E+00	1.91E+00	2.27E+00	2.02E+00	2.11E+00	2.77E+00	3.79E+00
0.9	1.03E-03	1.89E+00	1.40E+00	**3.09E-03**	2.71E+00	3.11E+00	2.48E+00	2.08E+00	3.07E+00

Table 2. Shows of the best vectors

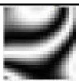
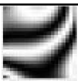
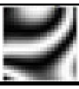
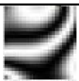
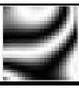
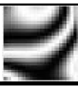
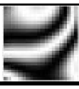
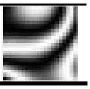
F_lower	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
F_higher	0.5	0.3	0.3	0.9	0.4	0.3	0.2	0.2	0.2
Error	1.03E-03	4.37E-02	1.08E-03	3.09E-03	1.58E-04	**1.12E-04**	1.28E-03	1.40E-04	9.63E-03
									

Table 3. Shows of the worst vectors

F_lower F_higher Error	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9
	0.8	0.9	0.7	0.8	0.9	0.9	0.9	0.7	0.8
	9.20E-01	1.89E+00	1.95E+00	1.91E+00	2.71E+00	3.11E+00	2.48E+00	3.29E+00	**3.79E+00**

The phases: original, best vector, as worst vector, is shows in the Fig. 4.

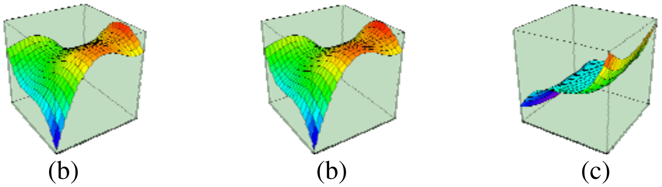


Fig. 4. Phases: (a) Original, (b) best vector of DE technique and (c) worst vector of DE technique.

4 Conclusions

A DE was applied to recover the modulating phase from closed and noisy fringe patterns. A fitness function, which considers the prior knowledge of the object being tested, is established to approximate the phase data. In this work a fourth degree polynomial was used to fit the phase.

A population of vectors was generated to carry out the optimization process. Each vector was formed by a codified string of polynomial coefficients. Then, the population of vectors was evolved using CR, F, and K.

The DE technique works successfully where other techniques fail (Synchronous and Fourier methods). This is the case when a noisy, wide bandwidth and/or closed fringe pattern is demodulated. Regularization techniques can be used in these cases but DE technique has the advantage that the cost function does not depend upon the existence of derivatives and restrictive requirements of continuity (gradient descent methods). Since the DE works with a population of possible solutions instead of a single solution, it avoids falling in a local optimum. Additionally, no filters and no thresholding operators were required, in contrast with the fringe-follower regularized phase tracker technique.

The DE has the advantage that if the user knows prior knowledge of the object shape, then a better suited fitting parametric function can be used instead of a general polynomial function. Additionally, due to the fact that the DE technique gets the parameters of the fitting function, it can be used to interpolate sub-pixel values and to increase the original phase resolution or interpolate where fringes do not exist or are not valid. A drawback is the selection of the optimal initial DE parameters (such as population size, F, K) that can increase the convergence speed.

Acknowledgements. We wish to thank the Centro de Investigación en Computación of the I.P.N. by the support to accomplish this project as well as the Centro de Investigaciones en Optica during the image recollections and tests. J. Vielma and L. Gomez thanks CONACYT by the scholarship received to complete his doctoral studies. H. Sossa thanks the SIP-IPN under grant 20091421 for the support. H. Sossa also thanks CINVESTAV-GDL for the support to do a sabbatical stay from December 1, 2009 to May 31, 2010. Authors thank the European Union, the European Commission and CONACYT for the economical support. This paper has been prepared by economical support of the European Commission under grant FONCICYT 93829. The content of this paper is an exclusive responsibility of the CIC-IPN and it cannot be considered that it reflects the position of the European Union. Finally, authors thank the reviewers for their comments for the improvement of this paper.

References

1. Martín, F., et al.: New advances in Automatic Reading of VLP's. In: Proc. SPC-2000 (IASTED), Marbella, España, pp. 126–131 (2000)
2. Malacara, D., Servin, M., Malacara, Z.: Interferogram Analysis for Optical Testing. Marcel Dekker, New York (1998)
3. Malacara, D.: Optical Shop Testing. Wiley, New York (1992)
4. Creath, K.: In: Wolf, E. (ed.) Progress in Optics, vol. 26, p. 350. Elsevier, Amsterdam (1988)
5. Creath, K.: In: Robinson, D., Reid, G.T. (eds.) Interferogram Analysis, p. 94. IOP Publishing, London (1993)
6. Takeda, M., Ina, H., Kobayashi, S.: Fourier–transform method of fringe–pattern analysis for computer–based topography and interferometry. Journal of Optical Soc. of America 72, 156–160 (1981)

7. Su, X., Chen, W.: Fourier transform profilometry: a review. Optics and Lasers in Engineering 35(5), 263–284 (2001)
8. Womack, K.H.: Interferometric phase measurement using spatial synchronous detection. Opt. Eng. 23, 391–395 (1984)
9. Servin, M., Rodriguez–Vera, R.: Two dimensional phase locked loop demodulation of interferograms. Journal of Modern Optics 40(11), 2087–2094 (1993)
10. Ghiglia, D.C., Romero, L.A.: Robust two–dimensional weighted and unweighted phase unwrapping that uses fast transforms and iterative methods. Journal of the Optical Society of America A 11(1), 107–117 (1994)
11. Su, X., Xue, L.: Phase unwrapping algorithm based on fringe frequency analysis in Fourier-transform profilometry. Optical Engineering 40, 637–643 (2001)
12. Servin, M., Cuevas, F.J., Malacara, D., Marroquin, J.L., Rodriguez-Vera, R.: Phase unwrapping through demodulation by use of the regularized phase–tracking technique. Applied Optics 38(10), 1934–1941 (1999)
13. Servin, M., Marroquin, J.L., Cuevas, F.J.: Demodulation of a single interferogram by use a two-dimensional regularized phase-tracking technique. Applied Optics 36(19), 4540–4548 (1997)
14. Villa, J., Servin, M.: Robust profilometer for the measurement of 3-D object shapes based on a regularized phase tracker. Optics and Lasers in Engineering 31(4), 279–288 (1999)
15. Quiroga, J.A., Gonzalez-Cano, A.: With a Regularized Phase-Tracking Technique. Applied Optics 39(17), 2931–2940 (2000)
16. Cuevas, F.J., Servin, M., Stavroudis, O.N., Rodriguez-Vera, R.: Multi–Layer neural network applied to phase and depth recovery from fringe patterns. Optics Communications 181(4-6), 239–259 (2000)
17. Cuevas, F.J., Servin, M., Rodriguez-Vera, R.: Depth object recovery using radial Basis Functions. Optics Communications 163(4-6), 270–277 (1999)
18. Cuevas, F.J., Sossa, J.H., Servin, M.: A parametric method applied to phase recovery from a fringe pattern based on a genetic algorithm. Optics Communications, Vol 203(3-6), 213–223 (2002)
19. Servin, M., Marroquin, J.L., Cuevas, F.J.: Fringe-follower regularized phase tracker for demodulation of closed-fringe interferograms. Journal of the Optical Society of America A 18(3), 689–695 (2001)
20. Price, K.V., Storn, R.M., Lampinen, J.A.: Differential Evolution - A Practical Approach to Global Optimization. Springer, Heidelberg (2005)
21. Storn, R., Price, K.: Differential Evolution - A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. Journal of Global Optimization 11(4), 341–359 (1997)
22. Li, X.: Efficient Differential Evolution using Speciation for Multimodal Function Optmization. In: Genetic and Evolutionary Computation Conference. Proceedings of the conference on Genetic and evolutionary computation, Washington, D.C., USA (2005)
23. Robic, T., Filipic, B.: DEMO: Differential Evolution for Multiobjective. In: Proceedings of the 3rd International Conference on Evolutionary MultiCriterion Optimization (EMO), pp. 520–533 (2005)
24. Roger, L.S., Tan, M.S., Rangaiah, G.P.: Global Optimization of Benchmark and Phase Equilibrium Problems Using Differential Evolution. National University of Singapore, Singapore (2006)
25. Gamperle, R., Müller, S.D., Koumoutsakos, P.: A Parameter Study for Differential Evolution. In: Proceedings WSEAS International Conference on Advances in Intelligent Systems, Fuzzy Systems, Evolutionary Computation, pp. 293–298 (2002)

ANIMA: Non-conventional Brain-Computer Interfaces in Robot Control through Electroencephalography and Electrooculography, ARP Module

Luis F. Reina, Gerardo Martínez, Mario Valdeavellano, Marie Destarac, and Carlos Esquit

Department of Electronics Engineering, Del Valle de Guatemala University

Abstract. ANIMA has as a primary objective to compare three non-conventional human com-puter interfaces that comply with the industrial robot ST Robotics R-17 instructions. This mod-ule, Alpha Waves Related Potentials -ARP- explains how brain waves are obtained, processed, analyzed and identified depending on their frequency. This module makes use of the Open EEG Project's open hardware monitor for brain wave activity, called the modular EEG. The brain waves are obtained through an electrode cap complying with the international 10-20 system for electrode positioning. The brain waves are processed with a fast Fourier transform using a mi-crocontroller and analyzed in software identifying the alpha wave's contribution. A program identifies the amount of time that alpha wave generation was maintained through concentration, and instructions are sent to the robotic arm, executing one of four pre-defined routines. Thirty percent of the users attained control over the robotic arm with the human computer interface.

1 Introduction

Brain waves can be obtained through electroencephalography –EEG- using an electrode cap. The electrodes obtain the brain's electrical variations caused by the neuronal interaction. To be able to observe potentials (micro volts scale) from a specific region of the brain, a comparison between two electrodes is needed; one at an area of reference and the other at the area of interest. The international system 10-20 for electrode positioning was created to standardize the position of electrodes in areas of interest.

Brain waves are classified according to their frequency; delta (0.2-3.5 Hz), theta (3.5 - 7.5Hz), alpha (7.5 -13Hz), beta (13-28Hz) and gamma (28-40Hz) [1]. Beta can be divided in two regions: Beta 1 or beta Low (13-20.5 Hz) and beta 2 or beta High (20.5 a 28 Hz).

Alfa waves are produced in moments of relaxation and tranquility. Most people can produce them when they close their eyes and relax. However, maintaining the generation of alpha waves with eyes wide open is not an easy task [1].

Alpha waves are one type of brain waves detected by electroencephalography –EEG- and predominantly originate from the occipital lobe or parietal lobe during

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 71–79, 2010.

wakeful relaxation with closed eyes. Alpha waves are reduced with open eyes and drowsiness and sleep.

The graphic representation of the Fast Fourier Transform -FFT- of a wave is a diagram named Fourier spectrum in which the frequency and magnitude of each sinusoid component is represented [2].

ANIMA project has three different modules: the ocular module, motor module and the alpha wave related potentials module –ARP-. All modules were financed by CONCYT (National Council for Science and Technology, for its Spanish abbreviation). Each module makes its own implementation for moving the robotic arm R17; signals produced by the changes in the electric field of the eyes caused by their movement [3], brain waves related to motor tasks [4] and brain waves related to alpha activity. Comparison between the three modules could be done in terms of effectiveness and speed for controlling a Robotics R17 robotic arm. Research in Guatemala is starting to flourish in this area and projects like this one are the very first steps.

2 Experimental Design

Brain waves on the parietal lobe are amplified, filtered, sampled and digitalized using the open hardware of the “Open EEG Project” [5] for brain wave acquisition. Brain waves were analyzed by applying the FFT to identify the frequencies involved with concentration. The signals are analyzed using a Python [6] software application to identify the alpha waves produced by concentration. This software identifies the contribution of the theta, alpha, beta1 and beta2 to the frequency spectrum on the FFT. The application uses the alpha waves contribution to identify alpha waves concentration.

Finally, the same application sends instructions to the robotic arm according to the amount of time the user maintained the alpha concentration, thus, executing one out of the four predefined routines for the R17.

3 Methodology

Fig. 1 shows the four main steps for this paper. These four steps will be explained on the following sections.

3.1 Signals Acquisition

An electrode cap complying with the IS 10-20 was used to obtain the brain waves. Electrodes used were P3 and GND as the reference. A modular EEG was built which has two printed circuit boards –PCBs-, one analog for amplification and filtering, and one digital for communication with a PC. The design for both PCBs is available on the Open EEG Project webpage. The modular EEG PCBs were manufactured at “Universidad del Valle de Guatemala” –UVG- (Fig. 2) and bought in Olimex at the same time [7]. PCBs from both manufacturers were compared and the same results were obtained. An application was made using Python to calibrate the gain obtained with the PCBs.

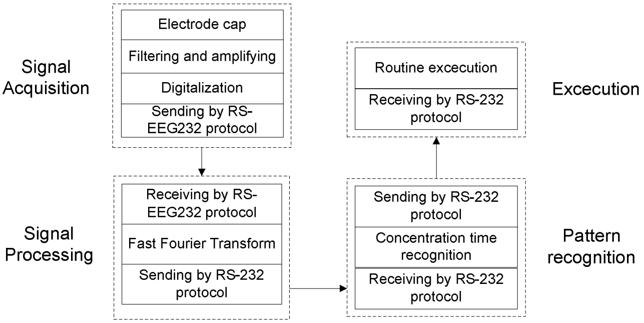

Signal Acquisition
Electrode cap
Filtering and amplifying
Digitalization
Sending by RS-EEG232 protocol
Signal Processing
Receiving by RS-EEG232 protocol
Fast Fourier Transform
Sending by RS-232 protocol
Excecution
Routine excecution
Receiving by RS-232 protocol
Pattern recognition
Sending by RS-232 protocol
Concentration time recognition
Receiving by RS-232 protocol

Fig. 1. Steps for the ARP module

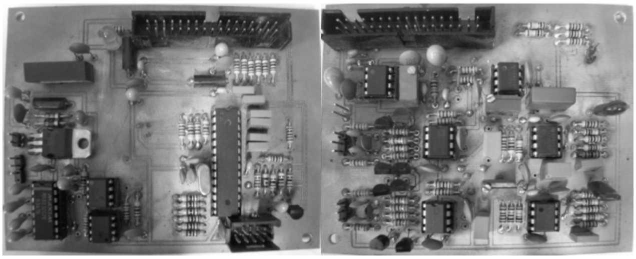


Fig. 2. Digital and analog PCB for Open EEG Projects modular EEG

3.2 Signal Processing

The FFT was calculated in a dsPIC microcontroller obtaining data from the modular EEG (AVR microcontroller) communicating through the RS-232EEG protocol and sending the results through RS-232 protocol to a PC (Fig. 3).

A dsPIC microcontroller with two RS-232 (UART) communication ports was used with a 10MHz 8xPLL, therefore the dsPIC could work at 80MHz. This allows the dsPIC to receive data from the modular EEG at 57600 baud and send new data to the PC at 115200 baud. This speed is needed so the FFT can be calculated and the data can be sent to a PC in real time. A 256-byte array is sent containing information about the frequency spectrum (magnitude for the 0-128 Hz range) with 1 Hz resolution.

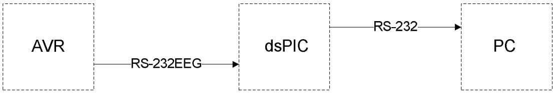


Fig. 3. Communication between the Modular EEG boards and PC

3.3 Pattern Recognition

An application was developed using Python to receive the dsPIC data over RS-232 protocol. The application discards the data related to frequencies above 30 Hz. The remaining region was divided into frequency regions according to brain waves classification: theta, alpha, beta1 and beta2. For each region an integral of the spectrum was computed. From these integrals the normalized contribution percentage was calculated.

On the GUI (Graphic User Interface) a graph presents the normalized contribution percentage for the four frequency bands (Fig. 4).

Using this application the brain waves are processed comparing the P3 and GND electrode positions from the IS 10-20 cap to identify concentration periods. For a period to be valid there are two requirements: the normalized contribution percentage for alpha waves must be above a threshold of 40 percent (0.4), and a difference of normalized contribution percentage of at least 20 percent (0.2) between alpha waves and the other brain waves (theta, beta1 and beta2) must be obtained. The application identifies concentration periods of 4, 6, 8 and 10 seconds.

The application generates different sounds (beeps) for each identified period. The sound acts as a feedback to the user because it is easier to achieve alpha state with closed eyes.

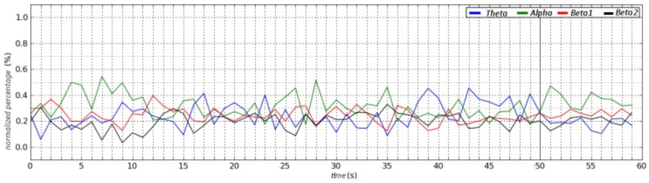


Fig. 4. Normalized percentage shown on the Python GUI

3.4 Execution

The GUI described in the previous section identifies alpha concentration periods of time. Another application was developed for the GUI to communicate with the robotic arm R17. This application has two goals: to send an order on each identified period and translate that order to the R17 programmed routines.

Each alpha state period of time was mapped to a R17 routine (Table 1.). The application contains defensive programming to prevent the R17 from going off its limits of movement.

Table 1. Called routine for each alpha state period

Seconds	Routine
4	Forward
6	Backward
8	Left
10	Right

4 Results

Fig. 5 shows an image where the difference in baud rate between the ATMEGA and dsPIC communication can be seen. Also the time of synchronization with the

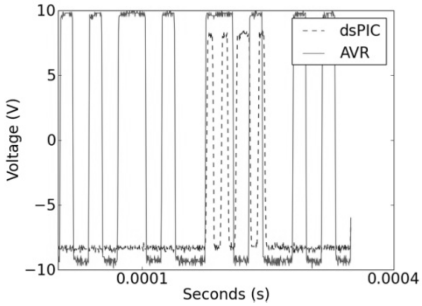


Fig. 5. Delay and difference on baud rate for communication between AVR y dsPIC

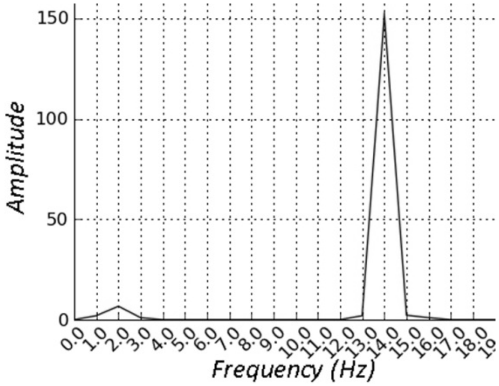


Fig. 6. FFT for a 14 Hz square wave signal

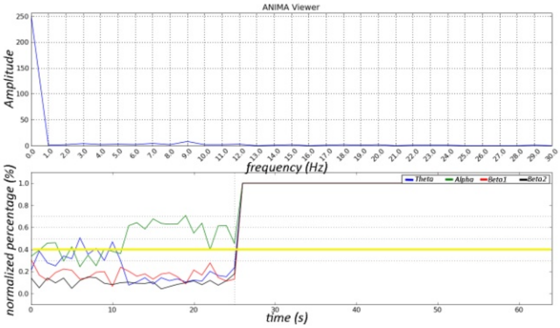


Fig. 7. Python application GUI with real time FFT

RS-232EEG can be seen. The real time FFT is shown in Fig. 6 using a 14Hz square wave signal as input.

The final application GUI is shown in Fig. 7, in which the real time FFT is integrated. In this figure it can be seen that the two required conditions are reached after 10 seconds. The yellow line stands for the threshold of alpha waves (green color), and the separation can be easily seen too. The times set for each task are based on various preliminary tests to assure that the alpha waves are being generated voluntarily and are not small involuntary periods of alpha wave generation. After generating alpha waves for a particular task the subject needs to generate 8 seconds of cumulative non-alpha waves to confirm the task, then the subject can start a new period of alpha wave generation and choose a different task.

Tests were made on twenty seven subjects, nine females and eighteen males (Table 2). Testing subjects were asked to close their eyes and then start counting from 1 to 10 until they listened to the first beep. If they achieved this, the second test was to achieve the fourth beep, else the test was ended. Subjects who achieved the fourth beep were able to successfully control all the tasks. In some cases the subject was not able to control the fourth beep (Task 4) but was able to control the other three tasks (Table 5). The order of appearance in Table 5 follows the order of the tests trial.

Table 2. ARP control test of success

	Yes	No
All	29.63%	70.37%
Men	38.89%	61.11%
Women	11.11%	88.89%

Table 3. Tasks test of success

	Task 1	Task 2	Task 3	Task 4
All	55.56%	33.33%	33.33%	29.63%
Men	61.11%	38.89%	38.89%	38.89%
Women	44.44%	22.22%	22.22%	11.11%

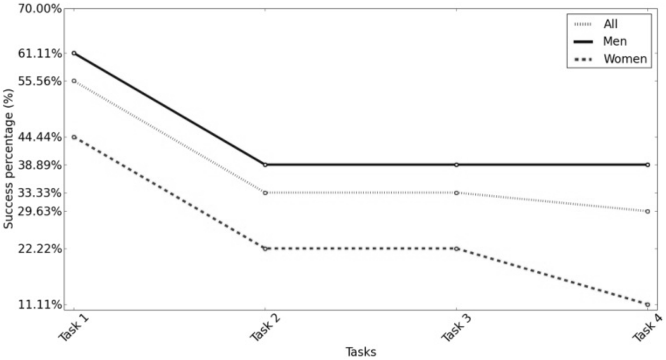


Fig. 8. Success percentage in task tests for women, men and all subjects

In Table 2 the success in the test is defined as the subject who can control the four tasks, the rest was considered as a failure.

In Table 3 the success for each task is shown. Males show equal difficulty in achieving the tasks two, three and four; either they achieved full control or just achieved to control the first task. It's interesting that women display a much bigger trouble achieving the fourth task. Also the percentage of women who accomplished task 1 is lower. This can be visually verified in Fig. 8.

Table 4 shows the improvement showed on the subjects that didn't achieve all four tasks. Improvement is successful if they managed to accomplish at least one task, in case they started without managing any. Women and men have similar percentage of

Table 4. Unsuccessful subjects that showed improvement

	Yes	No
All	36.84%	63.16%
Men	36.36%	63.64%
Women	37.50%	62.50%

Table 5. ARP control test results

Subject	T1	T2	T3	T4	Woman	Man
Subject 01	YES	YES	YES	YES		X
Subject 02	NO	NO	NO	NO	x	
Subject 03	NO	NO	NO	NO	x	
Subject 04	NO	NO	NO	NO		X
Subject 05	YES	NO	NO	NO		X
Subject 06	YES	YES	YES	YES		X
Subject 07	YES	YES	YES	YES		X
Subject 08	NO	NO	NO	NO		X
Subject 09	YES	YES	YES	YES		X
Subject 10	YES	NO	NO	NO	x	
Subject 11	YES	YES	YES	NO	x	
Subject 12	YES	YES	YES	YES		X
Subject 13	NO	NO	NO	NO	x	
Subject 14	NO	NO	NO	NO	x	
Subject 15	NO	NO	NO	NO		X
Subject 16	NO	NO	NO	NO		X
Subject 17	YES	NO	NO	NO		X
Subject 18	NO	NO	NO	NO		X
Subject 19	YES	YES	YES	YES		X
Subject 20	YES	NO	NO	NO		X
Subject 21	NO	NO	NO	NO	x	
Subject 22	YES	NO	NO	NO	x	
Subject 23	NO	NO	NO	NO		X
Subject 24	YES	NO	NO	NO		X
Subject 25	YES	YES	YES	YES		X
Subject 26	YES	YES	YES	YES	x	
Subject 27	NO	NO	NO	NO		X
					9	18

*T = Task

improvement, around 37 percent, which suggests that improvement doesn't depend on the skill to sustain alpha wave generation. This also suggests that subjects could manage the ARP control with proper training [8], but further studies are required to verify these hypotheses. It's interesting to notice that based on some additional tests, it was observed that alpha waves are affected by group pressure, or any distraction, e.g. music, people talking, or strong noises.

5 Conclusion

The hardware and software proposed implements a BCI to control the robotic arm R17 using alpha waves concentration by means of open EEG hardware and software developed at Universidad del Valle de Guatemala.

Thirty percent of the users were able to control the robotic arm through the ARP module. Thirty seven percent of the testing subjects showed an improvement on the BCI control.

The system is to be used as a base for a much complex BCI, as for now it represents the beginning of a non-conventional method of robot control that can be applied to different fields of study. The industrial robotic arm was just used to test the BCI, but it could be applied to any other machine or interface that complies with four basic movements.

The purpose of the entire application is to offer a neuro-feedback system that could be used to help handicapped people, but its actual state is still too basic.

6 Future Work

The project could use neuronal networks to recognize patterns for specific activities such as spacial and math-related activities, nevertheless results are not guaranteed.

The developed Python application can be used to make an exhaustive analysis of the brain waves. Data such as hemisphere asymmetry, normalized spectral contribution on each hemisphere, peak frequency and mean spectral area could be used as features. These data can be applied to a selection algorithm to identify which ones can be used for identifying a pattern. Then, a Bayesian Network classifier can be used to distinguish between each task [9] [10].

The use of gamma brain waves to determine high-stress states can be applied to avoid sending commands to the robotic arm and wait until the subject is back to the relaxed state so that commands can continue to be sent.

Some patterns could be found among non-standardized parameters using energy measurements of the spectrum (alpha, beta, theta, delta and gamma).

Mathematical tools like PCA, ICA, wavelets and multivariable statistical signal processing should be used to improve the processing algorithm.

Acknowledgment

This work was partially supported by the Guatemalan National Council for Science and Technology – CONCYT- (for its Spanish abbreviation) and by the Department of Electronics Engineering at Universidad Del Valle de Guatemala.

References

1. Johnston, W.: Silent Music: The Science of Meditation, p. 190. Fordham University Press, [trad.] Carmen Bustos (1997)
2. Transformada de Fourier: Escuela Universitaria de Ingeniería Técnica de Telecomunicación, Universidad Politécnica de Madrid (2009), http://www.diac.upm.es/acceso_profesores/asignaturas/tdi/tdi/transformadas/pdf/fourier1.pdf
3. Valdeavellano, M.R.: ANIMA: Métodos no convencionales de interfaz en el control de robots a través de la electroencefalografía y la electrooculografía: Módulo ocular. Universidad del Valle de Guatemala, Guatemala (2009)
4. Martínez, G.E.: ANIMA: Métodos no convencionales de interfaz en el control de robots a través de la electroencefalografía y la electrooculografía: Módulo motriz. Universidad del Valle de Guatemala, Guatemala (2009)
5. ModularEEG. OpenEEG Project (2008), http://openeeg.sourceforge.net/
6. Python Software Foundation. Python Programming Language (2010), http://python.org/
7. Open EEG Gadgets. OLIMEX Ltd. (2009), http://www.olimex.com/gadgets/index.html
8. Palke, A.: Brainathlon: Enhancing Brainwave Control Through Brain-Controlled Game Play, p. 37 (2003), http://www.webkitchen.com/brainathlon/files/thesis.pdf
9. Lee, J.C., Tan, D.S.: Using a low-cost electroencephalograph for task classification in HCI research. In: Symposium on User Interface Software and Technology on Sensing from head to toe, pp. 81–90 (2006)
10. Keirn, Z.A., Aunon, J.I.: A new mode of communication between man and his surroundings. IEEE Transactions on Biomedical Engineering 37(12), 1209–1214 (1990)

Efficient Roughness Recognition for Velocity Updating by Wheeled-Robots Navigation

Farid García and Matías Alvarado

Centro de Investigación y de Estudios Avanzados – IPN, Departamento de Computación, Av.
Instituto Politécnico Nacional 2508, San Pedro Zacatenco, CP 07360, México DF
farid@computacion.cs.cinvestav.mx, matias@cs.cinvestav.mx

Abstract. In this paper is shown that the Appearance-Based modeling is the best pattern recognition method for supporting the velocity updating of wheeled-robots navigation. Although Appearance-Based recognition algorithms have lower accuracy than the ones for detailed pattern recognition, they successfully classify terrain textures by regarding the average of the appearance. Actually, the detailed recognition algorithms success in recognizing patterns depicted with lines, dots or borders, but they fail for recognizing patterns where the average appearance is required. As human driving experience shows, the assessment of the average appearance is needed for velocity updating during navigation on outdoor terrains. Human drivers make the velocity adjusting based on an estimation of the terrain average appearance. Hence, as the experimental result illustrate, the algorithms for average appearance recognition are the best option for training wheeled-robot for velocity updating while navigating over outdoor terrains.

Keywords: Roughness Recognition, Velocity Updating, Wheeled-Robots Navigation.

1 Introduction

Outdoor autonomous robots are relevant for terrain exploration missions. The terrain difficulties of solar system planets –like Mars–, to move through terrains having soil, rocks and slopes, requires the usage of robots with the highest degree of autonomy to overcome such difficulties [1]. As well, in Earth exploration missions where human lives may be in dangerous circumstances, the autonomous robots are as well required. For instance, search of landmines or exploration of active volcano craters. Autonomous navigation on outdoor terrains is highly complex, obstacle detection and avoidance as well as the terrain features information for no slides, are both required. Environment data must be accurate and quickly processed by the robot's navigation systems. Besides, when data from human remote controllers is not quickly available, the autonomous robots should be equipped for convenient reactions, particularly in front of unpredicted circumstances. Actually, beyond the obstacle location and avoidance, the robot's velocity control, regarding the terrain features, has been few attended and it is a weakness for efficient and safe navigation nowadays.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 80–89, 2010.

The classification of terrain roughness has just recently been a bit more attended [2]. In [2] a path over a rough terrain is generated with a terrain-based criterion function, and then the robot is controlled so as to move on the chosen path. In [3] the navigation strategy assesses the terrain's features of roughness, slopes and discontinuity. Larson et al. [4] analyze the terrain roughness by means of spatial discrimination which then is (meta-) classified. In [5] roughness recognition is by using artificial vision, so novel textures recognition is later to an off-line recognition training from sample texture. Pereira et al. [6] plotted maps of terrains incorporating roughness information that is based on the measurement of vibrations occurring in the suspension of the vehicle; this online method can recognize textures at the moment the vehicle passes over them, what is a limitation for remote recognition.

For the purpose of velocity updating for autonomous navigation on rough terrains we claim that is not required to identify textures at high-detail level. Actually, as analyzed in Section 3, high precision recognition methods like Local Binary Patterns (LBP) [7], or Advanced Local Binary Patterns with Rotation Invariance (ALBPRI) [8], having 97.54% and 99.64% of respective efficacy, do not well performed accounting recognition of outdoors terrains textures. The listed accuracy percentages correspond to tests carried out on the texture image database of the Columbia Utrecht Reflectance and Texture Database [9], which is the most common benchmark used for testing texture recognition algorithms.

The LBP and ALBPRI methods have good performance for texture recognition. But these works do not mention anything about recognition of new textures, that is, nothing is said about how a different texture from the texture training set is classified. They just verify if the testing textures belong to any class of the texture training set, i.e., they only give two result values, false and true.

For our purpose, we desire to determine how similar the set of test textures and the set of training textures they are. The Appearance Based Vision (ABV) [10] method having 75% of detailed texture recognition efficacy is good enough for the velocity updating during outdoor navigation as results show in Section 3.2. Although ABV does not take into account the fine details of textures, it captures the so called average appearance of the textures. In other words, with a testing texture, even if it is a new texture, the ABV method compares it with the classes of the training set and indicates the texture class that resembles more to them, according to the average appearance.

In this paper is proposed to improve the process of robot velocity adaptation, by regarding the terrain features and imitating as human beings do. Humans use a quick imprecise estimation of the terrain features but enough to navigate without slides or falls. The human's estimation on the right velocity to safe navigate on irregular terrains is via imprecise but enough surface texture recognition [3]. Actually, we show below that concerning terrains exploration for robot navigation, the highest precision methods for texture recognition are not the adequate but failed.

Surface textures are captured via artificial vision, after image processing the estimation of texture class is gotten as well as the slopes inclinations. The algorithm's output indicates the velocity the robot can move depending on the terrain features. Bright and uniform lighting during navigation is required to guaranty consistent roughness recognition; therefore the presence of shadows, which treatment is a hard task to pattern recognition [11] is out of the scope of this work.

The rest of the article is organized as follows: Section 2 summarizes the closest antecedents in the field of texture recognition; then, the method and architecture of the fuzzy neural network for velocity updating is introduced. Section 3 describes tests and experimental results. A brief discussion is in Section 4, and the paper ends with conclusions.

2 Outdoor Terrains Recognition

Texture recognition is an issue that has been studied extensively; the local binary pattern-based methods are widely used for its good performance in the recognition of textures. Textures are modeled with multiple histograms of micro-textons; the micro-textons are extracted with a local binary pattern operator.

LBP [7] is a gray-scale invariant texture primitive statistic. For each pixel in an image, a binary code is produced by thresholding its neighborhood with the value of the center pixel. A histogram is created to collect up the occurrences of different binary patterns. LBP can be regarded as a micro-texton operator. At each pixel, it detects the best matching local binary pattern representing different types of curved edges, spots, flat areas, etc. After scanning the whole image to be analyzed, each pixel will have a label corresponding to one texton in the vocabulary. The histogram of labels computed over a region is then used for texture description.

Conventional LBP just considers the uniform patterns in the images. It discards important pattern information for images whose dominant patterns are not uniform patterns. ALBPRI [8] proposes a new rotation and histogram equalization invariant texture classification method by extending the conventional LBP approach to reflect the dominant pattern information contained in the texture images and capturing the spatial distribution information of dominant patterns.

These methods do well recognize patterns depicted with lines, dots or borders, but fails for recognizing different depicted appearances. When human drivers drive vehicles, they do not inspect the terrain textures with a magnifying glass nor take a look at a small distance to account details of particular lines, dots or borders; they just estimate the texture roughness basing on previous pattern recognition experience while driving [3].

Human beings classify textures according to past experience; when human drivers find a novel terrain texture, they employ their experience to estimate how rough the novel texture is. Then, they decide how fast they can drive without slide risks. By using fuzzy logic the human process for identifying the terrain roughness can be modeled in such a way to be used by the robot mimicking this human ability. To imitate the human experience during terrain recognition for navigation, it is clever to pay attention in convenient methods for recognizing the surface appearance average, and such that not lost in unnecessary details for the outdoors navigation purpose. Moreover, frequently the surfaces details recognition is computationally high-cost and it should be avoided for autonomous navigation. Actually, the Appearance Based Vision method is good enough for the velocity updating during outdoor navigation.

2.1 Appearance-Based Vision

The ABV method gets the principal components of image distribution, namely, the eigenvectors of the covariance matrix of the object images set. The ordered eigenvectors fashion the features accounting and charectizing the variation among the different images. The use of ABV for object recognition involves the next operators and operations.

Let $\{\mathbf{I}_1,\ldots,\mathbf{I}_N\}\subset\mathbf{R}^{n\times m}$ the set of training images, all the images are stacked so that we obtain the set $\{\phi_1,\ldots,\phi_N\}\subset\mathbf{R}^{n\cdot m}$. The vectors are normalized with $\tilde{\phi}_i=\phi_i/\|\phi_i\|$. The average vector is computed, $\mathbf{C}=\frac{1}{N}\sum_{i=1}^{N}\tilde{\phi}_i$. The images are centered by subtracting the average vector with each image; resulting vectors are placed to form the matrix $\mathbf{\Phi}=\left[\tilde{\phi}_1-\mathbf{C},\cdots,\tilde{\phi}_N-\mathbf{C}\right]$. The covariance matrix is computed, $\Omega=\mathbf{\Phi}\mathbf{\Phi}^T$ and its eigenvalues and eigenvectors are calculated. The eigenvectors are ordered in a decreasing fashion according to the eigenvalues, where they are placed as columns of matrix Ψ. All the training images are projected to the eigenspace with $\theta_i=\Psi^T(\tilde{\phi}_i-\mathbf{C}),i=1,\ldots,N$.

In the recognition phase the testing image is projected into the eigenspace and a supervised neural network classifies the image. In other words, let $\mathbf{I}_t$ the testing image, it is stacked and normalized, $\tilde{\varphi}_t$. Then, it is projected to the eigenspace with $\omega_t=\Psi^T(\tilde{\varphi}_t-\mathbf{C})$. A supervised neural network classifies ω_t.

During outdoors navigation, human drivers estimate the convenient vehicle velocity by regarding their previous experience when driving on similar terrain textures. In other words, human drivers estimate how rough, in average, the terrain is, instead if specific texture details are recognized. Human drivers that navigate on uneven terrains do not need to learn, or to know, about specific details but on the textures appearance average. The average recognition of textures, as the humans do, is the behavior that is mimicked and implemented in order to strengthen the robot navigation abilities.

2.2 The Fuzzy Neural Network for Velocity Updating

For robot velocity updating according to the terrain features, our proposal sets to imitate as human beings do. For safe navigation on irregular terrains, the human's velocity estimation is via imprecise but enough surface texture recognition [3]. When a human driver observes a novel terrain texture, uses his experience to estimate how rough the texture is; then decides the convenient car driving velocity. Thus, in the first step, the terrain's textures are neural-net-clustered in a roughness meta-class: a Supervised Neural Network (SNN) classifies textures; then, a Fuzzy Neural Network (FNN) makes a roughness meta-classification from the terrain texture class. By adding the texture roughness setting and the slope data, the FNN matches each terrain roughness with the corresponding velocity meanwhile the robot navigates safely.

For detection of slopes inclination, an infrared sensor located in the frontal part of the robot does parallel ray projection to the robot's motion; the other sensor projects its ray directly to the floor perpendicular to the first sensor. The inclination angle of slopes is computed by trigonometric operations. The off-line and on-line steps to update velocity regarding the terrains roughness and the inclination slopes while navigating are next described:

Off-line training steps

1) Select and model the representative outdoor textures images of the robot's environment. The images are captured while the robot is stationary.
2) Train the SNN to learn the texture classification established by the human expert driver.
3) Train the FNN to determine the velocity regarding the texture classes as well as the inclination angle of slopes, according to an expert driver's directives (build the fuzzy sets and make the inference IF-THEN rules system).

On-line steps

4) Acquisition of terrain images from the robot while it is in motion.
5) The SNN classifies the texture, this information is forwarded to the FNN.
6) The FNN inputs are the texture class and the slope inclination angle. The FNN indicates the updated velocity to the robot mechanical control system.
7) The cycle is repeated as the robot moves, and the velocity is cycle updated.

Following is the architecture of the five-layer FNN. The terrain features recognition followed by the robot velocity tuning is as shown in Fig. 1.

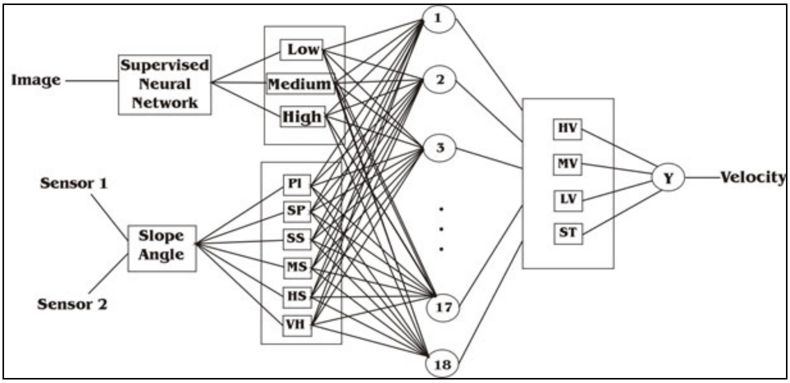


Fig. 1. The Fuzzy Neural Network

The texture class and slope input data are assessed to adjust the velocity that is the FNN output data. The FNN first layer inputs are the slope size and the texture class, the second layer sets the terms of input membership variables, the third sets the terms of the rule base, the fourth sets the term of output membership variables, and in the fifth one, the output is the robot's velocity. The textures roughness is meta-classified in three fuzzy sets, High (H), Medium (M) and Low (L). The inclination angles of slopes are meta-classified in six fuzzy sets: Plain (Pl), Slightly Plain (SP), Slightly Sloped (SS), Moderato Sloped (MS), High Slope (HS) and Very High (VH). The FNN output values are either: High Velocity (HV), Moderate Velocity (MV), Low

Velocity (LW) or Stop (ST). Membership functions of the input and output variables terms denote the corresponding texture roughness, slope inclination angle and velocity, respectively. The FNN output sets the velocity the robot can move safely.

The fuzzy-making procedure maps the crisp input values to the linguistic fuzzy terms with membership values in [0,1]. In this work the trapezoid membership functions (MF) for texture variable and the triangle MF for angle variable are respectively used. Taking X, Y, Z as variables of the respective predicates, the general form of inference rules is:

IF *Slope angle* is X **AND** *Roughness* is Y **THEN** *Velocity* is Z.

The inputs parameters are the slope angle and roughness, and the output is the estimated velocity. The de-fuzzy procedure maps the fuzzy output from the inference mechanism to a crisp signal. When the robot finds a slope steeper than the allowed threshold, it stops, and evaluates which movement to make, whose decision concerns to path planning.

3 Experimental Steps

A car-like Bioloid robot transformer kit [12] is used, which uses a processing unit, four servomotors for power transmission to the wheels, two infrared sensors located in the robot front, and a wireless camera on top-front of the robot. The robot dimensions are 9.5 cm width per 15 cm length. In these experiments the SNN is trained with terrain textures from images in Fig. 2. In this platform it is used a personal computer (PC) and the processor of the robot, to form a master-slave architecture, communicated wirelessly. On the PC is implemented and executed the velocity estimation algorithm. The robot, on one hand, reports to the PC the sensors readings and wirelessly transmits the images captured by video camera, on the other hand updates the velocity in accordance with instructions that the PC communicates it.

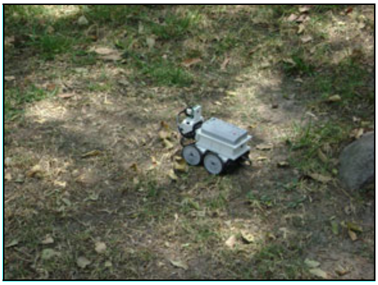


Fig. 2. Robot's navigation on outdoor surface

The navigation tests focused on velocity updating, the vehicle navigates on the terrain shown in Fig. 2; the robot recognizes textures from the captured images. As soon as significant change of terrain textures is detected the PC indicates the robot to update its velocity according to the terrain texture currently recognized.

In here reported experiments textures are respectively modeled with ABV, LBP, ALBPRI and a two-dimensional Fourier transform method for roughness classification of cast surfaces (FCS) [13], which it is deserved to assess the cast surface quality. Terrain images are used during the robot's training for texture recognition of the surfaces it navigates. There were conducted 15 tests with each method, the texture images are garden ground covered by a thin layer of dust, and dispersed little rocks; the grass is 2-centimeter cut height and dry; the paving stone contains leafs, tree branches with a thin dust cover.

3.1 Algorithm's Performance

The expected robot behavior is that the robot increases its velocity, from low to high, when it is detected grass, ground and paving stone. That is, the robot must move slowly on grass, fast on paving stone; on ground, faster than in grass but slower than in paving stone. The robot's navigation velocity updating results, regarding the surface textures, are displayed in Table 1, whose data are average velocity values. It shows that the robot moves faster on paving stone than on grass, and in turns, faster on ground than on grass.

Table 1. Velocity results in centimeters per second units

Method/Texture	Ground	Grass	Paving Stone
ABV	9.03	4.4	11.43
ALBPRI	9.68	7.13	11.7
LBP	12.18	11.7	12.85
FCS	1.08	4.3	2.16

The worst performances were with the FCS and LBP methods, the robot did not adjust its velocity as expected. The methods ABV and ALBPRI had better performance, because the robot adjusted its velocity according to the expected behavior. However, with ALBPRI, the estimated velocity for grass is relatively high, because it is close to ground value. With ABV, the velocities are slightly lower than those of ALBPRI, but ABV estimates a lower velocity for grass.

During the training attempts using images from a low roughness texture wall, in addition to other textures occurs that both, the ALBPRI and LBP methods do miss to identify the diverse surface textures, hence misclassifying all of them as wall texture. A likely explanation is that both recognition methods are based on the detection of borders, edges and dots in order to guaranty well recognition performance. Because the wall images are uniformly plain, the required graphical elements for methods well performance are not present. It provokes the methods losing to recognize surfaces not having clearly marked lines, borders or dots. However, wall images were replaced by another kind of wall images with a certain line pattern.

On the other hand, the ABV method does properly recognize the surface textures changes, and then the robot's velocity is updated, while it is displacing, according to the surface features. Therefore, experimental conclusion is that ABV advantages to LBP and ALBPRI to model and recognize the physical average appearance of textures. By using ABV the recognition of, what we call, the texture average among images is competent. ABV method well recognizes the wall plain texture average, as well as the texture average of grass, ground or paved. Thus, the textures average recognition supports the robots navigation on outdoors terrains. Car drivers do speed update by regarding the terrain texture average, i.e., velocity adjustments are according to the terrain appearance average variations, which are relevant to human drivers by navigating, and disregarding the irrelevant specific lines, dots or borders for navigation.

The FCS method shows high performance in classification by recognizing polished surfaces like glass, steel or plastics. The method applied to terrain images failed hardly. All the results were wrong; the method misclassified the terrain textures as low roughness textures. The plausible explanation is that this method works on polished textures that require high precision during recognition. But it misses on rough surfaces like grass, ground, soil or pave that do not demand a high precision during textures recognizing.

3.2 Simulation of Real Car Navigation

In this set of tests, images of the terrain textures are taken from a video film recorded by a video camera placed two meters above the floor on a car’s roof. The camera recorded the car’s path, under a visual field similar to that of a human driving a real car, see Fig. 3.

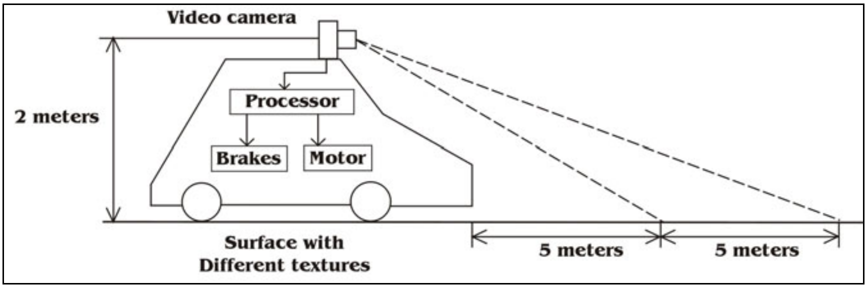


Fig. 3. Car vision/recognition system

The outdoor terrains textures images are loose stones, ground with grass, ground, asphalt and concrete, on which the truck was moving through. These images are used to train the SNN by using the ABV model for textures treatment. The experience of a human drive allows for defining the classes of textures and the respective car velocity. Tests of simulated car navigation are regarding that the vehicle maximum speed it can reach is 50 km/hr. The velocities are as follow: on loose stones, velocity is smaller than on ground with grass, and it is smaller than on sole ground, and in turns, it is smaller than on paved ways; these results are shown in Table 2, showing the average of the velocities resulting in specific experiments.

The minimum and maximum velocities recorded for loose stones are 10.06 km/hr and 18.91 km/hr, respectively. The velocities estimated for ground with grass are a little higher. Mostly the velocity remains at 18.11 km/hr. Ground with grass texture is less abrupt than loose stones. The grass does not cover the entire surface but there are

Table 2. Velocity updating results

Texture	Velocity km/hr
Loose stones	10.65
Ground with grass	18.47
Ground	27
Concrete	43.79
Asphalt	48.91

holes with ground, usually small, even so a car can overcome them at low-speed motion by the time it avoids damaging vibrations in the vehicle. Even when grass is a texture that favors slipping and skidding, but unlikely the loose stones, the wheels surfaces of the car have better contact with ground, thus the risk of skidding is smaller, but higher than in the next textures. For ground textures, the velocity remains constant in almost the entire path at 27.61 km/hr. The surface of ground texture is covered with dust and very small stones, and is almost flat, so the vehicle can move fast without being affected by strong vibrations, even the small stones and dust in the surface could make the car to skid. Velocity for concrete texture remains constant, 44.44 km/hr in almost the entire path. Finally, the velocities estimated for the asphalt remain without changes, 48.91 km/hr throughout the entire path. The textures of concrete, paving city streets, and the asphalt covering roads are very similar. The covering with this matter create a uniform surface, without holes and slopes, so avoiding the car skids.

An additional aspect to consider concerns with the vehicle's computing capacities for processing the texture images and the velocity updating in real time. Actually, 1) determine the range of the camera to capture images of the surface, and 2) the sampling time given the progress of the vehicle. The acquired images have a resolution of 480×640 pixels in grayscale. The microprocessor employed was a Centrino Core 2 Duo at 2GHz and 1.99Gb RAM. The processor spends 0.3 seconds for both image processing and velocity updating.

Actually, for efficient velocity control it must be considered the 0.3 seconds the process spends for texture recognition and velocity updating, by assuming that the maximum speed is around 50 km/hr, hence the vehicle will advance 5 meters. As shown in Fig. 3, the camera must process the next 5-meter road segment before the vehicle passes on. That is, when the vehicle moves the first 5-meter stretch, the computer processes the image of the posterior 5-meter stretch. When the second stretch processing is finished, the vehicle would have started to move in the second stretch. This cycle is successively repeated.

4 Discussion

Velocity updating according to the surface roughness, is a subject that has not been fully addressed. Most of the works focus on the detection and obstacle avoidance problem. For instance, Labert et al. [14] use a probabilistic modeling to avoid or to mitigate eventual collisions, regarding the environment perception, by updating a robot braking action. Selekwa et al. [15] and Ward & Zelinsky [16] addressed the navigation and path planning of an autonomous robot which varies the velocity according to the proximity of obstacles detected by infrared sensors. So far, all the referred works on outdoors autonomous robots do not include in their proposals information from terrain surface roughness during navigation.

Conclusions. The efficiency of algorithms for recognition of roughness textures is the key point for allowing velocity updating. According to results the appearance average instead of high-detailed recognition is the requisite for velocity updating on rough terrains. A clever issue is the human mimicking about the recognition and decision making for velocity updating. Human drivers make quick terrain recognition but enough to a right speed updating during navigation. The computationally low-cost and easy implementation of the algorithms make this approach suitable for velocity updating of wheeled-robots during autonomous navigation on outdoor terrains.

References

1. Bajracharya, M., Maimone, M.W., Helmick, D.: Autonomy for Mars Rovers: Past, Present, and Future. Computer 41, 44–50 (2008)
2. Ishigami, G., Nagatani, K., Yoshida, K.: Path Planning for Planetary Exploration Rovers and Its Evaluation Based on Wheel Slip Dynamics. In: IEEE International Conference on Robotics and Automation, pp. 2361–2366 (2007)
3. Seraji, H., Howard, A.: Behavior-Based Robot Navigation on Challenging Terrain: A Fuzzy Logic Approach. IEEE Trans. Robot Autom. 18, 308–321 (2002)
4. Larson, A.C., Voyles, R.M., Demir, G.K.: Terrain Classification Using Weakly-Structured Vehicle/Terrain Interaction. Auton. Robot 19, 41–52 (2005)
5. Brooks, C.A., Iagnemma, K.: Visual Detection of Novel Terrain via Two-Class Classification. In: Proceedings of the 2009 ACM Symposium on Applied Computing, pp. 1145–1150 (2009)
6. Pereira, G.A.S., Pimenta, L.C.A., Chaimowicz, L., Fonseca, A.F., de Almeida, D.S.C., Correa, L.Q., Mesquita, R.C., Campos, F.M.: Robot Navigation in Multi-Terrain Outdoor Environments. Int. J. Robot Res. 28, 685–700 (2009)
7. Pietikäinen, M., Nurmela, T., Mäenpää, T., Turtinen, M.: View-Based Recognition of Real-World Textures. Pattern Recogn. 37, 313–323 (2004)
8. Liao, S., Chung, A.C.S.: Texture Classification by Using Advanced Local Binary Patterns and Spatial Distribution of Dominant Patterns. In: IEEE International Conference on Acoustics, Speech, and Signal Processing, pp. 1221–1224 (2007)
9. Columbia Utrecht Reflectance and Texture Database, `http://www1.cs.columbia.edu/CAVE//software/curet/`
10. Turk, M., Pentland, A.: Eigenfaces for Recognition. J. Cognitive Neurosci. 3, 71–86 (1991)
11. Kahraman, F., Stegmann, M.B.: Towards Illumination-Invariant Localization of Faces Using Active Appearance Models. In: 7th Nordic Signal Processing Symposium, p. 4 (2006)
12. Robotis Co., Ltd., `http://www.robotis.com`
13. Tsai, D.M., Tseng, C.F.: Surface Roughness Classification for Castings. Pattern Recogn. 32, 389–405 (1999)
14. Lambert, A., Gruyer, D., Pierre, G.S., Ndjeng, A.N.: Collision Probability Assessment for Speed Control. In: 11th International IEEE Conference on Intelligent Transportation Systems, pp. 1043–1048 (2008)
15. Selekwa, M.F., Dunlap, D.D., Shi, D., Collins, E.G.: Robot Navigation in Very Cluttered Environments by Preference-Based Fuzzy Behaviors. Robot Auton. Syst. 56, 231–246 (2008)
16. Ward, K., Zelinsky, A.: Acquiring Mobile Robot Behaviors by Learning Trajectory Velocities. Auton. Robot 9, 113–133 (2000)

Fingerprint Verification with Non-linear Composite Correlation Filters

Saúl Martínez-Díaz and Javier A. Carmona-Troyo

Instituto Tecnológico de La Paz, División de Estudios de Posgrado e Investigación,
Boulevard Forjadores de Baja California Sur No.4720, La Paz BCS, México
smdiaz@marinos.itlp.edu.mx, jcarmona@marinos.itlp.edu.mx

Abstract. Fingerprint recognition has been used from many years for identification of persons. However, conventional fingerprint recognition systems might fail with poor quality, noisy or rotated images. Recently, novel non-linear composite filters for correlation-based pattern recognition have been introduced. The filters are designed with information from distorted versions of reference object to achieve distortion-invariant recognition. Besides, a non-linear correlation operation is applied among the filter and the test image. These kinds of filters are robust to non-Gaussian noise. In this paper we apply non-linear composite filters for fingerprint verification. Computer simulations show performance of proposed filters with distorted fingerprints. In addition, in order to illustrate robustness to noise, filters were tested with noisy images.

Keywords: Fingerprint verification, nonlinear filters, correlation filters, pattern recognition.

1 Introduction

One of the most important biometric person identification techniques is based on fingerprint recognition. Fingerprints are unique and unchangeable to each individual; for this reason had been used since many years ago for this purpose. Basically two tasks are performed with fingerprint recognition systems: identification of a person or verification of his/her identity. In the first case the system searches into a database, if a similar fingerprint is found the person is identified. In the second case, the system verifies if a person is who he/she claims to be.

Generally, fingerprint recognition systems are based on minutiae (ridge endings and bifurcations) extraction [1]. With these methods preprocessing is required to remove noise, enhance the image and extract the features of interest. Extracted features are employed to identify the fingerprint.

Other methods that use local and global features of fingerprints have been proposed [2], [3], [4]. However, those techniques do not use all available information from images. Besides, with the above mentioned techniques, orientation of images is crucial in the process. Therefore rotation, poor quality and noisy images make difficult the recognition.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 90–97, 2010.

Another way to carry out fingerprints recognition is by correlating the entire test image with a single template or filter [5]. When the test image is equal to the template, correlation output is high (typically one); in other case correlation is low. Figures 1(a) and 1(b) show examples of correlation planes for two equal images and for two different images, respectively. Note that when images perfectly match a sharp peak is observed at origin of coordinates. A threshold can be established at the output to determinate whether the tested fingerprint is authentic or not. Correlation methods are shift-invariant and exploit all information from images. Moreover, by using several samples of the expected distortions of objects is possible to design distortion-invariant filters. The training images are synthesized in a single template and the test image is correlated with such template.

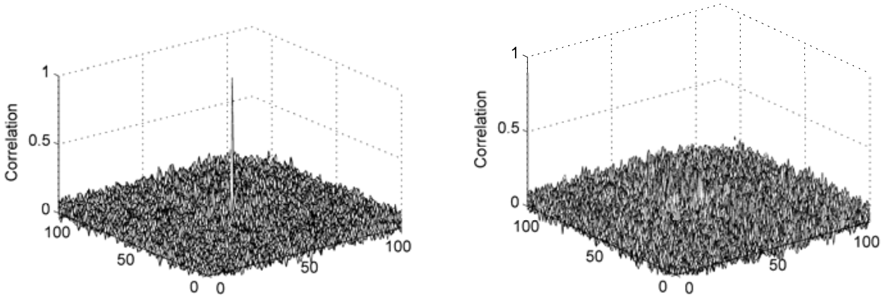


Fig. 1. Correlation planes obtained by a) exact match among images and b) not match among images

Normally, correlation filters are designed and optimized in a linear way. Linear filters used to be robust to Gaussian noise; however, real images are often corrupted by non-Gaussian noise. In practice, non-linear filters are more robust even for slight deviations from the Gaussian distribution. This kind of filters provides solutions in many cases where linear filters are inappropriate. In this paper we propose the use of novel non-linear composite filters. The filters are based on morphological and rank-order operations; its robustness to non-Gaussian noise has been shown.

The paper is organized as follows: In section 2 we review traditional composite filters. In section 3 we introduce non-linear filtering and proposed filters. In section 4 computer simulations are provided and discussed. Section 5 summarizes our conclusions.

2 Composite Filters

The simplest correlation filter, called Matched filter (MF), is a single image used as template, which is correlated with the test image [6]. It is known that the MF is very sensitive to small distortions of the object caused by variations in scale, rotation, or point of view. In order to overcome these problems, composite filters based on synthetic discriminant functions (SDF) were introduced [7].

2.1 Synthetic Discriminant Function Filters

Conventional SDF filters are a linear combination of MFs for different patterns. The coefficients of the linear combination are chosen to satisfy a set of constraints on the filter output requiring a previously specified value for each pattern used.

Suppose there are *N* training images from a true class, each image contains *d* pixels. The 2D arrays of the images are converted into 1D column vector by lexicographical ordering. These vectors are the columns of a matrix R of size *d*×*N*. The column vector u contains *N* elements, which are the desired values of the output correlation peaks corresponding to each training image. If the matrix (R^+R) is nonsingular, the conventional SDF filter can be expressed as follows:

$$h_{SDF} = R(R^+R)^{-1}u \ , \tag{1}$$

where superscript + means conjugate transpose. The main shortcoming of the linear SDF filters is appearance of sidelobes due to the lack of control over the whole correlation plane. In order to reject known objects from a false class, these objects can be included in the training set by setting zero in the corresponding values of vector u.

2.2 Minimum Average of Correlation Energy Filter

With the intention of suppress false correlation peaks, minimum average of correlation energy (MACE) filters were proposed [8]. MACE filters minimize the average correlation energy of the correlation outputs for a set of training images, satisfying at the same time the correlation peak constraints at the origin. Suppose that there are *N* training images, each image with *d* pixels. First, the 2D Fourier transform is performed on each training image and converted into 1D column vector. Then, a matrix X with *N* columns and *d* rows is constructed. The columns of X are given by the vector version of each transformed image. The frequency response of the MACE filter can be expressed as

$$h_{MACE} = D^{-1}X(X^+D^{-1}X)^{-1}u \ , \tag{2}$$

where the column vector u contains desired correlation peak values of the training images and the *d*x*d* diagonal matrix D contains the average power spectrum of the training images.

2.3 Optimal Tradeoff Filters

MACE filters maximize peak sharpness by minimizing correlation energy. However, tolerance to noise is not considered. In order to include noise tolerance, optimal tradeoff synthetic discriminant function (OTSDF) filters were introduced [9]. OTSDF filters allow a tradeoff among peak sharpness and noise tolerance by minimizing at the same time correlation energy and the output variance of correlation peak when the input images of the training set are corrupted by noise. The frequency response of the OTSDF filter can be expressed as

$$h_{OTSDF} = P^{-1}X(X^+P^{-1}X)^{-1}u \ , \tag{3}$$

where P = μD + (1-μ)C, μ ∈ [0,1]. The *d*x*d* diagonal matrix D contains the average power spectrum of the training images. C is a matrix of covariance obtained from noise realizations. The column vector u contains desired correlation peak values of the training images.

3 Nonlinear Filtering

Traditionally correlation-based filters use a linear correlation operation derived from minimization of the mean squared error (MSE). The correlation is computed between an input image and a shifted version of the target. On the other hand, minimization of the mean absolute error (MAE) leads to a nonlinear operation, called morphological correlation, which is computed as a sum of minima. This criterion is more robust when the noise has even slight deviations from the Gaussian distribution, and produces a sharper peak at the origin [10]. Also, local adaptive correlations based on rank order operations were proposed to improve recognition in images with non-Gaussian noise [11]. Recently, novel non-linear synthetic discriminant function filters (N-SDF) were proposed [12]. The filters are designed by applying logical operations among a set of training objects. Various objects to be recognized and rejected can be incorporated in the template synthesis. The morphological correlation is computed between the template and a test scene. The correlation is locally normalized to yield a desired output value. It was shown that nonlinear filters yield maximum correlation with objects utilized in the template synthesis. Besides, filters are robust in images corrupted by non- Gaussian noise.

3.1 Morphological Correlation

The proposed filtering is a locally adaptive processing of the signal in a moving window. The moving window is a spatial neighborhood containing pixels surrounding the central window pixel geometrically. The neighborhood is referred to as the *W*-neighborhood. The shape of the *W*-neighborhood is similar to the region of support of the target. The size of the neighborhood is referred to as $|W|$, and it is approximately taken as the size of the target.

Let $\{T(k,l)\}$ and $\{S(k,l)\}$ be a target image and a test scene respectively, both with Q levels of quantization. Here (k,l) are the pixel coordinates. The local nonlinear correlation derived from the MAE criterion between a normalized input scene and a shifted version of the target at coordinates (k,l) can be defined as

$$C(k,l) = \sum_{m,n \in W} MIN\left[a(k,l)S(m+k,n+l)+b(k,l),T(m,n)\right] \ , \tag{4}$$

where the sum is taken over the *W*-neighborhood. *a(k,l)* and *b(k,l)* are local normalizing coefficients, which take into account unknown illumination and bias of the target, respectively. The optimal coefficients with respect to the MAE can be estimated by minimizing the MSE between the window signal and the target. Their explicit estimates are given by:

$$a(k,l)=\frac{\sum_{m,n\in W} T(m,n)\cdot S(m+k,n+l)-|W|\cdot\overline{T}\cdot\overline{S}(k,l)}{\sum_{m,n\in W}\left(S(m+k,n+l)\right)^2-|W|\cdot\left(\overline{S}(k,l)\right)^2}\ , \tag{5}$$

$$b(k,l)=\overline{T}-a(k,l)\cdot\overline{S}(k,l)\ , \tag{6}$$

here $\overline{T}$ and $\overline{S}(k,l)$ are the average of the target and local window signal over the *W*-neighborhood at the *(k,l)'*th window position, respectively.

3.2 Nonlinear Synthetic Discriminant Function Filters

According to the threshold decomposition concept [13], a gray-scale image *X(k,l)* can be represented as a sum of binary slices:

$$X(k,l)=\sum_{q=1}^{Q-1}X^q(k,l)\ , \tag{7}$$

where $\left\{X^q(k,l), q=1,...Q-1\right\}$ are binary slices obtained by decomposition of the image with a threshold *q* as follows:

$$X^q(k,l)=\begin{cases}1, & if\ \ X(k,l)\geq q\\ 0, & otherwise\end{cases}\ . \tag{8}$$

Now, assume that there are *N* objects from the true class $\left\{T_i(k,l), i=1...N\right\}$ and *M* objects from the false class $\left\{P_j(k,l), j=1...M\right\}$. First, binary images are obtained by threshold decomposition of the training set. Next we construct the non-linear synthetic discriminant function filter (N-SDF) as logical combinations of the binary images. The composite filter can be expressed as:

$$H_{NSDF}(k,l)=\sum_{q=1}^{Q-1}\left[\bigcap_{i=1}^{N}T_i^q(k,l)\right]\bigcap\left[\overline{\bigcup_{j=1}^{M}P_j^q(k,l)}\right],\quad i=1...N,\ \ j=1...M\ , \tag{9}$$

where $\left\{T_i^q(k,l), q=1,...Q-1, i=1,...N\right\}$ and $\left\{P_j^q(k,l), q=1,...Q-1, j=1,...M\right\}$ are binary slices obtained by threshold decomposition from corresponding training images of true and false classes respectively. $\bigcup$ and $\bigcap$ represent the logical union and intersection, respectively. The neighborhood *W* is taken as the region of support of the composite filter. Finally, the nonlinear correlation in equation (4) is computed among the test image and the composite filter. The result is normalized by *u*/*s*. Here *u* is the desired value at the correlation output, and

$$s=\sum_{k,l\in W}H_{NSDF}(k,l)\ . \tag{10}$$

It can be shown that the composite correlation yields the value u at output correlation for objects belonging to the true class, while the output correlation peak for the false class objects is zero. Known false class images are normalized respect the target by using equations (5) and (6), before synthesize the template.

4 Computer Simulations

In this section computer simulation results obtained with the proposed filters are presented. A set of 200 fingerprints was utilized in the experiments. All images are 115x115 pixels at 256 levels of quantization. The performance of nonlinear filters is compared with that of MACE and OTSDF filters. For the OTSDF filter white noise is assumed and μ is set to 0.9. Figure 2(a) shows the fingerprint to be recognized (target) and figure 2(b) shows an impostor's fingerprint. Then N-SDF, OTSDF and MACE filters were designed with five rotated versions of target (-4, -2, 0 2 and 4 degrees) and five known false fingerprints.

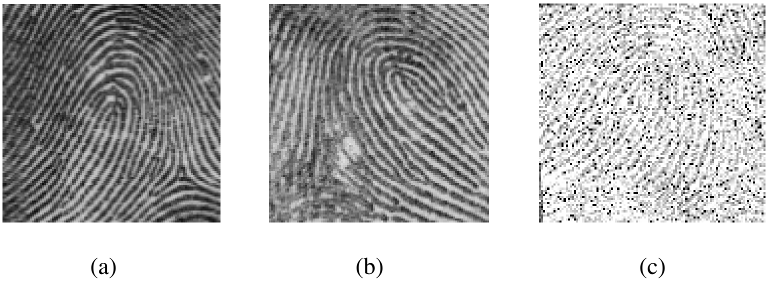


Fig. 2. (a) Fingerprint to be recognized (target). (b) False fingerprint. (c) An example of target corrupted by mixed additive Gaussian and impulsive noise. The mean and standard deviation of additive noise are 120 and 30 respectively. The probability of impulsive noise is 0.1.

As previously mentioned, correlation between filters and the versions of target used to synthesize the template are equal to one. Now, correlation was executed between filters and 194 unknown false fingerprints. Then, the minimum value at which all false objects are rejected with all filters was selected as threshold. Such threshold was set to 0.8.

Next, in order to test robustness to noise, correlation among composite filters and rotated versions of noisy target was computed. The noise was a mix of additive Gaussian and impulsive (salt and pepper) noise. The mean and standard deviation of additive noise were 120 and 40, respectively. The probability of impulsive noise was 0.1 with equal probability of occurrence for negative and positive impulses. Figure 2(c) shows an instance of target corrupted by mixed. To guarantee statistically correct results, 30 statistical trials of each experiment for different realizations of random processes were performed. Figure 3 shows performance of the filters. As can be seen, N-SDF filter is able to detect the target in all cases, even when scene is corrupted with non-Gaussian noise.

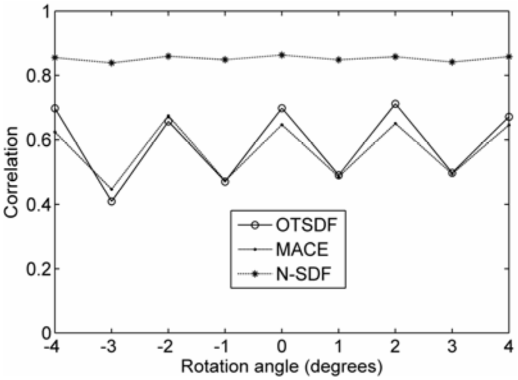


Fig. 3. Correlation output of MACE, OTSDF and N-SDF filters for rotated versions of target corrupted with mixed additive and impulsive noise

Table 1. Performance of composite filters in terms of maximum correlation peak when target is incomplete

% Of original image		Correlation peak	
	MACE	OTSDF	N-SDF
96 %	1.000	1.000	1.000
93 %	0.781	0.801	0.997
90 %	0.681	0.690	0.992
86 %	0.656	0.662	0.975
83 %	0.648	0.653	0.945
80 %	0.633	0.640	0.911
77 %	0.617	0.628	0.880
74 %	0.562	0.567	0.847
71 %	0.550	0.554	0.795
68 %	0.540	0.542	0.762

Often, because of fingerprint pressure differences, images are incomplete. For this reason we test performance of filters with incomplete fingerprints. First, pixels are removed at each edge of target. Then, correlation between designed filters and incomplete target is computed. Results are presented in table 1. Column 1 is the remaining percentage of original image. Column 2, 3 and 4 are the maximum correlation peaks for MACE, OTSDF and N-SDF filters, respectively. Note that correlation with N-SDF filter decreases slower than correlation with MACE and OTSDF filters. N-SDF yields a correlation value above threshold with less than 75 percent of original image.

5 Conclusions

In this paper, composite nonlinear filters for fingerprint verification were proposed. The filters are designed as a logical combination of given training images. Various properties of filters were tested. Their recognition performance and noise robustness were compared with those of conventional linear composite filters. Computer simulations illustrated an improvement in recognition of distorted fingerprints in heavy non-Gaussian noise situations, when the proposed filters were used. As well, proposed filters were capable of recognize even incomplete target. Further simulations can be done in order to test extensively non-linear filters.

References

1. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition. Springer, London (2009)
2. Jain, A.K., Prabhakar, S., Hong, L., Pankanti, S.: Filterbank-Based Fingerprint Matching. IEEE T. on Image Process. 9, 846–859 (2000)
3. Ross, A., Reisman, J., Jain, A.: Fingerprint Matching Using Feature Space Correlation. In: Tistarelli, M., Bigun, J., Jain, A.K. (eds.) ECCV 2002. LNCS, vol. 2359, pp. 48–57. Springer, Heidelberg (2002)
4. Cappelli, R., Maio, D., Maltoni, D., Nanni, L.: A Two-Stage Fingerprint Classification System. In: Workshop on Biometrics Methods and Applications, pp. 95–99. ACM, California (2003)
5. Venkataramani, K., Vijaya-Kumar, B.V.K.: Fingerprint Verification Using Correlation Filters. In: Kittler, J., Nixon, M.S. (eds.) AVBPA 2003. LNCS, vol. 2688, pp. 886–894. Springer, Heidelberg (2003)
6. VanderLugt, A.B.: Signal Detection by Complex Filtering. IEEE Trans. Inf. Theory. 10, 135–139 (1964)
7. Hester, C.F., Casasent, D.: Multivariant Technique for Multiclass Pattern Recognition. Appl. Opt. 19, 1758–1761 (1980)
8. Mahalanobis, A., Vijaya-Kumar, B.V.K., Casasent, D.: Minimum Average Correlation Energy Filters. Appl. Opt. 31, 1823–1833 (1987)
9. Refregier, P.: Filter Design for Optical Pattern Recognition: Multicriteria Optimization Approach. Optical Society of America 15, 854–856 (1990)
10. Maragos, P.: Morphological Correlation and Mean Absolute Error Criteria. In: Proc. Conf. IEEE Trans. Acoust. Speech Signal Process., pp. 1568–1571 (1989)
11. Kober, V., Alvarez-Borrego, J., Ovseyevich, I.A.: Adaptive Rank Order Correlations. Pattern Recognition and Image Analysis 14, 33–39 (2004)
12. Martínez-Díaz, S., Kober, V.: Nonlinear Synthetic Discriminant Function Filters for Illumination-Invariant Pattern Recognition. Opt. Eng. 47, 067201 (2008)
13. Fitch, J.P., Coyle, E.J., Gallagher Jr., N.C.: Median Filtering by Threshold Decomposition. IEEE Trans. Acoust. Speech Signal Process., 1183–1188 (1984)

Automatic Ultrasound Image Analysis in Hashimoto's Disease

Robert Koprowski[1], Zygmunt Wrobel[1], and Witold Zieleznik[2]

[1] University Of Silesia, Faculty Of Computer Science And Materials Science, Institute Of Computer Science, Department Of Biomedical Computer Systems, Ul, Bedzinska 39, 41-200 Sosnowiec,

[2] Internist's Office, ul. Dworcowa 25, 41-902 Bytom, Poland

koprow@us.edu.pl, wrobel@us.edu.pl, wzieleznik@gmail.com

Abstract. The paper presents diagnostics of parenchyma echogenicity and organ dimensions in thyroid examinations in the case of Hashimoto's disease using image processing methods. In the event of discovering focal changes within the thyroid, a method for their pathology evaluation was suggested. The detector proposed operates fully automatically; using the information on the image texture it detects an artery in the image, which fulfils the role of reference point, and based on it - detects the area of interest.

1 Introduction

The tissue of regular thyroid features homogeneity and high echogenicity, which determines follicular structure of the gland [4]. In autoimmunological inflammation the follicles architecture destruction and lymphocytic infiltrations occur, what is the reason of echogenicity decrease [5]. Till 2000 the change in echogenicity was described as a subjective parameter, which was evaluated based on rough visual comparison with the surrounding muscular tissue of the neck (most frequently with the sternocleidomastoid muscle) [6], [7]. Now a computer histogram of grey scale is suggested for quantitative measurement of echogenicity decline in the thyroid in autoimmunological inflammations [7], [8], [9], [10], [12], [13]. This method excludes the subjective element in echogenicity evaluation, what has a substantial impact on the value and repeatability of ultrasonographic examination [11]. The lack of procedure standardisation is a significant drawback of this method, because individual authors were using various initial settings of the ultrasonograph, what affects gland's echogenicity. The control group in this study consisted of 10 volunteers without clinical symptoms, without a thyroid disease and illnesses of immunological basis in anamneses, with proper results of anti-TPO, anti-TG antibodies level and with proper TSH, FT3, FT4 levels as well as 10 volunteers with Hashimoto's disease. Thyroid examinations were performed (in both groups) within a week starting from subjective and objective examinations up to laboratory investigations. For each patient one image of the left and one of the right side of the disc was obtained (Fig. 1).

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 98–106, 2010.

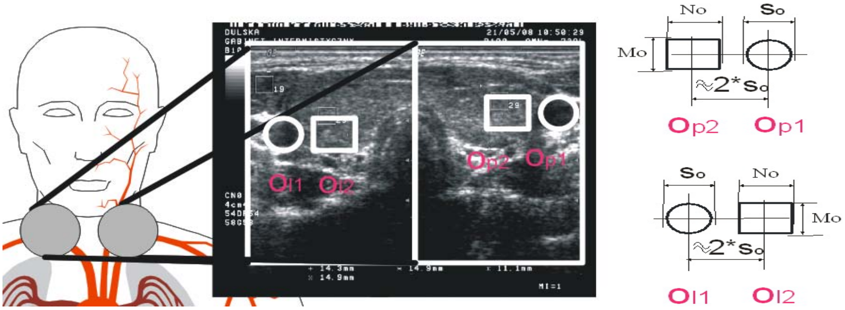


Fig. 1. Method of thyroid images obtaining and areas of o_{l1} and o_{p1} arteries, s_o in diameter, and also determined based on them, using the described algorithm, areas on thyroid lobes o_{l2} and o_{p2} of M_oxN_o dimensions

In the image presented in Fig. 1, areas o_{l1}, o_{l2}, o_{p1}, o_{p2} are marked, which will be automatically detected using the algorithm presented in this paper. Because of the image specific nature (Fig. 1), arteries on the left and right side will be initially detected, denoted as o_{l1} and o_{p1}, s_o in diameter and then, on their basis, the areas o_{l2}, o_{p2} on thyroid lobes, of M_oxN_o size.

From among known methods for texture analysis: statistical approach, structural method, transformation methods or model-based methods, a hybrid approach has been suggested, combining two aforementioned methods (statistical-structural).

2 Statistical-Structural Method

2.1 Image Preprocessing

Image $L(m, n)$ in grey levels, where m - line, n - column, of $MxN = 620x400$ resolution, is obtained from USG apparatus with a 9 MHz head. Then the filtering operation is carried out using a median filter of mask size $M_hxN_h = 3x3$ pixels [1], [2]. In the next stage the illumination unevenness is removed from the

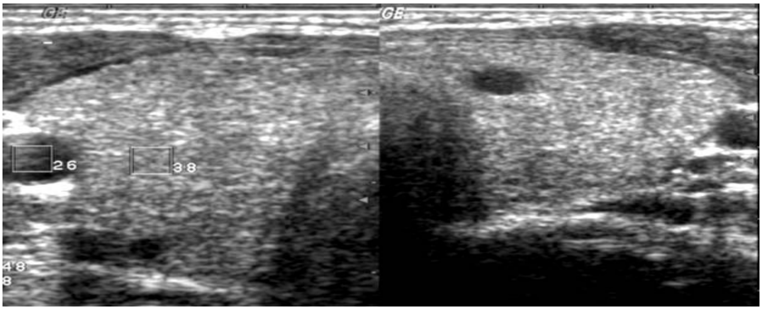


Fig. 2. Input image L_M

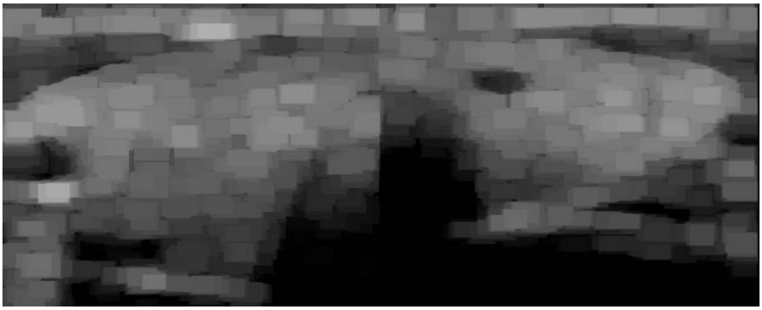


Fig. 3. Result L_0 of input image L_M opening

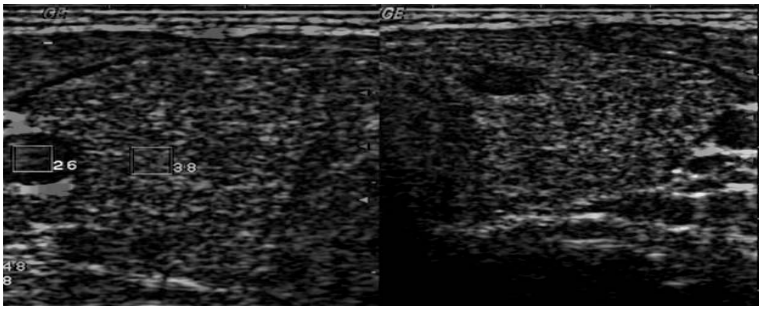


Fig. 4. Image L_T as normalised $L_M - L_0$ difference

image, what results in partial absorption and dispersion of ultrasonic beam by individual organs (objects).

To this end the operation of morphological opening was carried out (image created - L_o) with a structural element SE of $M_{SE}xN_{SE} = 19x19$ pixels size. Image L_T with removed background is computed as the difference $L_M - L_o$ normalised to $0 - 1$ interval. Image L_T created this way is fed to objects - texture detector input.

2.2 Suggested Detector Description

Image L_M is then subject to operations of erosion (Fig. 5), obtaining image $L_{e\alpha}$, using structural element $SE2$ of M_{SE2} x $N_{SE2} = 3x3k$ pixels size for $k = 1, 2, , 16, 17$. Each resultant image L_k for $k = 3, 6, 9, 12, 15, 17$ has been shown in the form of a colour contour in image L_M - Fig. 6.

Resultant image $L_{k=17}$ has been further used to determine circles by means of Hough's transform. Groups of pixels of L_b contour image of circle of radius $r_0 = 20, 21, 22, , 89, 90$ have been sought. The circles radii interval was assumed based on anthropometric and anatomic data of arteries cross-sections (Fig. 1). For example, for image from Fig. 7 the following coordinates of circles centres location of (m,n) pair and their radius r were obtained: Tab. 1.

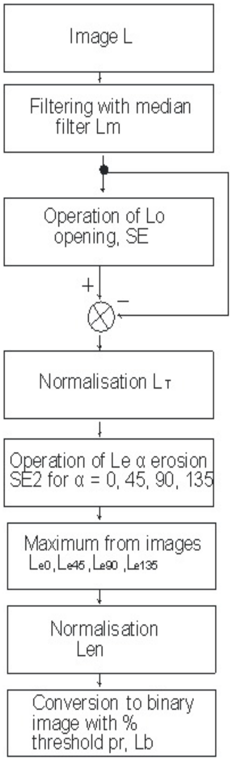


Image L
Filtering with median filter Lm
Operation of Lo opening, SE
+
−
Normalisation LT
Operation of Le α erosion SE2 for α = 0, 45, 90, 135
Maximum from images Le0, Le45, Le90, Le135
Normalisation Len
Conversion to binary image with % threshold pr, Lb

Fig. 5. Block diagram of suggested segmentation algorithm

When analysing results obtained from Table 1 and the location and number of circles visible in Fig.7, the following problems may be noticed:

- redundancy - too large number of circles found;

- determined circles not comprising the whole area of interest;

- determined circles comprising too large area of interest. These drawbacks have been eliminated using coefficients w_p, w_k, w_o defined as follows:

$$w_p(i) = \frac{\sum_{n=1}^{N}\sum_{m=1}^{M} L_w(m,n)}{2\pi r_o} \tag{1}$$

$$L_w(m,n) = \begin{cases} 1 \quad for & \sum_{n=1}^{N}\sum_{m=1}^{M} L_r(m,n) = 1 \wedge \\ & \wedge \sum_{n=1}^{N}\sum_{m=1}^{M} (L_p(m,n) \oplus SE_p) = 1 \\ 0 \quad other & \end{cases} \tag{2}$$

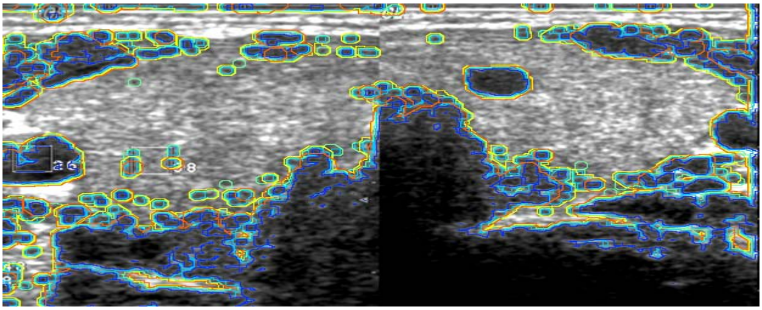


Fig. 6. Image L_M with colour contours of resultant binary images Lb for 30% threshold and $SE2$ at $k = 0, 45, 90$, and 135^o

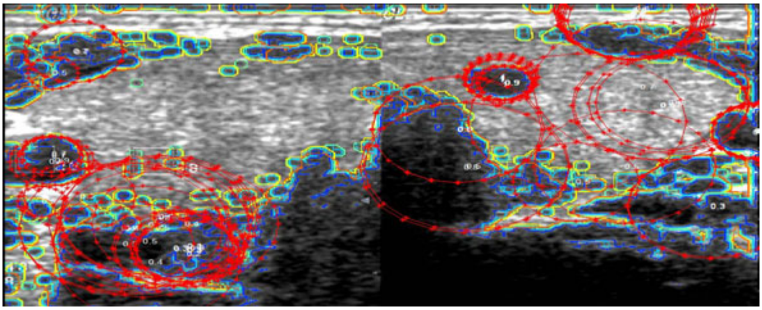


Fig. 7. Image L_M with marked red circles determined using Hough's transform

$$w_k(i) = \frac{\sum\limits_{n=1}^{N}\sum\limits_{m=1}^{M}[L_b(m,n)(L_r(m,n) \bullet SE_r)]}{\sum\limits_{n=1}^{N}\sum\limits_{m=1}^{M}(L_r(m,n) \bullet SE_r)} \tag{3}$$

$$w_o(i) = \frac{\sum\limits_{n=1}^{N}\sum\limits_{m=1}^{M}[(L_b(m,n)(L_r(m,n) \oplus SE_o))(1 - L_r(m,n) \bullet SE_r)]}{\sum\limits_{n=1}^{N}\sum\limits_{m=1}^{M}[(L_r(m,n) \oplus SE_o)(1 - L_r(m,n) \bullet SE_r)]} \tag{4}$$

where:

$\oplus$ - Minkowski summation,

$\bullet$ - Minkowski closure,

SE_r - mask of $M_r x N_r = (2r_0 + 1)x(2r_0 + 1)$ size, and:

$$L_p(m,n) = xor(L_b(m,n), L_b(m,n) \oplus SE) \tag{5}$$

Individual coefficients fulfil the following role:

w_p - is a relative measure of the number of circle points of coordinates (m_o, n_o) and radius r_0 coinciding in the SE_p mask size interval with the edge of the detected area,

w_k - is a relative measure of percentage number of image L_b objects points situated within the analysed circle,

w_o - is a relative measure of percentage number of image L_b objects points situated outside the analysed circle within the radius determined by mask SE_o.

For so defined coefficients: w_p, w_k, w_o and having analysed individual variation in a few dozen USG images, the following values have been arbitrarily set $w_p > 0.5$, $w_k < 0.2$, $w_o > 0.5$. When these assumptions are satisfied ($w_p > 0.5$, $w_k < 0.2$, $w_o > 0.5$) the number of wrongly recognised circles substantially declines and only those, which are significant from the diagnostic point of view, remain (Fig. 8). For example, in Table 1 this condition is fulfilled by the circle of number $L_p = 4$ (marked grey).

For 20 analysed patients no wrongly recognised areas have been found, which contour could be roughly approximated to a circle.

Table 1. Part of the first nine elements of the table of circles centres location coordinates and their radii and weights (sizes are given in pixels)

Lp	n_o	m_o	r_o	w_p	w_k	w_o
1	31	262	20	0.82	0.63	0.47
2	37	209	33	0.72	0.18	0.93
3	39	199	24	0.70	0.14	0.65
4	39	202	26	0.66	0.13	0.74
5	40	92	21	0.72	0.29	0.48
6	42	208	32	0.70	0.21	0.88
7	57	63	41	0.64	0.42	0.74
8	58	64	40	0.64	0.41	0.74
9	62	255	46	0.52	0.61	0.47

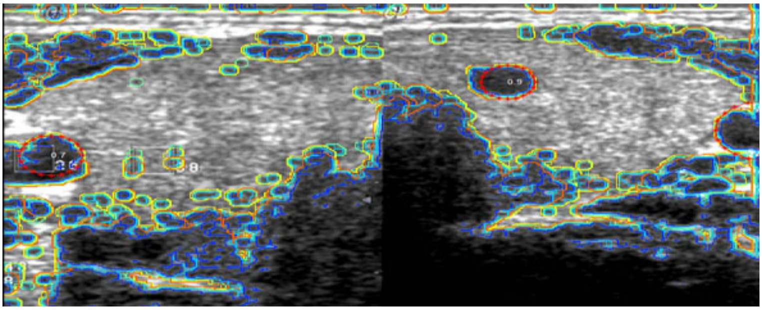


Fig. 8. Image L_M with marked red correct (at satisfied conditions $w_p > 0.5$, $w_k < 0.2$, $w_o > 0.5$) circles determined using Hough's transform

3 Suggested Method'S Use in Hashimoto's Disease Diagnostics

The reference analysis areas (arteries on both sides of the trachea - areas o_{l1} and o_{p1} - Fig. 1) found, fully automatically, enable acquiring measures, interesting from diagnostic point of view, of echogenicity changes (relative grey level) in thyroid lobes limited by areas $M_o x N_o$ (o_{l2} and o_{p2}). Sizes of areas o_{l2} and o_{p2}, i.e. $M_o x N_o$ have been determined based on trials carried out on the test group. The best results have been obtained for $M_o x N_o \simeq 40x40$ pixels [2], [3] (Fig. 1). At anatomically regular artery the size of this area is close to its diameter $M_o x N_o \simeq s_o x s_o$ and the displacement of centres of areas o_{l1} and o_{l2} as well as o_{p1} and o_{p2} is $\simeq 2 * so$. The obtained results of the average from differences in grey level between the areas o_{l1} and o_{l2} as well as o_{p1} and o_{p2} (Fig. 1) are presented in Fig. 9, Fig. 10, Fig. 11 for $k = 10$ patients from both groups (10 healthy patients and 10 patients with Hashimoto's disease). Values of differences o_{lp} in averages in areas o_{l1} and o_{l2} as well as o_{p1} and o_{p2} and of mean standard deviation std_{ol} and std_{op} have been calculated from the relationship:

$$\Delta_{olp} == \frac{\sum_{n=1}^{N_o}\sum_{m=1}^{M_o} o_{l1}(m,n) + \sum_{n=1}^{N_o}\sum_{m=1}^{M_o} o_{p1}(m,n)}{2M_oN_o} - \tag{6}$$

$$- \frac{\sum_{n=1}^{N_o}\sum_{m=1}^{M_o} o_{l2}(m,n) + \sum_{n=1}^{N_o}\sum_{m=1}^{M_o} o_{p2}(m,n)}{2M_oN_o} \tag{7}$$

$$std_{ol} = \sqrt{\frac{\sum_{n=1}^{N_o}\sum_{m=1}^{M_o}\left(o_{l2}(m,n) - \frac{1}{M_oN_o}\sum_{n=1}^{N_o}\sum_{m=1}^{M_o} o_{l2}(m,n)\right)}{M_oN_o - 1}} \tag{8}$$

$$std_{olp} = max\left(std_{ol}, std_{op}\right) \tag{9}$$

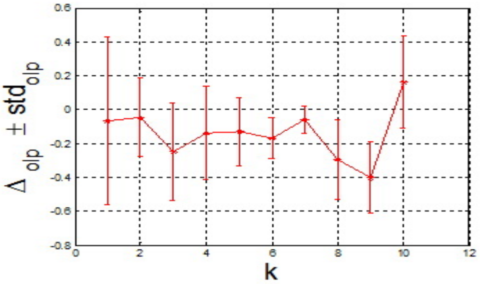


Fig. 9. Average difference between areas o_{l1} vs. o_{l2} and o_{p1} vs. o_{p2} for $k = 10$ healthy patients

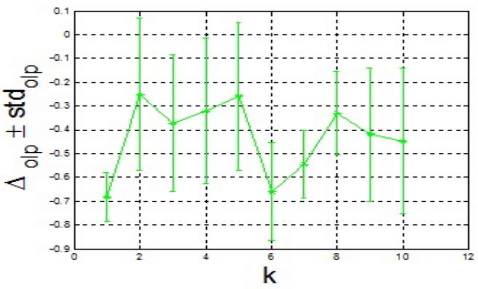


Fig. 10. Average difference between areas o_{l1} vs. o_{l2} and o_{p1} vs. o_{p2} for $k = 10$ patients with Hashimoto's disease

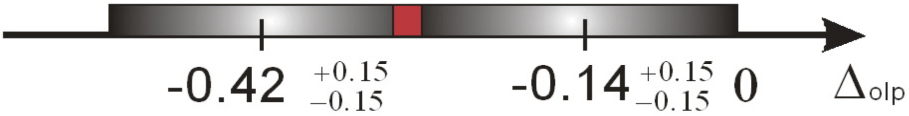


Fig. 11. Distribution on x axis of differences in grey level and their standard deviation of the average for healthy patients -0.14 ± 0.15 and for patients with Hashimoto's disease -0.42 ± 0.15

The obtained results are shown in Fig. 9 and Fig. 10.

As it results from the results presented above, in the case of inflammation in Hashimoto disease average values of grey level difference fall within the range $(\Delta_{olp} \pm std_{olp}) -0.42 \pm 0.15$ for patients with Hashimoto's disease and -0.14 ± 0.15 for healthy patients - Fig. 11. The area of uncertainty (red colour) visible in Fig. 11, covering the range from $-0.14 - 0.15 = -0.29$ to $-0.42 + 0.15 = -0.27$ of the difference in pixels brightness average value is the subject of further research.

4 Summary

The algorithm presented in the paper enables fully automatic determination of interest areas o_{l1} and o_{p1} and based on them o_{l2} and o_{p2} enabling computation of grey level average values and other necessary statistics. In the analysis and comparison of healthy patients with patients with Hashimoto's disease it turned out that the difference in average grey levels between o_{l1} and o_{p1} and o_{l2} and o_{p2}, respectively, is characteristic. In addition, it has been shown that the range of grey levels amounts to 0.42 ± 0.15 for patients with Hashimoto's disease and -0.14 ± 0.15 for healthy patients. The algorithm presented is a fundamental algorithm for automatic finding of arteries in USG images and may be successfully used for other automatic computations of areas correlated by morphometric and/or anthropometric dimensions.

References

1. Gutekunst, R., Hafermann, W., Mansky, T., Scriba, P.C.: Ultrasonography related to clinical and laboratory findings in lymphocyticthyroiditis. Acta Endocrinologica 121, 129–135 (1989)
2. Koprowski, R., Wrobel, Z.: Automatic segmentation of biological cell structures based on conditional opening or closing. Machine Graphics Vision 14(3), 285–308 (2005)
3. Koprowski, R., Wrobel, Z.: The automatic measurement of a staining reaction level. Machine Graphics Vision 15(2), 227–238 (2006)
4. Koprowski, R., Wróbel, Z.: Analysis of properties of automatic analysis algorithm for areas in USG image in Hashimoto's disease. Paper Submitted to ISSPA 2010 (2010)
5. Muller, H.W., Schroder, S., Schneider, C., Seiffert, G.: Sonographic tissue characterisation in thyroid gland diagnosis. A correlation between sonography and histology. Klinische Wochenschrift 63, 706–710 (1985)
6. Marcocci, C., Vitti, P., Cetani, F., Catalano, F., Concetti, R., Pinchera, A.: Thyroid ultrasonography helps to identify patients with diffuse lymphocytic thyroiditis who are prone to develop hypothyroidism. Journal of Clinical Endocrinology and Metabolism 72, 209–213 (1991)
7. Pedersen, O.M., Aardal, N.P., Larssen, T.B., Varhaug, J.E., Myking, O., Vik-Mo, H.: The value of ultrasonography in predicting autoimmune thyroid disease. Thyroid 10, 251–259 (2000)
8. Schiemann, U., Avenhaus, W., Konturek, J.W., Gellner, R., Hengst, K., Gross, M.: Relationship of clinical features and laboratory parameters to thyroid echogenicity measured by standardized grey scale ultrasonography in patients with Hashimoto's thyroiditis. Med. Sci. Monit. 9(4) (2003)
9. Schiemann, U., Avenhaus, W., Konturek, J.W., Gellner, R., Hengst, K., Gross, M.: Standardized grey scale ultrasonography in Graves' disease: correlation to autoimmune activity. European Journal of Endocrinology 141, 332–336 (1999)
10. Szczeklik, A.: Choroby Wewnêtrzne (Internal Diseases), WNT, Kraków (2006)
11. Wrobel, Z., Koprowski, R.: Praktyka przetwarzania obrazów z zadaniami w programie Matlab (Practice of Image in the Matlab Software), Wyd. Exit, Warszawa (2004)
12. Vitti, P., Rago, T., Mancusi, F., Pallini, S., Tonacchera, M., Santini, F., Chiovato, L., Marcocci, C., Pinchera, A.: Thyroid hypoechogenic pattern at ultrasonography as a tool for predicting recurrence of hyperthyroidism after medical treatment in patients with Graves' disease. Acta Endocrinologica 126, 128–131 (1992)
13. Vitti, P., Lampis, M., Piga, M., Loviselli, A., Brogioni, S., Rago, T., Pinchera, A., Martino, E.: Diagnostic usefulness of thyroid ultrasonography in atrophic thyroiditis. Journal of Clinical Ultrasound 22 (1994)

Estimating Quality Bounds of JPEG 2000 Compressed Leukocytes Images

Alexander Falcón-Ruiz[1], Juan Paz-Viera[1], and Hichem Sahli[2]

[1] Center for Studies on Electronics and Information Technologies, Universidad Central de Las Villas, Carretera a Camajuaní km 5 ½, Santa Clara, VC, CP58430, Cuba
{afalcon,jpaz}@uclv.edu.cu
[2] Vrije Universiteit Brussel, Dept. Electronics & Informatics, VUB-ETRO,
B-1050 Brussels, Belgium,
sahli@etro.vub.ac.be

Abstract. Several pathologies are detected by counting different types of leukocytes indigital microscopic images. However, manipulation of these images, i.e. storage and/or transmission, can be complicated by the large sizes of the files containing them. In order to tackle this particular situation, *lossy* compression *codecs*such as JPEG2000 have been employed while preserving the overall perceived image quality. In this paper a strategy based on objective quality metrics and performance of segmentation algorithms is proposed for the estimation of the maximal allowable compression rate (CR) where deterioration introduced in the images by the JPEG 2000 codec does not affect identification of white blood cells. Results indicate that the estimated value lays around CR = 142:1as measured by the metrics employed.

Keywords: JPEG 2000, microscopic images, leukocytes, compression, segmentation.

1 Introduction

Several pathologies such as acquired immunodeficiency syndrome,cancers, or chronic infections,are detected nowadays as a specialistobserves and extracts information from images containingwhite blood cells,also called leukocytes.Traditionally the expert select an area of interest in a peripheral blood or bone marrow slide, and by using a microscope, detectsdifferent types of leukocytes, increasing the counts for each one, providing important information to doctors in the diagnosis of such diseases.

Microscope-based biomedical imaging technique is characterized by large file sizes due to the bit depths employed and the high resolution properties of the digital acquisition devices. Some issues might arise when manipulating these images, i.e. during storage of everyday image production and/or transmission through digital communication networks [1], [2]. The amount of such images obtained in everyday practice, depending on the type of studies required for every particular detection task, can be enormous.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 107–114, 2010.

Although diagnosis is not recommended over compressed images,there has been an effort to employ *lossycodecs*for images that undergo second evaluation as part of a second opinion or a follow-upprocess. These algorithms are reported to have an order of magnitude higher in compression rate (CR) in comparison to *losslesscodecs*.

One ofthese*codecs* is JPEG 2000 (ISO 15444-1), based on the wavelet transform and added to DICOM standard around November 2001, in Supplement 61: JPEG 2000 Transfer Syntaxes[3], [4].

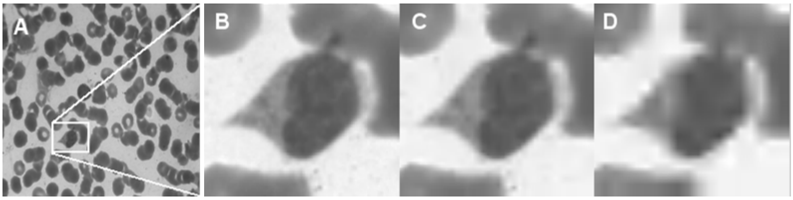


Fig. 1. Section Ashows a 1536V x 2048H pixel size bitmap image which occupies 9.00 MB of disk space. Sections B, C and D showa region containing a monocyte extracted from A after it is compressed at 3 different JPEG 2000 CRs, i.e. 50:1, 100:1 and 200:1. The edges, texture and contrast are severe distorted as compression rate increases.

Although JPEG2000 has been adopted by DICOM standard, there are still no regulations for the use of its*lossy*mode where, the higher the CRs are, the more distortion is introduced in the image, affecting particularly edge definition and so jeopardizing the correct identification of the structures and the diagnosis made through these images [5]. The example in Fig. 1 shows a typical image and a Region of Interest (ROI) extracted from this imageafter compression at different CRs.

Several researches have been carried out in order to establish a CRlimit for specific image types where the overall perceived image quality is not perceptually affected when using *lossycodec* [6], [7], [8]. In this paper, we propose estimating the maximum allowable CR where deterioration introduced, by the codec, in the images does not affect the quality of leukocytes images. The estimation is based on objective quality metrics, and its performance is evaluated using several segmentation algorithms.

2 Materials and Methods

2.1 The Images

Images were acquired using a Micrometrics 318CU CMOS digital camera, resulting in 24-bit color pictures of 2048H x 1536V size. The camera was attached to an Accu-scope 3016PL trinocular microscope with 100x oil immersion objective and 10x eyepieces. For the test, we selected 15 images per leukocyte class, where theclasses of interest were: lymphocytes, monocytes, neutrophils, basophils and eosinophils. Some manually cropped images are shown in Fig.2.

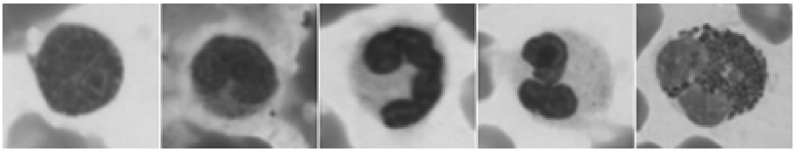


Fig. 2. Leukocytes. Left to right: lymphocyte, monocyte, neutrophil, basophil, eosinophil

Leukocytes classification is characterized by the observation, detection and classification of details or singularities within those images. A specialized observer extracts features such as shape, texture and color from them in order to classify the cells according to the types mentioned above. The *lossy*compression of such images might introduce distortions that directly affect the way in which those details are perceived as CR increases. Their preservation is crucial for assuring correct classification of leukocytes.

2.2 Compression with JPEG 2000 Codec

The implementation of JPEG2000 known as *JasPer*[9]was employed.Each entire image (as in Fig. 1 A) was compressed in a wide range of CR values from 33:1, where images show little degradation in quality, up to 1000:1, where image quality is highly degraded, the deterioration observed in the images is significant, cells loose important information such as edge definition, and contrast is also affected.Using a compression factor (CF=1/CR) step of 0.001, a set of 30 compressed images (CF from 0.001 to 0.030) was produced. Later on, the ROIs were extracted from the uncompressed and every reconstructed image.

The CR is calculated as the necessary memory space (in bytes) for allocating uncompressed image divided by the number of bytes necessary for allocating the same image in its compressed format.

2.3 Quantitative Measures

Traditionally, the overall estimation of image quality has been carried out with the calculation of several objective uni-variate and bi-variate metrics, altogether with subjective criteria involving human observers. Their reliability in different situations and image types has been also widely investigated by many authors [1], [2], [10].

For our particular research the following bi-variate measures are chosen:

- The Peak Signal-Noise Ratio (*PSNR*):considering *X(i,j)*as the uncompressed image and *Y(i,j)*the restored one, *PSNR*is defined as:

$$PSNR(dB) = 10 \cdot \log_{10}\left(\frac{MAXp^2}{MSE}\right), \quad (1)$$

where$MAXp=2^B-1$,B is the image bitdepth and *MSE*(mean square error) is defined as:

$$MSE = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} (X(i,j) - Y(i,j))^2 \ , \qquad (2)$$

where*m* and *n* are the number of rows and columns in the image, respectively.

- The spectral distance (*SD*):a measure of distance between uncompressed and reconstructed Fourier domainimages given by:

$$SD = \frac{1}{m \cdot n} \sum_{i=1}^{m} \sum_{j=1}^{n} (|\varphi(i,j)| - |\hat{\varphi}(i,j)|)^2, \qquad (3)$$

where$\varphi(i,j)$and$\hat{\varphi}(i,j)$are the imaginary parts ofFourier transforms of uncompressed and restored images, respectively.

- The gain in Contrast to Noise ratio(*gCNR*) is defined as:

$$gCNR(dB) = 10 \cdot \log_{10}\left(\frac{CNR_X}{CNR_Y}\right), \qquad (4)$$

whereCNR_X and CNR_Y are the contrast-to-noise ratios in the uncompressed and reconstructed images respectively calculated as $CNR_i = (\bar{X}_{i2} - \bar{X}_{i1})/\sigma_i$, with $\bar{X}_{i1}$ and $\bar{X}_{i2}$ being the mean values of intensity from two different regionsin image *i* and σ_i the standard deviation of noise in same image.

- The structural similarity index (*MSSIM*):a powerful measure proposed by Wang *et al.* [10] was also employed. It can be calculated as:

$$MSSIM(X,Y) = \frac{1}{M} \sum_{i=1}^{M} SSIM(x_{i,}y_{i,}), \qquad (5)$$

where*M*is the number of image blocks $x_i$and $y_i$of uncompressed and reconstructed image respectively and *SSIM* calculated as:

$$SSIM(X,Y) = \frac{(2\mu_X\mu_Y + C_1)(2\tau_{XY} + C_2)}{(\mu_X^2 + \mu_Y^2 + C_1)(\tau_X^2 + \tau_Y^2 + C_2)}, \qquad (6)$$

where$\mu_x$and$\mu_y$are the luminance values, $\tau_X$and $\tau_Y$the contrast estimation values for uncompressed and reconstructed image respectively and $\tau_{XY} = \frac{1}{N-1} \sum_{i=1}^{N} (x_i - \mu_x)(y_i - \mu_y)$. The constants C_1 and $C_2$are placed to avoid instability:$C_i=(K_iL)^2$ where L= 255, for 8bpp images and K_i<< 1.

All bi-variate calculations are made between the uncompressed image and every reconstructed image after being compressed at each CR value in the interval studied.

2.4 The Segmentation Algorithms

Typically, leukocytes identification is based on visual inspection of individual images of wider Field Of View than the size of individual cells and containing other structures as well as noise and/or artifacts. The approach of having experts dedicated to this task is time consuming, exhausting and prone to human error, requiring frequent repetitions to validate results [11].These situations, altogether with the great amount of images necessary to achieve a diagnosis, encourage scientists to develop segmentation algorithms as an early step for automated classification.

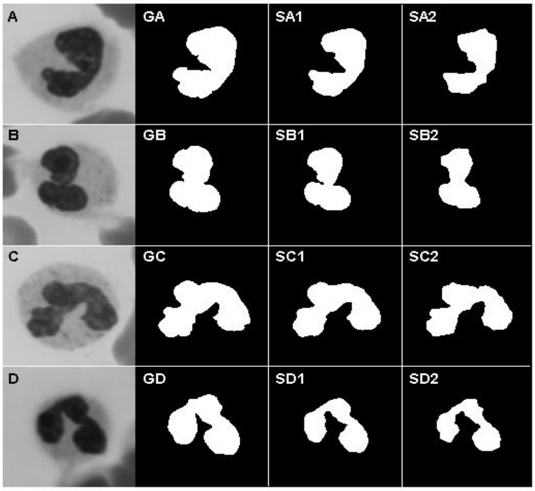


Fig. 3. Images (A-D) in first column are green component sections containing leukocytes extracted from bigger images as in Fig. 1 section A. Second column (GA - GD) contains the manually extracted Ground Truths for these sections, while columns three and four contains the segmentation results at CR=33:1 (SA1- SD1) and CR=1000:1 (SA2 – SD2).

These algorithms are conceived to analyze the images acting as secondary readers where they reanalyze the image after the initial diagnose by the physician. They are also designed to objectively quantify features in a robust, reliable and reproducible manner.

In the present paper, to assess the CR three automatic segmentation algorithms are tested over a set of leukocytes images, each one compressed at 30 different CR values within the interval 33:1 to 1000:1, i.e. Otsu's method [12], Active Contours (AC)method [13, 14] and the Mixture of Gaussians (MoG) method [15]. For assessing the segmentation results, of each of the proposed methods, applied at specific CR, theHausdorff distance [16], between Ground Truths (GTs) and segmentation results have been estimated.GTswere selected in each ROI at initial state, i.e. without compression. Fig. 3 shows some of the ROIs from the test images set, their GTs and segmentation results at minimum and maximum CR.

4 Results

Fig. 4 shows four rate-distortion curves for the four different quality metrics calculated over the ROIs within the CR interval investigated and averaged over the 15 images in the test set. From the graph, it is observed that metrics such as *PSNR* and *gCNR* show a stronger dependency with CR variation while *SD* and *MSSIM* show less dependence with CR.

A nick point is observed in the curves near CR=142:1 (CF=0.007). For CR values bigger that this, image quality is severely distorted. At this point *PSNR* is around 91

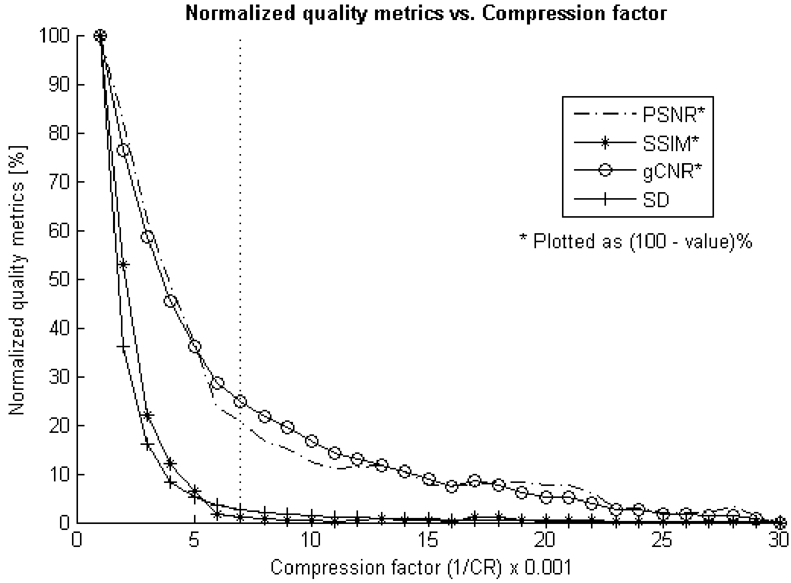


Fig. 4. The objective metrics are shown in a percent scale. Metrics such as *PSNR* and *gCNR* show a stronger dependency with variation in CR while metrics such as *SD* and *MSSIM* show less dependence with CR. The nick point in the curves at CR = 142:1 suggests a lower CR bound. For CRs bigger that this, image quality is severely distorted.

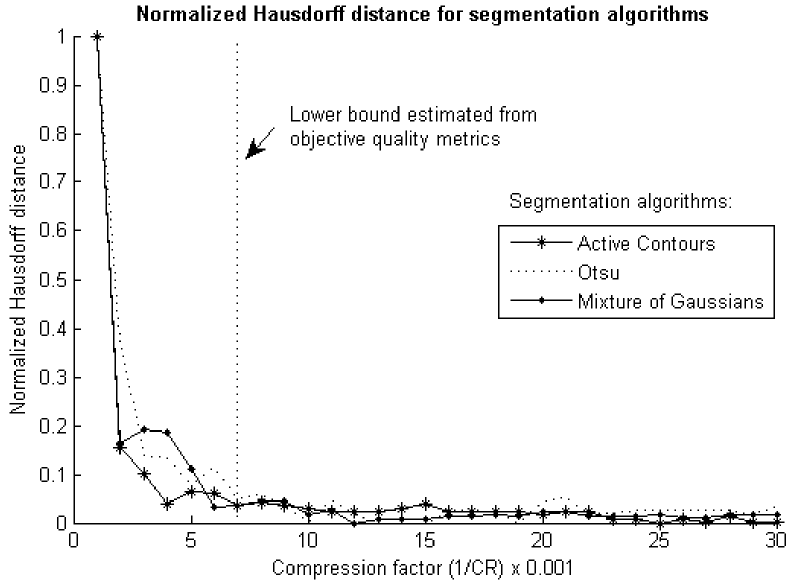


Fig. 5. Normalized Hausdorff distance for the three segmentation algorithms tested. Dotted line indicates the estimates lower bound in correspondence with previous results from objective quality metrics

dB, *gCNR*is around20dB, *SD* is 1.19 units and *MSSIM* is 0.98. At this CR, file size is reduced from 9 MB to approximately 65 KB.

Fig. 5 shows the normalized Hausdorff distances for the three segmentation algorithms tested. Although in this graph the three methods show similar behavior as quality metrics, Otsu's method had the best performance in our experiment with lower Hausdorff distance (HD) to the GT (at CR=33:1, HD_{Otsu} = 4.5, HD_{MoG} = 8.2, and HD_{AC} = 10.1 Hausdorff distance units). The Hausdorff distance for CRs below 142:1 has a standard deviation below 5% of the Hausdorff distance for the maximum CR tested.

5 Conclusions

The analysis with objective metrics suggestedan interval of CR values from 33:1 up to 142:1 where is *safe* to use JPEG 2000. This initial and partial result is later confirmed by the automatic segmentation algorithms tested which agrees in the upper most CR value of 142:1.

Both, metrics for evaluating objective quality distortions and the performance of segmentation algorithms, are considered representative for estimating quality degradation caused by the lossy codec.

The result presented are preliminary and lack of subjective experience in interpreting this type of images. A more complex investigation including subjective evaluation should be carried out in order to precise the bounds for lossy compression. Nevertheless, a CR limit of 142:1 was estimated through both metric types as a limit for using JPEG 2000 compression in leukocytes identification tasks.

Acknowledgements

The authors would like to thank Alberto Taboada-Crispí for the analysis and review, the Canadian International DevelopmentAgency Project Tier II-394-TT02-00 and the Flemish VLIR-UOS Programme for InstitutionalUniversity Cooperation (IUC) for partly supporting this investigation.

References

1. Acharya, T., Ray, A.K.: Image processing Principles and applications. John Wiley & Sons, Inc., Hoboken (2005)
2. Lau, C., et al.: Telemedicine.Handbook of Medical Imaging. In: Kim, Y., Horri, S. (eds.), vol. 3, pp. 305–331. SPIE, Bellingham (2000)
3. Clunie, D.A.: DICOM Supplement 61: JPEG 2000 Transfer Syntaxes (2002), ftp://medical.nema.org/medical/dicom/final/sup61_ft.pdf
4. Rabbani, M., Joshi, R.: An overview of the JPEG2000 still image compression standard. 1, Signal Processing: Image Communication 17, 3–48 (2002)
5. Foes, D.H., et al.: JPEG 2000 compression of medical imagery. In: SPIE Proc., San Diego, California, vol. 3980 (2002)

6. Penedo, M., Lado, M.J., Tahoces, P.G., Souto, M., Vidal, J.J.: Effects of JPEG2000 data compression on an automated system for detecting clustered microcalcifications in digital mammograms. IEEE Trans. on Information Technology in Biomedicine 10(2) (2006)
7. Zhang, Y., Pham, B., Eckstein, M.P.: Evaluation of JPEG2000 encoder options: human and model observer detection of variable signals in X-Ray coronary angiograms. IEEE Trans. on Med. Imagin. 23(5) (2004)
8. Paz, J., Pérez, M., Schelkens, P., Rodríguez, J.: Impact of JPEG 2000 Compression on Lesion Detection in MR Imaging. Journal of Medical Physics 36(11), 4967–4976 (2009)
9. Adams, M., Kossentini, F.: JasPer: a software based JPEG2000 codec implementation. In: Proc. of IEEE International Conference on Image Processing, Vancouver, British Columbia, Canada. Institute of Electrical and Electronics Engineers, vol. 2, pp. 53–56 (2002)
10. Wang, Z., Bovik, A.C., Sheikh, H.R., Simoncelli, E.P.: Image Quality Assessment: FromError Visibility to Structural Similarity. IEEE Trans. on Image Proc. 13(4) (2004)
11. Lee, J.K.T.: Interpretation accuracy and pertinence. American College of Radiology 4 (2002)
12. Otsu, N.: A Threshold Selection Method from Gray-Level Histograms. IEEE Transactions on Systems, Man and Cybernetics 9(1), 62–66 (1979)
13. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International Journal of Computer Vision 1(4), 321–331 (1988)
14. Chan, F.C., Vese, L.A.: Active Contours Without Edges. IEEE Transactions on Image Processing 10(2), 266–277 (2001)
15. Gupta, L., Sortrakul, T.: A gaussian-mixture-based image segmentation algorithm. Pattern Recognition 31(3), 315–325 (1998)
16. Huttenlocher, D., Klanderman, G.A., Rucklidge, W.J.: Comparing Images Using the Hausdorff Distance. IEEE Transactions on Pattern Analysis and Machine Intelligence 15(9), 850–863 (1993)

Surface Material Segmentation Using Polarisation

Nitya Subramaniam and Edwin Hancock

Department of Computer Science, University of York, YO10 5DD, United Kingdom
{nitya,erh}@cs.york.ac.uk

Abstract. This paper describes the use of polarisation information for surface segmentation based on material characteristics. We work with both polarised and unpolarised light, and hence domains where the polarisation is either specular or diffuse. We commence by using moments to estimate the components of the polarisation image (mean-intensity, polarisation degree and phase) from images obtained through multiple polariser orientations. From the Fresnel theory, the phase of light remitted from a surface is equal to the azimuth angle of the remitted direction, and for materials with restricted ranges of refractive index the polarisation degree determines the zenith angle. Based on this observation, we parameterise the angular distribution of the mean intensity for remitted light using spherical harmonics. We explore how vectors of spherical harmonics can be used to characterise varying surface reflectance distributions, and segment a scene into different material patches using Mahalanobis distances and normalized graph cuts.

1 Introduction

The polarisation of light is used by some animals (e.g. mantis shrimp) to augment the information contained in the visible spectrum. Although humans are insensitive to polarisation, it is a useful addition to colour and intensity information for computer vision applications [16]. Polarisation imaging has been used to develop a variety of techniques in computer vision, including surface quality inspection [15],[8],[10], shape recovery [1],[2],[9],[11] and material characterisation [17],[7]. Polarisation can also be used to infer information concerning the reflectance properties of surfaces. For instance, Atkinson and Hancock have shown in [3] how diffuse polarisation can be used to estimate the birectional reflectance function. However, their method is computationally demanding, using simulated annealing to estimate the BRDF. In this paper we take a simpler view of the problem. For fixed light source direction and approximately planar samples, provided that the range of refractive indices for different materials in a scene is limited, the polaristion image allows the angular distribution of reflected or remitted light to be estimated.

Here we exploit this property and parameterise the distribution using spherical harmonics. Vectors of harmonic coefficients are then used to characterise the reflectance distribution on a pixel-by-pixel basis. We can then segment a scene into regions of different reflectance properties using the coefficient vectors. Here we compute the difference in reflectance characteristics using the Mahalanobis distance between coefficient vectors and then use normalised cuts [14] to segment the scene into regions of different material composition.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 115–124, 2010.

The Fresnel theory of light (see [6],[4]) is a quantitative description of reflection and refraction at a smooth boundary between two media. The analysis is relatively straightforward for dielectrics, but the situation is less tractable for metals due to the induction of surface currents by the time varying electromagnetic field of light. In dielectrics, polarisation in remitted light may arise in two different ways. Specular polarisation arises when polarised incident light is reflected from the object surface in the specular direction. In the case of diffuse polarisation, initially unpolarised light is refracted into the surface and the remitted light acquires a spontaneous polarisation due to refraction at the surface.

In both cases the zenith angle of the reflected or remitted light is determined by the degree of polarisation and the azimuth angle determines the polarisation phase angle.

2 Polarisation Image

When scattered light is measured through a linear polarising filter, the intensity changes as a sinusoidal function of the polariser angle α_p and the transmitted radiance sinusoid (TRS) is given by

$$I(\alpha_p) = \frac{(I_{max} + I_{min})}{2} + \frac{(I_{max} - I_{min})}{2} cos(2\alpha_p - 2\phi) \tag{1}$$

where I_{max} is the maximum brightness, I_{min} the minimum brightness and ϕ the phase angle. It is more convenient to write the above formula in terms of the mean-intensity

$$\hat{I} = \frac{1}{2}(I_{max} + I_{min}) \tag{2}$$

and the degree of polarisation ρ, giving

$$I(\alpha_p) = \hat{I}(1 + \rho \cos(2\alpha_p - 2\phi)) \tag{3}$$

Suppose that we take N equally spaced polarisation images, so that the polariser angle index is $p = 1, 2, ..., N$. Let

$$x_p = (I(\alpha_p) - \hat{I})/\hat{I}, \tag{4}$$

$$\hat{x} = \frac{1}{N}\sum_{p=1}^{N} x_p \tag{5}$$

and

$$\sigma^2 = \frac{1}{N}\sum_{p=1}^{N}(x_p - \hat{x})^2 \tag{6}$$

The moments estimators of the three components of the polarisation image are the mean intensity

$$\hat{I} = \frac{1}{N}\sum_{p=1}^{N} I(\alpha_p) \tag{7}$$

the polarisation degree

$$\rho = \sqrt{2/\pi}\sigma \tag{8}$$

and the phase angle

$$\phi = \frac{1}{2}\cos^{-1}(\langle \hat{x}\cos(2\alpha)\rangle/\pi\rho) \tag{9}$$

We use a moment-based method along with least squares fitting to estimate the degree and phase of polarisation in the scattered light. To improve the robustness of our calculation, observations with large deviations (more than 25% of the TRS amplitude) are not used in the estimation.

From the Fresnel theory it is straightforward to show that the azimuth angle for reflected polarised light or remitted diffusely polarised light is equal to the phase angle ϕ [17]. The zenith angle θ depends on whether the polarisation is specular or diffuse. For diffuse polarisation, the polarisation degree is given by

$$\rho_d = \frac{(n-1/n)^2\sin^2\theta}{2-2n^2-(n+1/n)^2\sin^2\theta+4\cos\theta\sqrt{n^2-\sin^2\theta}} \tag{10}$$

while the degree of specular polarisation ρ_d is given as:

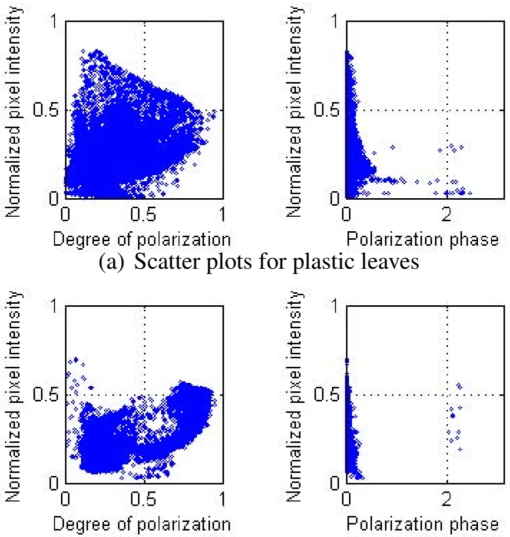
$$\rho_s = \frac{2\sin^2\theta\cos\theta\sqrt{n^2-\sin^2\theta}}{n^2-\sin^2\theta-n^2\sin^2\theta+2\sin^4\theta} \tag{11}$$

where n is the refractive index.

Here we aim to use (1) through (11) to analyse the distribution of relectance from approximately planar samples of different material. Provided we know whether we are measuring the specular polarisation of reflected polarised light, or the diffuse polarisation of remitted initially unpolarised light, then θ and ϕ are the zenith and azimuth angles of light with respect to the surface normal. To do this we assume the range of refractive index is small, and can be treated as a constant. In our experiments we work with the value $n = 1.45$, which is typical of a wide range of dielectrics.

3 Reflectance Distributions

The observation underpinning this paper is that under the restrictions of local surface planarity and slowly varying refractive index, the polarisation image allows us to measure the distribution of mean intensity $\hat{I}$ with the zenith and azimuth angle of remitted light, θ and ϕ. To provide some illustrative motivation, Fig.1 shows a scatter plot of the intensity versus the degree of polarisation and surface azimuth angle for real and plastic leaves. The leaves are approximately planar, and the angle of incidence is approximately 15 degrees. Here we work with initially unpolarised light and use the formula for diffuse reflectance in (10) to estimate the zenith angle from the measured polarisation. There are a number of features to note from the plot. First, the distributions are quite different for the two materials. We attribute this to the fact that natural leaves have a layered sub-surface structure, which affects distribution of remitted light through subsurface refraction according to Snell's law. Artifical leaves do not exhibit such structure. Second, when the refractive index is changed within the known range for dielectrics, there is a small shift in the plots at all zenith angles. Since the shift is uniform, the effect of approximating refractive index in the feature calculations can be neglected.



(a) Scatter plots for plastic leaves

(b) Scatter plots for real leaves

Fig. 1. Scatter plots: The variations in pixel intensity plotted against ρ and ϕ

3.1 Calculating Spherical Harmonic Features

Our idea is to parameterise the distribution of mean intensity as function of the azimuth and zenith angles. The polarisation image consists of a set of triples

$$P = \{(\hat{I}_i, \rho_i, \phi_i), i = 1, ..., M\}$$

from which we compute the set

$$D = \{(\hat{I}_i, \theta_i, \phi_i), i = 1, ..., M\}$$

using the expression for diffuse polarisation in terms of zenith angle in (10). The distribution of mean image intensity at each pixel is expressed as a function of azimuth and zenith angles. Any such spherically symmetric function $f(\theta, \phi)$ can then be expressed as a weighted sum of the orthonormal basis functions Y_l^m (called the spherical harmonics of degree l and order m) as follows:

$$f(\theta, \phi) = \sum_{l=1}^{\infty} \sum_{m=-l}^{l} a_{l,m} Y_l^m(\theta, \phi), a \in \mathbb{R} \tag{12}$$

where $Y_l^m(\theta, \phi)$ is a function of the associated Legendre polynomials $P_l^m(z)$ with $z = \cos\theta$, given by

$$Y_l^m(\theta, \phi) = \sqrt{\frac{2l+1}{4\pi} \frac{(l-m)!}{(l+m)!}} P_l^m(\cos\theta) e^{im\phi}. \tag{13}$$

Using the orthonormality properties of the spherical harmonics, the coefficients are given by

$$a_{l,m} = \int_0^{2\pi}\int_0^{\pi} f(\theta,\phi) Y_l^m(\theta,\phi) \sin\theta \, \mathrm{d}\theta \, \mathrm{d}\phi \tag{14}$$

From (14), we obtain the following moments estimators of the spherical harmonic coefficients of the mean-intensity distribution.

$$a_{l,m} = \frac{1}{M}\sum_{i=1}^{M} \hat{I}_i Y_l^m(\theta_i,\phi_i) \tag{15}$$

In practice we estimate the set of coefficients over non-overlapping 10x10 blocks of pixels, and truncate the spherical harmonic expansion at $l = 8$ with m varying from $-l$ to l. As a result the mean intensity distribution is each pixel block is parameterised by an 81 element vector of spherical harmonic coefficients

$$A = [a_{0,0}, a_{1,-1}, a_{1,0}, a_{1,1}, ..., a_{8,8}]^T.$$

Estimation of harmonic functions in previous literature includes residual fitting approaches by [13] and [5] and spherical FFT by [12]. We use a MATLAB function to compute the Legendre polynomials and a moments based approach to estimate the coefficients $a_{l,m}$. We divide the image into windows and calculate the average coefficients over each window. The window size is chosen to ensure that the instensity function is a reasonable representation of shape while taking care to not over-smooth the features.

3.2 Distribution of Information in the Feature Vector

We aim to use the coefficient vectors for both segmenting and classifying regions in scenes. To this end we commence by computing the variance matrix over blocks of the image. If the image blocks are indexed by $k = 1, ..., L$ and the k-th block has coefficient A_k, then the mean coefficient vector is

$$\hat{A} = \frac{1}{L}\sum_{k=1}^{L} A_k \tag{16}$$

and the covariance matrix is

$$\Sigma_A = \frac{1}{L}\sum_{k=1}^{L}(A_k - \hat{A})(A_k - \hat{A})^T \tag{17}$$

The Mahalanobis distance between the coefficient vectors for the blocks indexed k_1 and k_2 is

$$D_{k_1,k_2} = (A_{k_1} - A_{k_2})^T \Sigma_A^{-1} (A_{k_1} - A_{k_2}) \tag{18}$$

From the Mahalanobis distance we compute the $L \times L$ block affinity matrix S with elements

$$S(k_1,k_2) = \exp[-R D_{k_1,k_2}] \tag{19}$$

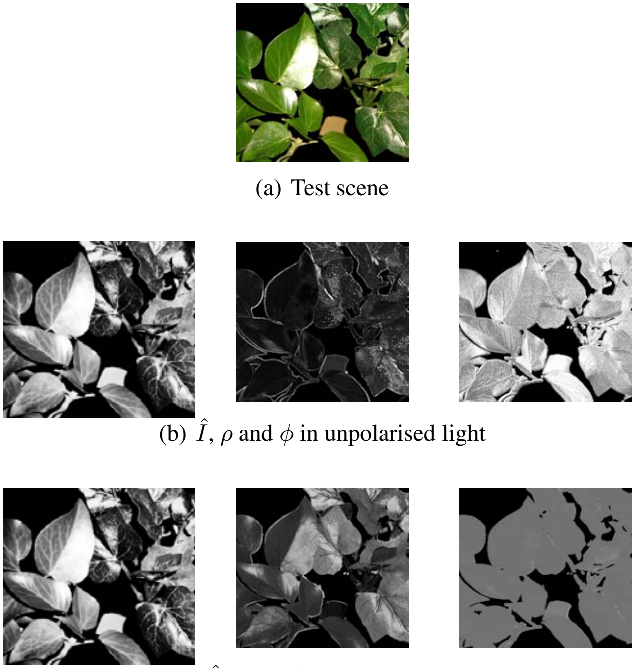
where R is a constant. We segment the polarisation image into regions by recursively applying Shi and Malik's [14] algorithm to the affinity matrix.

4 Experiments

The images are recorded in a darkened room with matte black walls and working surfaces. The studied objects and the camera are positioned on the same axis and a halogen light source (visible spectrum) is positioned at approximately 15 degrees from the viewing axis, to reduce specular reflection. Linear polarising filters are placed in front of the source and the camera. The camera polaroid is rotated through 180 degrees and images are captured with fixed aperture size and exposure time.

Objects studied include fruits, vegetables, natural leaves and plastic leaves. Wolff [16] suggests taking images at polariser orientations 0, 45 and 90, while Atkinson and Hancock [1] use 10 degree intervals of polariser orientations. We choose to record images at 30 degree intervals as a compromise between data collection time and resilience to noise. However with the availability of liquid polarisation cameras data collection time is now a trivial issue.

The polarisation degree captures edges and fine surface texture in unpolarised light and coarse features in polarised light. The polarisation phase captures more surface detail in unpolarised than in polarised light as demonstrated in Fig.2 for a mixed scene of artificial and natural leaves.



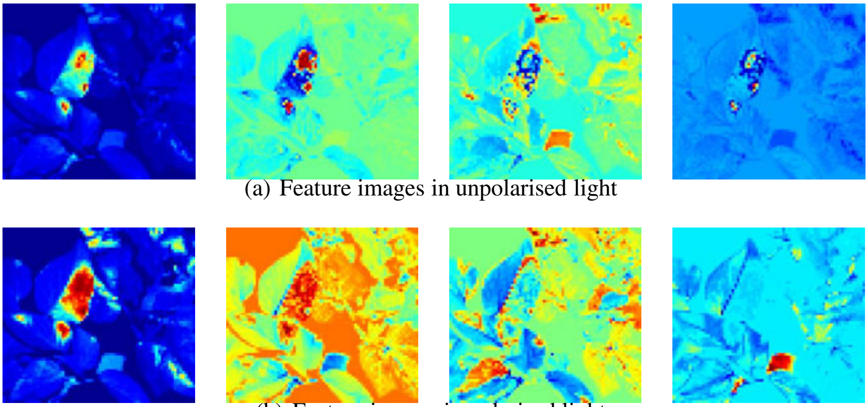
(a) Test scene

(b) $\hat{I}$, ρ and ϕ in unpolarised light

(c) $\hat{I}$, ρ and ϕ in polarised light

Fig. 2. Polarisation image: Grayscale images represent the values of (L-R) mean intensity $\hat{I}$, degree of polarisation ρ and polarisation phase ϕ computed for the scene in (a), using images captured in (b) unpolarised incident light and (c) polarised incident light

We have performed PCA on the coefficient vector covariance matrix Σ_A. The results are shown in Fig.3 which shows the first four principal components which accout for 95% of the variance. These four components are used to compute a block-by-block feature vector. The features emphasize the vascular structure of real leaves and are weaker in polarised light because the spontaneous polarisation of light on multiple scattering within the real leaf is harder to detect in strongly polarised light.



(a) Feature images in unpolarised light

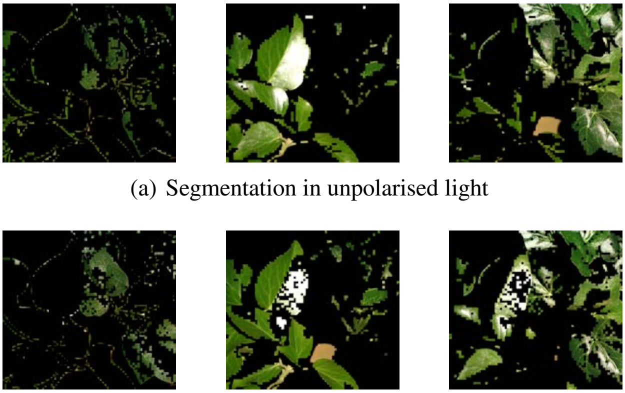
(b) Feature images in polarised light

Fig. 3. Feature images (L-R): The first four features for Fig.2(a) calculated from pca-mapped spherical harmonics coefficients

The results of segmenting the scene in Fig.2(a) using normalized graph cuts algorithm from [14] are shown in Fig.4. These results were obtained with 81 features from spherical harmonic coefficients up to degree 8 on a 660x720 image. The affinity matrix was computed using the Mahalanobis distances between the feature vectors in blocks of 10x10 pixels. Specularities cause some difficulty in correct segmentation when using polarised light. Segmentation is better in unpolarised light even in the presence of specularities due to stronger spontaneous polarisation degree and therefore stronger discrimination in coefficient features. The results shown represent a sample of the segmentation results obtained using the proposed method.

The method was also used to successfully segment images taken in natural sunlight. It is well known that light from the sun gets strongly polarised due to scattering by air molecules in the atmosphere. Thus direct sunlight is polarised light. However reflection from object surfaces tends to spontaneously depolarise light. Thus in outdoor settings, images taken in direct and diffuse light can be considered to be under polarised and unpolarised states of incident light. Figure 5(a) contains a section of plastic leaves imaged among the branches of shrub in diffuse sunlight and 5(c) captures camouflage in the midst of a hedge in direct sunlight.

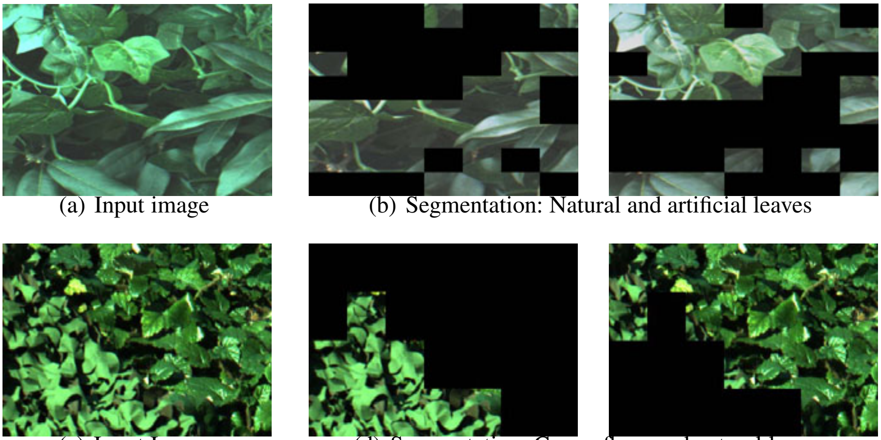
The outdoor scene in Figure 5(a) consists of an image 336×384 pixels in size and is divided into blocks of size 24×24 for spherical harmonic coefficient calculation. The harmonic expansion is truncated at order 30. Out of 112 blocks, 66 are classified correctly and 16 blocks contain both materials, giving a segmentation accuracy of 71%.



(a) Segmentation in unpolarised light

(b) Segmentation in polarised light

Fig. 4. Segmentation: The image in Fig.2(a) is segmented using normalized cuts into (L-R) background, natural leaves and plastic leaves



(a) Input image

(b) Segmentation: Natural and artificial leaves

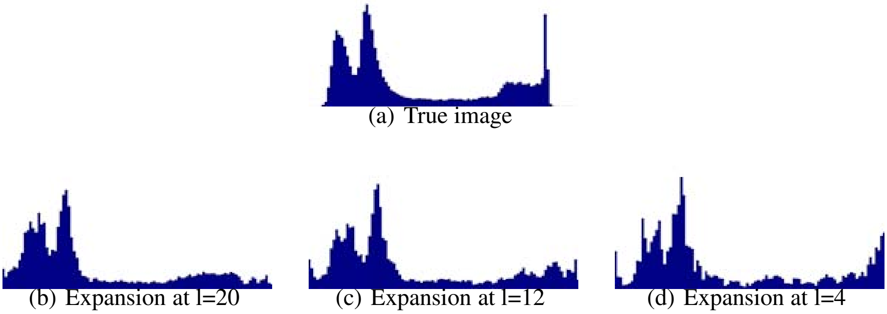
(c) Input Image

(d) Segmentation: Camouflage and natural leaves

Fig. 5. Outdoor scenes : Imaged scenes include (a) real and plastic leaves in diffused sunlight (c) natural leaves and camouflage in direct sunlight. (b) and (d) show the segmentation results using the proposed method.

For the 370×420 image in Figure 5(c), a block size of 66×66 for feature generation produced the best segmentation performance. Out of 28 blocks, 2 blocks are classified incorrectly, and 3 blocks contain sections of both materials. Assuming that all blocks containing more than one material are incorrectly classified gives an accuracy of 82% while a more optimistic estimation gives an accuracy of 93%.

Segmentation results were tested for different values of window size and orders of expansion. The window size that produces optimal segmentation is found to vary from scene to scene. It is noted that for the window size that produces optimal segmentation,



(a) True image

(b) Expansion at l=20

(c) Expansion at l=12

(d) Expansion at l=4

Fig. 6. Histograms: (a) shows an image histogram and (b) to (d) show histograms of reconstructed pixel intensities with exapnsions upto $l = 20, 12$ and 4

spherical harmonic expansion truncated at order 12 gives the desired segments and the results are stable over multiple runs. Adding higher order terms to the feature vector does not improve performance noticeably. This implies that most of the key feature information is contained in low order harmonics, up to order 12. For other window sizes however the normalized cuts seem to produce unpredictable results. Further investigation to analyze the factors that affect the accuracy of segmentation presents a topic for further research in this method.

The accuracy of our intensity function can be checked by reconstructing the intensity at each pixel as a spherical harmonic expansion using (12). The histograms in Fig.fig:histo compare the original histogram (a) with reconstructions (b) - (d) using coefficients of degree 20, 12 and 4, for a patch of the scene in Fig.2(a). The reconstruction error shows a drop with increase in degree of expansion, as expected. The resolution of reconstruction however is limited by the choice of window size in coefficient calcuation. The reconstructed images show a smoothing effect attributed to the truncation of spherical harmonic expansion and the windowing in calculation of coefficients.

References

1. Atkinson, G., Hancock, E.R.: Recovery of surface orientation from diffuse polarization. IEEE Transactions on image Processing 15(6), 1653–1664 (2006)
2. Atkinson, G., Hancock, E.R.: Polarization-based surface reconstruction via patch matching. Computer Analysis of Images and Patterns, 466–473 (2007)
3. Atkinson, G., Hancock, E.R.: Two-dimensional brdf estimation from polarisation. Computer Vision and Image Understanding 111(2), 126–141 (2008)
4. Born, M., Wolf, E.: Principles of Optics, 7th (expanded) edn. Cambridge University Press, Cambridge (1999)
5. Chung, M.K., Dalton, K.M., Davidson, R.J.: Tensor-based cortical surface morphometry via weighted spherical harmonic representation. IEEE Transactions on Medical Imaging 27(8), 1143–1151 (2008)
6. Hecht, E.: Optics, 4th edn. Addison-Wesley, Reading (2002)
7. Jones, B.F., Fairney, P.T.: Recognition of shiny dielectric objects by analyzing the polarization of reflected light. Image and Vision Computing Journal 7 (1989)

8. Meriaudeau, F., Ferraton, M., Stolz, C., Morel, O., Bigué, L.: Polarization imaging for industrial inspection, vol. 6813 (2008)
9. Miyazaki, D., Kagesawa, M., Ikeuchi, K.: Transparent surface modeling from a pair of polarization images. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(1), 73–82 (2004)
10. Morel, O., Stolz, C., Meriaudeau, F., Gorria, P.: Active lighting applied to three-dimensional reconstruction of specular metallic surfaces by polarization imaging. Applied Optics 45(17), 4062–4068 (2006)
11. Rahmann, S., Canterakis, N.: Reconstruction of specular surfaces using polarization imaging. In: Proc. IEEE Conf. Computer Vision and Pattern Recognition, vol. 1, pp. 149–155 (2001)
12. Saupe, D., Vranić, D.V.: 3-d model retrieval with spherical harmonics and moments. In: Proceedings of the 23rd DAGM-Symposium on Pattern Recognition, pp. 392–397. Springer, Heidelberg (2001)
13. Shen, L., Ford, J., Makedon, F., Saykin, A.: A surface-based approach for classification of 3d neuroanatomic structures. Intelligent Data Analysis 8(6), 519–545 (2004)
14. Shi, J., Malik, J.: Normalized cuts and image segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence 22(8), 888–905 (2000)
15. Sun, G., Onoichenco, E., Fu, Y., Liu, Y., Amell, R., McCandless, C., Reddy, R., Kumar, G., Guest, M.: High-throughput polarization imaging for defocus and dose inspection for production wafers, vol. 6518 (2007)
16. Wolff, L.B.: Polarization vision: a new sensory approach to image understanding. Image and Vision Computing 15, 81–93 (1997)
17. Wolff, L.B., Boult, T.E.: Constraining object features using a polarisation reflectance model. IEEE Transactions on Pattern Analysis and Machine Intelligence 13(7), 635–657 (1991)

Measurement of Defocus Level in Iris Images Using Different Convolution Kernel Methods

J. Miguel Colores-Vargas[1], Mireya S. García-Vázquez[1],
and Alejandro A. Ramírez-Acosta[2]

[1] Instituto Politécnico Nacional-CITEDI, Av. Del Parque No.1310, Tijuana BC,
{colores,mgarciav}@citedi.mx
[2] MIRAL R&D, 1047 Palm Garden, Imperial Beach, 91932 USA
ramacos10@hotmail.com

Abstract. During the video and fixed image acquisition procedure of an automatic iris recognition system, it is essential to acquire focused iris images. If defocus iris images are acquired, the performance of the iris recognition is degraded, because iris images don't have enough feature information. Therefore it's important to adopt the image quality evaluation method before the image processing. In this paper, it is analyzed and compared four representative quality assessment methods on the MBGC iris database. Through methods, it can fast grade the images and pick out the high quality iris images from the video sequence captured by real-time iris recognition camera. The experimental results of the four methods according to the receiver operating characteristic (ROC) curve are shown. Then the optimal method of quality evaluation that allows better performance in an automatic iris recognition system is founded. This paper also presents an analysis in terms of computation speed of the four methods.

Keywords: Convolution kernel, defocus, iris, quality, video.

1 Introduction

Nowadays, the development of better image quality metrics is an active area of research. The image quality plays a crucial role in the pattern matching system, particularly in automated biometric systems, like iris recognition where performance is based upon matching fine texture information in the annular region between the pupil and the sclera. Some studies report that using a high quality image affects recognition accuracy and can improve system performance [1]. Then, it is necessary to select a suitable image with high quality from an input sequence before all sequent operations. Otherwise, it can have a negative impact on segmentation algorithms and may be difficult to normalize and match, increasing the error probability [2], [3]. For example, in a capturing iris images system the subject to recognize usually moves his head in different ways gives rise to non-ideal images (with occlusion, off-angle, motion-blur and defocus) for recognition. A sample set of all these problems in images are shown in figure 1. As noted, choosing an appropriate image with quality seems a challenge.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 125–133, 2010.

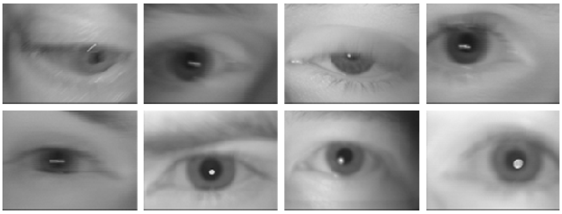


Fig. 1. Video sequences depicting various problems during acquisition of eyes images

To address the problem of image quality, related work on this subject can be placed into two categories [4]: local and global analyses. Local methods try to classify each pixel of the iris providing additional information about each region of the iris texture. The major feature of these approaches is that the evaluation of iris image quality is reduced to the estimation of a single or a pair of factors, such as out-of-focus blur, motion blur, and occlusion. Iris quality should not be limited to one or two quality factors [2-6]. Zhang and Salganicaff [7] examine the sharpness of the region between the pupil and the iris. Moreover, the majority of previously methods require involvement of traditional segmentation methods that are iterative and thus computationally expensive.

In this work are analyzed the mainly used global methods. These approaches are good for quickly eliminating very poor quality images. They are based on convolution kernel for measuring defocus level in iris images. The segmentation is not required because the operator is applied to the entire image, giving the possibility of its implementation at hardware level.

This paper is organized as follows. Section 2 explains the principles of kernel-based defocus measurements and presents the main representative kernels used for of iris recognition systems. Methodologies for comparing kernels and results are given in Section 3, and Section 4 gives the conclusion.

2 Measurement of Blurring and Defocus in Iris Images

A clear image has relatively uniform frequency distribution in the 2D Fourier spectrum. On the other hand, the energy of a defocused or blurred image concentrates on the lower frequency part [7]. Therefore, spectral analysis of the frequency suggests that an effective way to estimate the degree of focus is measure its energy total at higher spatial frequencies. This is a common method in research on image quality assessment [8-11]. In a recognition system if an image can pass a minimum focus criterion, it will be used for recognition.

Thus, it needs a discrete formulation to obtain only the high frequency power, this can be solved filtering the low frequency part of the image, calculating the energy of the processing image (low frequency filtrated image) and set predefined threshold i.e. that images with energy values higher than the threshold are identified as clear images. Section 3 explains the methodology for calculating the thresholds, the optimum values for different filters are shown in table 1.

2.1 Convolution Kernels

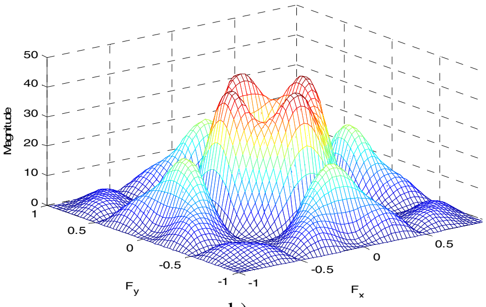
In order to obtain the high frequency power of the image, a proper high-pass convolution kernel is really important. In this section, it will give a brief description of convolution kernels presented in literature to determine the defocus degree in eye-iris images.

2.1.1 Daugman's Convolution Kernel

In his pioneering work [8], Daugman proved that the defocus primarily attenuates high spatial frequencies. Due to this relationship he improved the convolution operation in real-time, id est., to reduce the computational complexity of the Fourier transform proposed a high pass 8×8 convolution kernel to extract the high frequency of an image. The convolution kernel and his spectrum response are shown in figure 2. The weights consists of two square box functions, one of size 8x8 with amplitude -1, and the other one of size 4x4 and amplitude of +4.

-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	3	3	3	3	-1	-1
-1	-1	3	3	3	3	-1	-1
-1	-1	3	3	3	3	-1	-1
-1	-1	3	3	3	3	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1
-1	-1	-1	-1	-1	-1	-1	-1

a)



b)

Fig. 2. a) The 8x8 Convolution kernel proposed by Daugman [8] b) The frequency response (Fourier Spectrum)

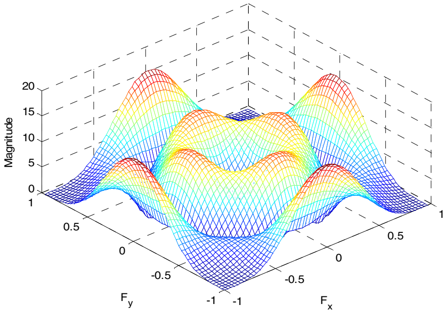
2.1.2 The Convolution Kernel of Wei et al.

Wei et al. [9] also suggest a convolution kernel, with a similar shape as Daugman's for detecting defocused on still and segmented images. Additionally, they detect other problems presented in iris images; motion blur and occlusion. Each problem has its own peculiarity, so, the three features are used to classify them using SVM (Support Vector Machine) that is a machine learning algorithm method used for classification [11]. The total quality of an image according with their method is a vector Q (q1, q2, q3), where the values represent the levels from defocus, motion blur and occlusion respectively.

To determine the defocus degree they proposed a 5×5 convolution kernel as shown in figure 3. Compared with Daugman's 8×8 convolution kernel is also a lower frequencies filter but computationally less demanding. The operator is formed by three box functions, one of size 5x5 with amplitude -1, one of size 3x3 with amplitude +3, and the last one of size 1x1 with amplitude -2.

-1	-1	-1	-1	-1
-1	2	2	2	-1
-1	2	0	2	-1
-1	2	2	2	-1
-1	-1	-1	-1	-1

a)



b)

Fig. 3. a) The 5x5 Convolution kernel proposed by Wei et al. b) The frequency response (Fourier Spectrum)

2.1.3 Laplacian of Gaussian Convolution Kernel

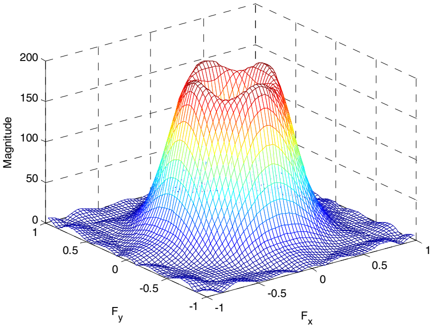
J. Wang et al. [10] propose a convolution kernel operator is based on a Laplacian of Gaussian function (LoG). The Laplacian give a 2-D isotropic measure of the second spatial derivative of an image, in the image highlights regions represent the rapid intensity change and is used for edge detection. The Gaussian smooth filter is applied first to the image; the purpose is to reduce the noise sensibility of the second derivative. They calls filter Laplacian of Gaussian filter. The combined filter function is centered on zero and with Gaussian standard deviation σ has the form given by the equation:

$$LoG(x,y) = -\frac{1}{\pi\sigma^4}\left[1 - \frac{x^2+y^2}{2\sigma^2}\right] e^{-\frac{x^2+y^2}{2\sigma^2}} \tag{1}$$

Since the image is represented as a set of discrete pixels, they sought a discrete convolution kernel that can approximate the Laplacian operator. Set different values of the Gaussian, finally they used $\sigma = 1.4$, this process results in a LoG operator as shown in figure 4.

0	1	1	2	2	2	1	1	0
1	2	4	5	5	5	4	2	1
1	2	4	5	5	5	4	2	1
2	5	3	-12	-24	-12	3	5	2
2	5	0	-24	-40	-24	0	5	2
2	5	3	-12	-24	-12	3	5	2
1	2	4	5	5	5	4	2	1
1	2	4	5	5	5	4	2	1
0	1	1	2	2	2	1	1	0

a)



b)

Fig. 4. a) The 9x9 Convolution kernel based in Laplacian and Gaussian filter b) The frequency response (Fourier Spectrum)

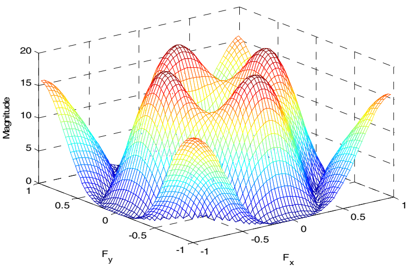
2.1.4 The Convolution Kernel of Kang and Park

Kang & Park [11] propose 5x5 pixels sized convolution kernel as shown in figure 5. It consists of three square box functions, one of size 5x5 with amplitude -1, one of size 3x3 and amplitude +5, and other of size 1x1 and amplitude -5.

However, they argue that their 5x5 pixels convolution kernel contains more high frequency bands than the 8x8 pixels convolution kernel proposed by Daugman [8]. From that, theoretically the operator can detect much better the high frequency of iris texture, using less processing time due to the short sized kernel.

-1	-1	-1	-1	-1
-1	-1	4	-1	-1
-1	4	4	4	-1
-1	-1	4	-1	-1
-1	-1	-1	-1	-1

a)



b)

Fig. 5. a) The 5x5 Convolution kernel proposed by Kang & Park for defocus measure b) The frequency response (Fourier Spectrum)

3 Experimental Results

3.1 The Iris Video Dataset

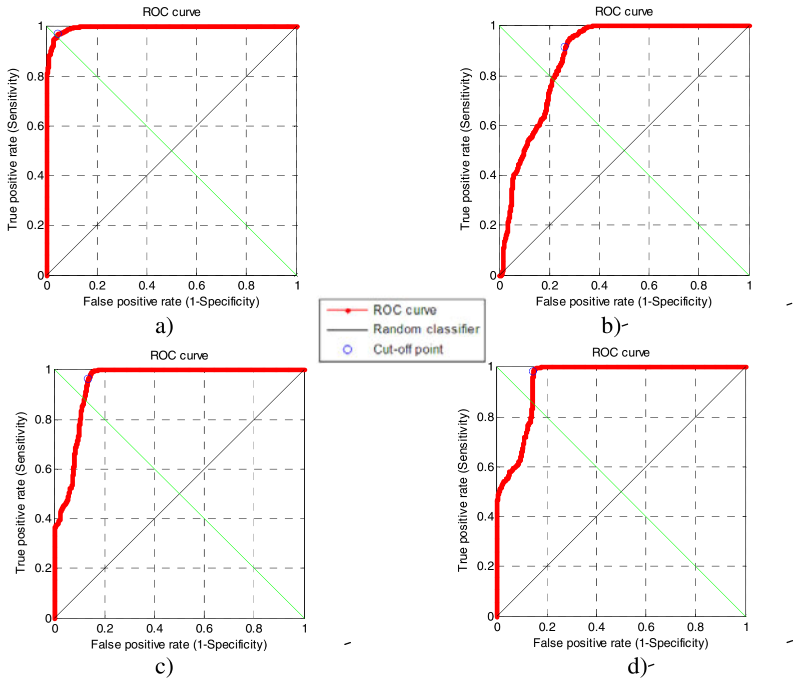
There are many iris databases such as the CASIA, UBIRIS and NIST ICE etc [15-17], but they do not include many motion blurred and optical defocus iris images. For the purpose of validating the effectiveness and compare the performance of the kernels in this work, it is produced a database with 4432 images for testing. The images were extracted from 100 different videos from the MBGC (Multiple Biometrics Grand Challenge) database.

The MBGC dataset used for this experiment was collected by The Computer Vision Research Lab at the University of Notre Dame and provided for the (MBGC) evaluation [13]. Actually this dataset has been distributed to over 100 research groups around the world. All videos were acquired using an LG2200 EOU iris capture system [14]. The camera uses near-infrared illumination of the eye. During acquisition, the eye is illuminated by one of three infrared LEDs: one above the lens, one to the left, and one to the right. The iris video sequences were digitized from the NTSC (National Television System Committee) video signal from the sensor. The signal produced was digitized by a DayStar XLR8 USB video digitizer attached to a Macintosh host system and stored in MPEG-4 format with a high bit rate allowance thus yielding with lossless encoding. The size for each frame in the video has 480 rows and 640 columns in 8 bits-grayscale space (intensity values between 0 to 255). Our dataset (4432 images) contains 2077 clear images (positive samples) and 2355 defocused images (negative samples). The entire collection of images was passed through the selection subjective process (based on human perception of quality). In images where we had uncertainty about whether it was a clear image or a defocus one, verification tests were implemented. Thus, the negative samples come from those images that cannot be segmented by Libor Masek recognition algorithm [18].

3.2 Best Iris Image Selection

For the testing process, it is computed the energy of every image from our iris database. The Receiver Operating Characteristic (ROC, also known as a Relative Operating Characteristic) curves was used to obtain the optimal threshold decision, we compared the difference between the two curves generated by energy values; one curve is generated by tests with defocused images and other curve by tests with focus images.

When it is considered the results of a particular test in two classes, it will rarely observe a perfect separation between the two groups. Indeed, the distribution of the test results will overlap. For every possible threshold point or criterion value you select to discriminate between the two classes. If an accepted/positive image (focus image) is a defocus image, it is called a false accept. The percentage of false accepts is called false accept rate (FAR). If a rejected image (defocus image) is a focus image, it is called a false reject. The percentage of false reject is called false reject rate (FRR).



a)

b)

c)

d)

Fig. 6. ROC curves generated from the evaluated convolution kernels a) Daugman b) Wei et al c) LoG d) Kang & Park

In a ROC curve the true positive rate (Sensitivity) is plotted in function of the false positive rate (Specificity) for different cut-off points. Each point on the ROC plot represents a sensitivity/specificity pair corresponding to a particular decision threshold. Figure 6 not only indicates the performance of different methods but also provides information of how much the performance of a given method can vary. That is, the ROC curve is directly related to these two distance distributions. Therefore it is set the threshold for distinguishing the two kinds of images. By the threshold good quality images in the testing dataset passed the evaluation, and the poor quality images were rejected. The False Acceptance Rate (FAR) and False Rejection Rate (FRR) of all kernels are shown in the table 1.

Table 1. The table shows the parameters of the curves generated by experiments and the optimal thresholds desicion for each kernel

Kernel	**Clear iris images**		**Defocus iris images**		**Optimal Decision threshold**	**FAR (%)**	**FRR (%)**
	Mean	**Standard deviation**	**Mean**	**Standard deviation**			
Daugman	55.063	8.9134	26.838	5.7642	**39.9490**	2.8	3.6
Wei et al.	19.928	4.5584	12.291	5.3759	**14.1631**	8.3	2.6
LoG	108.49	10.651	72.88	14.658	**92.0776**	3.7	5.2
Kang & Park	25.792	6.0658	13.25	3.9636	**15.8247**	1.6	2.3

The table contains the obtained results, the first column identifies the evaluated kernel, the next four columns contain the parameters of the distribution curve produced from the tests with defocus and clear images, the sixth column contains the optima's threshold for discrimination of defocus iris images, and the last columns are the error percentages for every kernel. As shown in table, the best performance scores (minimum FAR and FRR) was generated by the kernel proposed by Kang & Park followed of the Daugman kernel who presented also low error rates. The worse results performance kernels were presented by kernels proposed in [9,10], these kernels had the highest error rates.

To compare the convolution kernels in terms of speed is calculate the total multiplication count (TMC). With the 8x8 and 9x9 pixels convolution kernels, the convolution value is calculated per every fourth row and fourth column in the iris image [19] the TMC=1228800 (8x8x640/4 x 480/4) and TMC=1555200 (9x9x640/4 x 480/4). For a 5x5 pixels convolution kernel, the convolution value is calculated per every third row and third column, the TMC=852000 (5x5x640/3 x 480/3). The 5x5 pixels convolution kernels are 30.66% faster than the 8x8 pixels convolution kernels.

4 Conclusions

In this paper, it was analyzed four representative convolution kernels for image quality assessment on the MBGC iris database. They are a simple, fast quality descriptors based on the Fourier spectral of an iris image is defined to discriminate from any given eye image video-sequence, clear iris images from low quality images due to motion blur and defocus. The performance of every convolution kernel analyzing the relationship between the quality of iris images in terms of ROC curves. Some discussions can be given based on the results:

- The algorithm deals with the whole image, avoiding the location and segmentation of the pupil and iris.
- The execution time of the algorithm is suitable for a real-time recognition system.
- The algorithm is effective for the defocused and motion blurred images, but it is ineffective for the occluded images of eyelids and eyelashes there are rich in middle and high frequency components, which is an important factor in discriminating such images from clear images.
- It can adjust the threshold to get a satisfied FAR and a tradeoff between FAR and FRR.

From the above analysis and results is concluded that the Kang & Park convolution kernel is superior to the other three kernels in terms of speed and accuracy.

Acknowlegment. This work was realized under grant SIP20100030.

References

1. Gamassi, M., Lazzaroni, M., Misino, M., Piuri, V.: Quality assessment of biometric systems: a comprehensive perspective based on accuracy and performance measurement. IEEE Transactions on Instrumentation and Measurement 54, 1489–1496 (2005)
2. Chen, Y., Dass, S.C., Jain, A.K.: Localized iris image quality using 2-D wavelets. In: Zhang, D., Jain, A.K. (eds.) ICB 2005. LNCS, vol. 3832, pp. 373–381. Springer, Heidelberg (2005)
3. Kalka, N.D., Zuo, J., Schmid, N.A., Cukic, B.: Image quality assessment for iris biometric. In: SPIE 6202: Biometric Technology for Human Identification III, vol. 6202, pp. D1–D11 (2006)
4. Zuo, J., Schmid, N.A.: Global and local quality measures for NIR iris video. In: IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops, CVPRW 2009, pp. 120–125 (2009)
5. Zhu, X.-D., Liu, Y.-N., Ming, X.: A quality evaluation method of iris images sequence based on wavelet coefficients in region of interest. In: CIT '04: Proceedings of the The Fourth International Conference on Computer and Information Technology, Washington, DC, USA, pp. 24–27. IEEE Computer Society, Los Alamitos (2004)
6. Ma, L., Tan, T., Wang, Y.: Personal identification based on iris texture analysis. IEEE Transactions on Pattern Analysis and Machine Intelligence 25(12), 1519–1533 (2003)
7. Zhang, G., Salganicoff, M.: Method of measuring the focus of close-up image of eyes. Tech. Rep. 5953440, United States Patent (1999)
8. Daugman, J.G.: How iris recognition works. IEEE Trans. Circ. Syst. Video Tech. 14(1), 21–30 (2004)
9. Wei, Z., Tan, T., Sun, Z., Cui, J.: Robust and Fast Assessment of Iris Image Quality. In: Zhang, D., Jain, A.K. (eds.) ICB 2005. LNCS, vol. 3832, pp. 464–471. Springer, Heidelberg (2005)

10. Wang, J., He, X., Shi, P.: An Iris Image Quality Assessment Method Based on Laplacian of Gaussian Operation. In: MVA2007 IAPR Conference on Machine Vision Applications, Tokyo, Japan, May 16-18, pp. 248–251 (2007)
11. Kang, B.J., Park, K.R.: A study on iris image restoration. In: International Conference on Audio- and Video-Based Biometric Person Authentication, pp. 31–40 (2005)
12. Burges, J.C.: A Tutorial on Support Vector Machines for Pattern Recognition. Data Mining and Knowledge Discovery 2, 121–167 (1998)
13. Multiple Biometric Grand Challenge, `http://face.nist.gov/mbgc/`
14. Jonathon Phillips, P., Bowyer, K.W., Flynn, P.J., Liu, X., Todd Scruggs, W.: The Iris Challenge Evaluation 2005. In: Biometrics: Theory, Applications and Systems, Washington, DC (September 2008)
15. CASIA iris image database, `http://www.sinobiometrics.com`
16. National Institute of Standards and Technology (NIST). Iris Challenge Evaluation (2008), `http://iris.nist.gov/ice/`
17. Proença, H., Alexandre, L.A.: UBIRIS: A noisy iris image database. In: Roli, F., Vitulano, S. (eds.) ICIAP 2005. LNCS, vol. 3617, pp. 970–977. Springer, Heidelberg (2005)
18. Masek, L.: Recognition of human iris patterns for biometric identification. Master's thesis, University of Western Australia (2003)
19. Daugman, J.G.: How Iris Recognition Works. IEEE Trans. on Circuits and Systems for Video Technology 14(1), 21–30 (2004)

Radon Transform Algorithm for Fingerprint Core Point Detection

Miguel Mora-González[1], Julio C. Martínez-Romo[2], Jesús Muñoz-Maciel[1],
Guillermo Sánchez-Díaz[3], Javier Salinas-Luna[3], H.I. Piza-Dávila[4],
Francisco J. Luna-Rosas[2], and Carlos A. de Luna-Ortega[1]

[1] Universidad de Guadalajara (UdeG), Centro Universitario de los Lagos, Av. Enrique
Díaz de León 1144, Col. Paseos de la Montaña, C.P. 47460, Lagos de Moreno, Jal., México
mmora@culagos.udg.mx, jesnek@hotmail.com,
alejandro.deluna@upa.edu.mx
[2] Instituto Tecnológico de Aguascalientes, Av. A. López Mateos 1801 Ote. Col. Bona Gens,
C.P. 20256, Aguascalientes, Ags., México
jucemaro@yahoo.com, fjluna@ita.mx
[3] UdeG, Centro Universitario de los Valles, Km 45.5 carr. Guadalajara-Ameca,
C.P. 46600, Ameca, Jal., México
guillermo.sanchez@profesores.valles.udg.mx,
javier.salinas@profesores.valles.udg.mx
[4] Instituto Tecnológico y de Estudios Superiores de Occidente, Apdo. Post. 31-175,
C.P. 45604, Tlaquepaque, Jal., México
hpiza@iteso.mx

Abstract. This article presents an innovative technique for solving the problem of finding the core within a fingerprint. The Radon transform and a tree clustering algorithm were key to locating the coordinates of the core. Binarization and high-pass filtering processes to improve the contrast in fingerprints are proposed. The core of a fingerprint is located in the geometric cross section of maxima and minima in the Radon transforms at 0° and 90°. The technique is very stable, since it only presents difficulties when the fingerprint core is located on the edges of the image or is nonexistent.

Keywords: Core Point, Fingerprint, Edge Detection, Radon Transform.

1 Introduction

Fingerprint analysis is a rather common method for identifying people. This is so because of two biometrical features of fingerprints: they do not change with time and they are unique for each person [1]. That is the reason fingerprint recognition has a wide field of application in security and recognition systems.

There are different parameters to identify in a fingerprint: ridges, rows, deltas, cores, etc., the latter being the parameter around which the others converge. Due to its importance, many authors have implemented several techniques to identify the geometrical position of the core within the image of a fingerprint. At first they found the core by dividing the image into sub-images [2], [3]. Then, curvature detection

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 134–143, 2010.

methods and region geometry were used [4]. Others began to use the Poincare Index [5], [6], [7]. Finally, some researchers worked with the orientation, segmentation [8] and curvature [9] patterns of the fingerprint.

In this work, we implemented an alternative approach to finding the core in a fingerprint by using the Radon Transform (RT) as a means to quantify the grey levels between the curves generated by ridges and rows of each fingerprint. We also performed an analysis in convolution filters to improve the images in the process for obtaining the RT. Finally, a tree clustering algorithm is used to match the core coordinates through empathy between the horizontal and vertical RTs of a fingerprint. It was decided to use RT as a method to find the core of a fingerprint because its algorithm is easy and quick to implement and because the results are very straightforward.

In the following sections, the proposed methodology to develop the fingerprint core-locating algorithm is presented. Then, the experimental setup for capturing fingerprints is explained. After that, the experimental results obtained through the proposed algorithm are shown and discussed. Finally the conclusions of the results are expressed.

2 Methodology

The motivation to use RT as a method to locate the core in fingerprints came up from observing the shape of the prints themselves, since the cores in the great majority of them are shown as a collection of concentric circles. When the grey levels (GL) around the circles are analyzed, it is observed that the highest GL is in the convergence point of such circles.

In order to find the core in the fingerprint, the following digital processes need to be applied to the image: binarization, convolution, Radon transform, least squares, and tree clustering algorithm. These processes are detailed below.

2.1 Image Binarization

A large amount of GL is obtained through the scanning of fingerprints. This also depends on the way the image is scanned. Ideally, there should be only one GL binary range, i.e. two GL (0 and 1). To perform the binarization, an intermediate grey level is chosen as a threshold value, thus making the binarized image obtainable through these two equations:

$$\mathrm{Im}_b(x,y) = \begin{cases} \mathrm{Im}(x,y) < GL_m, & 0 \\ \mathrm{Im}(x,y) > GL_m, & 1 \end{cases} \tag{1}$$

and

$$\mathrm{Im}_n(x,y) = 1 - \mathrm{Im}_b(x,y)\,, \tag{2}$$

where Im, Im_b, Im_n, (x,y) and GL_m are the original image, the binarized image, the negative, the coordinates of each image and the GL threshold, respectively. An unprocessed fingerprint (Im, the location of the core is enclosed) can be observed in

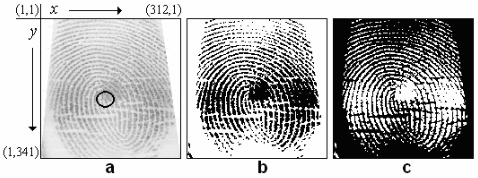


Fig. 1. Fingerprint binarization, a) original fingerprint, b) binary fingerprint, and c) negative fingerprint. Threshold GL_m=200 GL.

Fig. 1a; the binarized image (Im$_b$) is shown in Fig. 1b; and the negative (Im$_n$) is presented in Fig. 1c.

2.2 Convolution

The image of a binarized fingerprint shows the ridges and rows as the skeleton of the original image. However, it is necessary to highlight the data in order to obtain more defined lines and thus be able to get a clearer distinction between ridges and rows (distinguishing the edges). The most common method to detect the edges of an image is spatial filtering, which convolutions the image against a high-pass mask. Convolution mask types that highlight the edges (for North, South, East, and West orientation) are gradient and relief. These types of masks are shown in Table 1 [10], [11]. The mathematical model of discrete convolution applied to images is defined as [12]

$$\mathrm{Im}(x,y) * hp(x,y) = \sum_{m=-\frac{M+1}{2}}^{\frac{M+1}{2}} \sum_{n=-\frac{N+1}{2}}^{\frac{N+1}{2}} \mathrm{Im}(m,n)hp(x-m,y-n), \qquad (3)$$

where *hp*, (*m*,*n*) and *M*x*N* are the convolution mask, the coordinates where the convolution is performed, and the size of the convolution masks, respectively.

Table 1. Some edge-detecting matrices or masks

	North	South	East	West
Gradient	1 1 1	−1 −1 −1	−1 1 1	1 1 −1
	1 −2 1	1 −2 1	−1 −2 1	1 −2 −1
	−1 −1 −1	1 1 1	−1 1 1	1 1 −1
Relief	1 1 1	−1 −1 −1	−1 0 1	1 0 −1
	0 1 0	0 1 0	−1 1 1	1 1 −1
	−1 −1 −1	1 1 1	−1 0 1	1 0 −1

The masks shown in Table 1 vary on their performance, depending on the characteristics of the fingerprint to be convolutioned. Images in Fig. 2 are the result of the application of equation 3 (through masks in Table 1) to the fingerprint in Fig. 1a. In Fig. 2 we can observe the edges detection being highlighted in four directions. Combinations such as Northeast, Southeast, Northwest and Southwest are also possible.

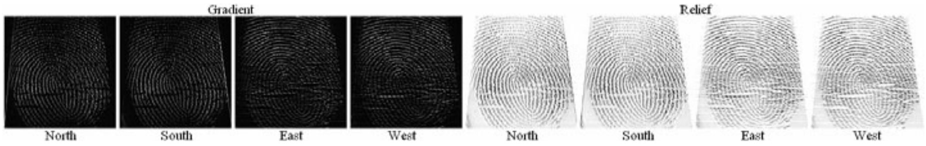


Fig. 2. Skeletonization of a fingerprint (Fig 1a) through Table 1 convolution masks

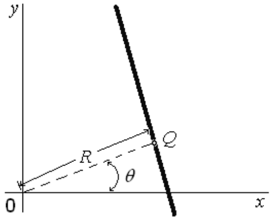


Fig. 3. Radon transform parameters

2.3 Radon Transform

The Radon transform may be considered the image's grey levels projection over a given angle with respect to the *x* axis. The RT mathematical model is [13]

$$\Re\{\mathrm{Im}(x,y)\} = \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} \mathrm{Im}(x,y)\,\delta(R - x\cos\theta - y\sin\theta)\,dxdy, \tag{4}$$

where $\Re$, δ, R and θ are the Radon transform operator, the unit impulse function, the distance from the origin to the profile line and the angle of direction of the same line, respectively. Each of these parameters can be observed in fig. 3, where Q is the origin of the profile line to be obtained (thick bold line). We find the Radon transform useful in the detection cores within fingerprints due to its ability to detect lines; a fingerprint may be considered roughly as composed of vertical and horizontal lines with the core located at the intersection of such lines. Radon transforms (with $\theta = 0°$ and 90°, RT_0 and RT_{90}) for fingerprints from figures 1 and 2 are shown in Fig. 4. It can be observed that the core is the point in the fingerprint where RT_0 minima and RT_{90} maxima intersect (dotted lines).

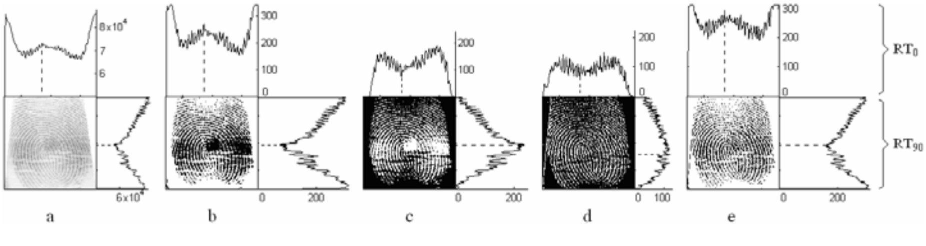


Fig. 4. Radon transform 0° and 90° for: a) original fingerprint, b) binarized, c) negative, d) north gradient, and e) south relief image. Core point is in the intersection of dotted lines.

2.4 Curve fitting by Least Squares

Due to the need to adjust the RTs for better identification of maxima and minima, least squares method has been employed. The basic idea is fitting the data obtained from RTs through the polynomial

$$h(x) = \sum_{j=1}^{k+1} c_j x^{k-j-1}, \tag{5}$$

calculating the deviation of the curve with respect to the polynomial through [14]

$$r_i = f_i - h(x_i), \text{ with } i = 1,2,\cdots,I, \tag{6}$$

where c, k, f, and I are the polynomial coefficients, the polynomial degree, the data to be fit, and the number of data, respectively. Thus, the addition of deviations raised to the second power is represented by the following equation

$$D = \sum_{i=1}^{I} r_i^2, \tag{7}$$

which is why the minimum of every squared deviation will become evident when the partial derivative with respect to each coefficient c is zero. This is

$$\frac{\partial D}{\partial c_j} D = 0, \tag{8}$$

when substituting equations (5-7) in equation (8) and partially deriving them against each coefficient, we obtain

$$\sum_{j=1}^{n+1} \left[\sum_{i=1}^{N} x_i^{n-j-1+k} \right] \cdot c_j = \sum_{i=1}^{N} x_i^k f_i \, . \tag{9}$$

Polynomial $h(x)$ coefficients can be obtained through equation (9), which results in the parameters needed for the least squares fit.

2.5 Tree Clustering Algorithm

In order to find the minima and maxima observed in the RTs of images shown above, a tree clustering algorithm is proposed. This algorithm performs a comparative adjustment between data and threshold value. It detects sign changes in the data function slope. The algorithm consists of the following steps [15]:

```
Comparison ratio r is defined, where r<<xi;
Counters kmax=0 and kmin=0 are defined;
for i=r to I-r; // I is the number of data.
   for j=i-r to i+r;
   if h(xi)> h(xj); // to find maxima Tmax.
      then kmax=kmax+1 and Tmax(kmax)=h(xi);
   if h(xi)< h(xj); // to find minima Tmin.
      then kmin=kmin+1 and Tmin(kmin)=h(xi);
```

```
maxima=Tmax(1); // initialization variable maxima.
for i=1 to kmax-1; // to find principal maxima.
  if Tmax(i+1)>Tmax(i);
    then maxima=Tmax(i+1);
minima=Tmin(1); // initialization variable minima.
for j=1 to kmin; // to find principal minima.
  if Tmin(j+1)<Tmin(j);
    then minima=Tmin(j+1);
```

Finding all the maxima and minima in a signal will depend on two factors: signal noise and size of r. It is recommended that the noise frequency be lower than 2r, thus fitting into Nyquist criterion [16].

3 Experimental Setup

The setup used to capture fingerprints is quite simple: a Microsoft FingerPrint scanner connected via USB to a PCG-K35F Sony Vaio laptop. The scanner produces 355×390-pixel images in BMP format. The images have a margin with no information (electronically removed), resulting in 312×341-pixel images. Matlab® was the software used to implement the methods seen in the previous section.

4 Experimental Results

Based on the results from Fig. 4, it can be observed that the core perimeter is signaled by a series of maxima and minima in the RT graphs. The problem with such graphs is the excessive noise they contain, which is generated by the high number of grey levels and/or the low contrast between rows and ridges in each of the sub-images shown in Figure 4. It was also observed that images with a black background (negative and gradient in Fig. 4) had a smaller amount of maxima and minima, which facilitates pinpointing the cores. It was then decided that an adjustment had to be made in the RT graphs using least square fitting. This softens the RTs graphs without losing relevant information in them, as can be seen comparing Figures 4 and 5.

Table 2. Core coordinates and standard deviations for 7 individuals' fingerprints using different convolution and binarization parameters. Where: A= Binarizated gradient, B= Binarizated negative relief, P= Process, S= Sample and GL_m=180.

S	P	Core coordinates and σ (in pixels)							Fig.
		(x_n,y_n)	(x_s,y_s)	(x_e,y_e)	(x_w,y_w)	(x_c,y_c)	σ_x	σ_y	
1	A	(143, 189)	(152, 188)	(155, 115)	(156, 87)	(152, 145)	5.9	51.8	5c
	B	(147, 186)	(140, 186)	(147, 180)	(148, 188)	(146, 185)	3.7	3.4	5d
2	A	(166, 216)	(147, 237)	(154, 154)	(171, 176)	(160, 196)	11	37.7	6a
	B	(157, 199)	(152, 202)	(154, 201)	(154, 208)	(154, 203)	2	3.9	6b
3	B	(138, 180)	(131, 173)	(168, 85)	(167, 88)	(151, 132)	19.3	52.1	6c
4	B	(119, 183)	(116, 185)	(112, 214)	(117, 221)	(116, 201)	2.9	19.6	6d
5	B	(156, 173)	(149, 177)	(162, 175)	(162, 191)	(157, 179)	6.2	8.2	6e
6	B	(174, 200)	(173, 209)	(143, 77)	(142, 77)	(158, 141)	18	73.7	6f
7	A	(103, 317)	(242, 316)	(235, 261)	(173, 226)	(188, 280)	64.7	44.5	6g
	B	(171, 243)	(166, 202)	(178, 243)	(173, 255)	(172, 236)	5	23.2	6h

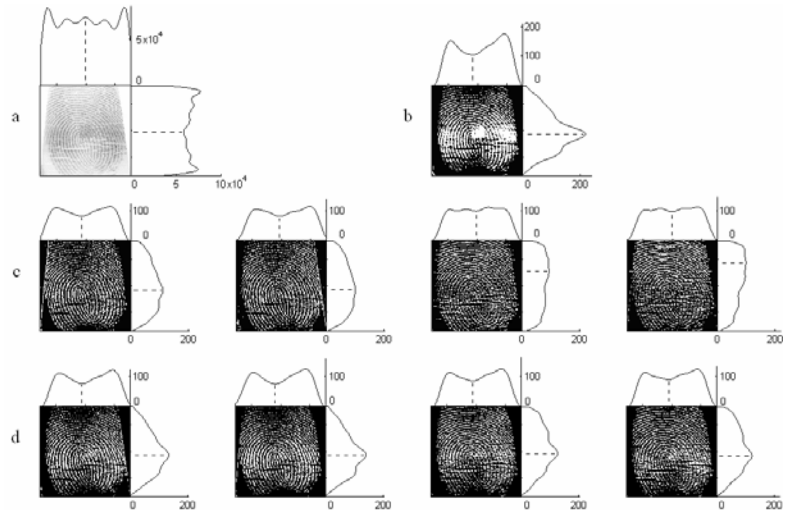


Fig. 5. Softened RTs at 0° and 90° for a) original fingerprint individual 1; b) binarized negative image; North, South, East, West images for c) binarized gradient; and d) binarized negative relief. (x,y), (x_n,y_n) (x_s,y_s), (x_e,y_e) and (x_w,y_w) core points are in the intersection of dotted lines.

We began by comparing the adjusted RT graphs using a 21th grade polynomial, thus it was possible to fit the curves without a lost of maxima or minima of the original RT. The softened RTs from the original fingerprint and the softened RTs for the 4 directions of the binarized gradient and the binarized relief are shown in Fig. 5. A negative relief was used in order to obtain images with a black background. It was observed that images processed with relief-type convolution masks only have one maximum in TR_{90} and a minimum in TR_0, whereas images processed with gradients have, in most cases, several maxima and minima. The reason is that relief-processed images have a better contrast than gradient-processed images. Maxima and minima were obtained through a tree clustering algorithm (described in section 2.5), given the fact that image skeletonization processes (North, South, East, West) both for gradients and reliefs have a slight image deviation in the sense of direction of the process to be carried out. Then, it was decided to obtain the core location using the average of the 4 minima for the RT_0 and the 4 maxima for the RT_{90}. This is

$$x_c = (x_n + x_s + x_e + x_w)/4, \tag{10}$$

and

$$y_c = (y_n + y_s + y_e + y_w)/4, \tag{11}$$

where (x_c,y_c), (x_n,y_n), (x_s,y_s), (x_e,y_e) and (x_w,y_w) are the coordinates of the core in the original, North, South, East, and West images, respectively. The standard deviation was also calculated for coordinates in x (σ_x) and for coordinates in y (σ_y). This was done to observe a possible core deviation with respect to the real one. We obtained softened RTs for other 6 individuals' fingerprints, whose core shapes and core positions are different (Fig. 6). Table 2 shows the results of the calculations of core coordinates of the 7 individuals' fingerprints.

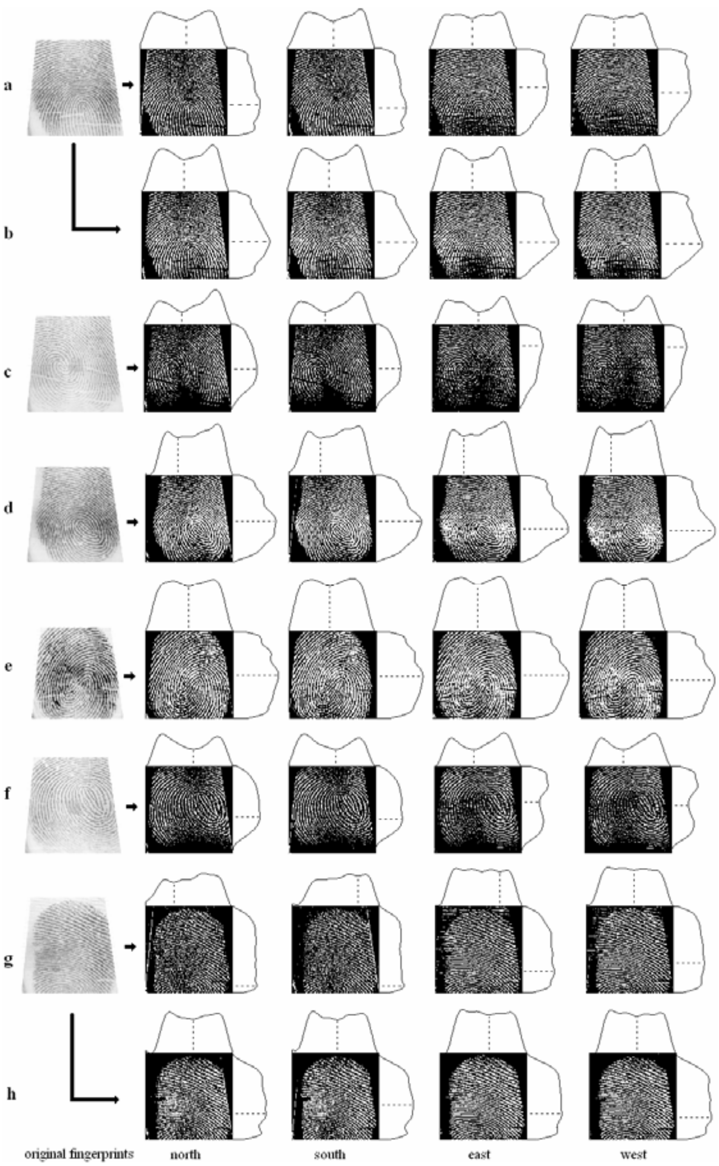


Fig. 6. Softened RTs at 0° and 90° for 6 individuals. (x_n,y_n) (x_s,y_s), (x_e,y_e) and (x_w,y_w) core points are in the intersection of dotted lines

5 Conclusions

A new technique for locating cores in fingerprints has been described. The Radon transform was used as a mathematical tool to locate the cores. RTs are algorithms which are fast to process and to interpret in small images. Therefore it was relatively fast to identify maxima and minima that pointed at the core coordinates.

The technique described above was very stable when used in one or two core fingerprints. However, it destabilizes when the core is hidden. There are two possible cases: a) when the core is hidden in North or South positions, there is more than one maximum in RT_{90}; b) when the core is hidden in the East or West positions, there is more than one minimum in RT_0.

By applying only RT and least square fitting to a fingerprint works fine when dealing with well-scanned or high contrast images. In fingerprints with digitalization mistakes (low contrast or missing data,) however, it is better to apply convolution masks (gradient or relief) before in order to obtain more defined rows and ridges and average the 4 pairs of the softened RTs at the end.

The best convolution mask option depends on the digitalization and type of fingerprint. The results show that masks that presented a higher accuracy finding the core were those in which the standard deviation was minimal, it is to say, when $\sigma_x \rightarrow 0$ and $\sigma_y \rightarrow 0$.

The core in a fingerprint is the most important parameter. Consequently, being able to find it correctly is a major step in the identification of people through fingerprint recognition. For double-cored fingerprints, the central point between the two cores was located (see Fig. 6e).

The core was identified localizing the minima of the RT_0 and the maxima of the RT_{90} using a tree clustering algorithm. These minima and maxima are global ones; however the minima of the RT_0 should discard the borders since they do not contain the fingerprint information.

Finally, we found that the proposed method, as compared with others, does not require high resolution images (with 312×341-pixel images were enough); the speed of the algorithm is similar to Poincaré and Edge Map methods and showed a high repeatability rate in the detection of single and double core in fingerprints.

Acknowledgments. The authors would like to thank David A. Leith for his contributions to the development of this work. Also Mr. Mora would like to acknowledge to the Centro Universitario de los Lagos (UdeG) for financing his research stay at Instituto Tecnológico de Aguascalientes in the summer of 2008.

References

1. Cho, B.-H., Kim, J.-S., Bae, J.-H., Bae, I.-G., Yoo, K.-Y.: Core-based Fingerprint Image Classification. In: 15th International Conference on Pattern Recognition, pp. 859–862. IEEE Press, Barcelona (2000)
2. Kameswara Rao, C.V., Balck, K.: Finding the core point in fingerprint. IEEE Trans. Comp. C-27(1), 77–81 (1978)
3. Ohtsuka, T., Kondo, A.: Improvement of the fingerprint core detection using extended relation graph. In: Nonlinear Signal and Image Processing, p. 20. IEEE Press, Sapporo (2005)
4. Julasayvake, A., Choomchuay, S.: An algorithm for fingerprint core point detection. In: 9th International Symposium on Signal Processing and Its Applications, pp. 1–4. IEEE Press, Sharjah (2007)
5. Liu, M., Jiang, X., Kot, A.C.: Fingerprint Referente Point Detection. In: Zhang, D., Jain, A.K. (eds.) ICBA 2004. LNCS, vol. 3072, pp. 272–279. Springer, Heidelberg (2004)

6. Khan, N.Y., Javed, M.Y., Khattak, N., Chang, U.M.Y.: Optimization of core point detection in fingerprints. In: Digital Image Computing Techniques and Applications, pp. 260–266. IEEE Press, Glenelg (2007)
7. Ohtsuka, T., Watanabe, D., Tomizawa, D., Hasegawa, Y., Aoki, H.: Reliable detection of core and delta in fingerprints by using singular candidate method. In: Computer Vision and Pattern Recognition Workshops, pp. 1–6. IEEE Press, Anchorage (2008)
8. Akram, M.U., Tariq, A., Nasir, S., Khanam, A.: Core point detection using improved segmentation and orientation. In: Computer Systems and Applications, pp. 637–644. IEEE Press, Doha (2008)
9. Sun, Q.-s., Mao, Z., Mei, Y.: Detection of Core Points in Fingerprint Images Based on Edge Map. In: 2009 International Conference on Electronic Computer Technology, pp. 126–129. IEEE Press, Macau (2009)
10. Vélez, J.F., Moreno, F.B., Calle, A.S., Sánchez-Marín, J.L.E.: Visión por computador. Dykinson, Madrid (2003)
11. Bow, S.: Pattern Recognition and Image Preprocessing. Marcel Dekker, New York (2002)
12. Gonzalez, R.C., Woods, R.E.: Digital Image Processing. Prentice-Hall, New Jersey (2002)
13. Bracewell, R.N.: Two-Dimensional Imaging. Prentice Hall, New Jersey (1995)
14. Nakamura, S.: Métodos numéricos aplicados con software. Pearson, Naucalpan (1992)
15. Marques de Sá, J.P.: Pattern Recoanition: Concepts, Methods and Applications. Springer, Oporto (2001)
16. Pratt, W.K.: Digital image processing. John Wiley & Sons, Inc., New York (2001)

Genetic Algorithms and Tabu Search for Correcting Lanes in DNA Images

M.J. Angélica Pinninghoff, Q. Daniel Venegas, and A. Ricardo Contreras

Department of Computer Science
University of Concepción, Chile
{mpinning,rcontrer}@udec.cl

Abstract. This paper describes an experience that combines Genetic Algorithms and Tabu Search as a mechanism for correcting lanes in DNA images obtained through Random Amplified Polymorphism DNA (RAPD) technique. RAPDs images are affected by various factors; among these factors, the noise and distortion that impact the quality of images, and subsequently, accuracy in interpreting the data. This work proposes a hybrid method that uses genetic algorithms, for dealing with the highly combinatorial feature of this problem, and tabu search, for dealing with local optimum. The results obtained by using them in this particular problem show an improvement in both, fitness of individuals and execution time.

1 Introduction

Randomly Amplified Polymorphism DNA (RAPDs) [11] is a type of molecular marker which has been used in verifying genetic identity. During the past few years RAPDs have been used for studying philogenetic relationships [1,10], gene mapping [5], trait-associated markers [9], and genetic linkage mapping [2]. This technique has been used as support for many agricultural, forest and animal breeding programs [6].

In Figure 1, a photograph of a RAPD reaction is shown. In this case, 12 samples were loaded of which lanes 1 and 14 correspond to the molecular weight standards. In this case, four different genotypes of Eucalyptus globulus were studied, including three identical copies of each (known as ramets). If the ramets are identical, then quite similar band patterns should be expected when analyzed by the same primer. However, this is not always the case, due to, for example, mislabeling of samples.

The RAPD technique consists of amplifying random sequences of the genomic DNA by using primers, which are commonly 10 bp (base pairs) in length. This process is carried out by polymerase chain reaction (PCR) and generates a typical pattern for a single sample and different primers. The PCR products are separated in an agarose gel, under an electric field which allows smaller fragments of the PCR products to migrate faster, while larger ones much slower. The gel is stained with a dye (typically ethidium bromide) and photographed for further data analysis. One way of analyzing the picture obtained is simply

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 144–153, 2010.

by comparing visually the different bands obtained for each sample. However, this can be a tedious process when various samples with different primer combinations have to be analyzed. At the same time, since, in this case, the presence or absence of bands is to be scored, sometimes the band assessment can be very subjective and there is no reliable threshold level, since the intensities of the bands are affected by several factors (i.e staining, gel quality, PCR reaction, DNA quality, etc.).

During the process of generating the RAPD image, many physical-chemical factors affect the electrophoresis producing different kinds of noise, rotations, deformations and other abnormal distortions in the image. The effect of this problem is, unfortunately, propagated through the different stages in the posterior analysis, including visualization, background extraction, band detection, and clustering, can lead to erroneous biological conclusions. Thus, efficient image processing techniques will, on the other hand, have a positive impact on those biological conclusions.

Typical errors consider rotation in lanes, that is the problem we try to solve. This is the first step; once lanes are corrected, i.e., the complete image shows a minimum slope for each lane, it will be necessary to work in band correction, a difficult problem due to the nature of distortions, different to lane distortion. The second step, band correction, can be carried out in an analogous way; however, it is beyond the scope of this paper.

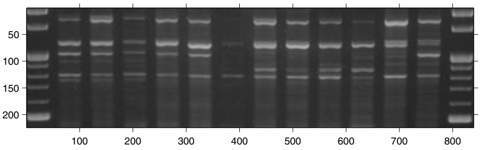


Fig. 1. A sample RAPD image with two reference lanes, and 12 lanes representing four ramets

The basis for this work is the experience described in [7], in which genetic algorithms are used to deal with a population of potential solutions, where solutions are intended as the best templates. A template is a set of lines representing lanes and therefore, the best template is one in which lines match closely to lanes in the original image. This work offered good solutions, although execution time is greater that expected. The other problem this approach presents, is the presence of local minimum.

The aim of this work is to correct distortions in lanes, by using genetic algorithms hybridized with *Tabu Search.* This allows a comparison of two strategies: the first one, already mentioned, that considers single genetic algorithms, and the second one, that uses genetic algorithms and Tabu Search as collaboration mechanisms.

This article is structured as follows; the first section is made up of the present introduction; the second section describes the specific problem to be faced; the third section is devoted to genetic algorithms and tabu search considerations, while the fourth section shows the results we obtained with our approach, and the final section shows the conclusions of the work.

2 The Proposed Approach

The problem addressed in this paper can be formally stated as follows.

Consider an image (matrix) $A = \{a_{ij}\}, i = 1, \ldots, n$ and $j = 1, \ldots, m$, where $a_{ij} \in Z^+$, and A is a RAPD image. Usually, a_{ij} is in the range [0..255] in a grey scale image, and we use a a_{ij} to refer to an element $A(x, y)$, where x and y are the pixel coordinates.

To deal with lane distortions, a set of templates is used. These templates are randomly created images with different distortion degrees, having lines that are in a one-to-one correspondence with lanes in the original RAPD image. A good template is the one that reflects in a more precise degree the distortions that the RAPD image under consideration has.

The template we consider is a matrix L (lanes) where $L = \{l_{ij}\}, i = 1, \ldots, n$ and $j = 1 \ldots, m$, $l_{ij} = 0$ or $l_{ij} = 1$ (a binary image), with 1 meaning that l_{ij} belongs to a line and 0 otherwise. A procedure described in [8] is used to approximately detect the initial position of the lanes. In doing so, the generation of matrix L is limited to those regions that correspond to lanes in matrix A. Due to the rotation of the lanes, it is necessary to consider different alternate configurations. If we are dealing with an image with 12 lanes, and if for each lane we consider 14 possible rotations, we are considering 12^{14} different configurations to evaluate. This causes a combinatorial explosion, which justifies the use of genetic algorithms.

Genetic algorithms allow to manage a large number of templates, and those that are similar to the original image are chosen. Thus, it is necessary to seek for an objective function that reflects this similarity in a precise way. This function is used as a measure for the quality for the selected template.

When the lane correction procedure is applied, templates contain straight lines. Different templates will show different slopes for each line, as shown in Figure 2. A template contains non-intersecting vertical lines, which are not necessarily parallel.

Results obtained in a previous work are promising but not really good. In considering this, we decided to hybridize the solving strategy by adding a Tabu

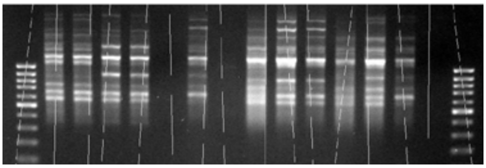


Fig. 2. A sample template for lane correction

Search component. Tabu Search is a mathematical optimization method belonging to the class of local search techniques. Tabu search enhances the performance of a local search method by using memory structures: once a potential solution has been determined, it is marked as "taboo" so that the algorithm does not visit that possibility repeatedly.

3 Genetic Algorithms and Tabu Search

Genetic algorithms (GA) are a particular class of evolutionary algorithms, used for finding optimal or good solutions by examining only a small fraction of the possible space of solutions. GAs are inspired by Darwin's theory about evolution. The basic concept of GAs is designed to simulate processes in natural system necessary for evolution, specifically those that follow the principles of survival of the fittest. As such they represent an intelligent exploitation of a random search within a defined search space to solve a problem.

The structure of a genetic algorithm consists of a simple iterative procedure on a population of genetically different individuals. The phenotypes are evaluated according to a predefined fitness function, the genotypes of the best individuals are copied several times and modified by genetic operators, and the newly obtained genotypes are inserted in the population in place of the old ones. This procedure is continued until a *good enough* solution is found [3].

In this work, the templates are the chromosomes, lines in a template are the genes, and a line having a particular slope represents the value (allele) that a gene has.

A good fitness means that a particular template (matrix L) fits better to the original RAPD image (matrix A). To evaluate a template, images corresponding to matrices A and L are put together, and a sum of intensities is obtained by considering neighborhood pixels within a range and for each line. This range is determined in this work considering the width of the brightest part of the lane. The aim of this is to gain precision in the fitness function. If a line in the template coincides with a lane, a higher value of the sum is obtained. In contrast, if they do not coincide, the value is lower than in the first case, because we are adding background pixel intensities (values close to zero).

Another issue added is that, the value obtained in the evaluation of each line is stored as part of the *gene*. In this way, the sum of intensities of pixels is only done when a new line is created; and this occurs in mutation and in tabu search. As a consequence of this issue, the execution time is reduced considerably compared to previous experiments.

Genetic operators: Different genetic operators were considered for this work. These genetic operators are briefly described below:

- Selection. Selection is accomplished by using the roulette wheel mechanism [3]. It means that individuals with a best fitness value will have a higher probability to be chosen as parents.

- Cross-over. Cross-over is used to exchange genetic material, allowing part of the genetic information of one individual to be combined with part of the genetic information of a different individual. For example, if we have two templates each containing $r+s$ lines, after cross-over, the generated children result in: children 1 will have the first r lines that correspond to template 1, and the following s lines that correspond to template 2. For children 2, the process is slightly different, in which the order the templates are considered is modified.
- Mutation. By using this genetic operator, a slight variation is introduced into the population so that a new genetic material is created. In this work, mutation is accomplished by randomly replacing, with a low probability, a particular line in a template.

Tabu search (TS) is a meta-heuristic that guides a local heuristic search procedure to explore the solution space beyond local optimality. The local procedure is a search that uses an operation called *move* to define the neighborhood of any given solution. One of the main components of TS is its use of adaptive memory, which creates a more flexible search behavior. In a few words, this procedure iteratively moves from a solution x to a solution x' in the neighborhood of x, until some stopping criterion has been satisfied. In order to explore regions of the search space that would be left unexplored by the local search procedure, tabu search modifies the neighborhood structure of each solution as the search progresses [4].

A solution is a template representing a RAPD image, let us say the x solution; then to move from a solution x to a solution x' means that the template is modified. To modify a template we have chosen two possibilities: the first one is the change in the value of the slope for one or more lines in the template, i.e., a rotation movement; the second one is a shifting movement, the line is moved to the left or to the right, without changing the value of the slope for that line. In other words, if we call x_{inf} and x_{sup} the bottom and top points in a line respectively, a rotation movement is realized by changing these points to $x_{inf} - \delta$ and $x_{sup} + \delta$ (or changing to $x_{inf} + \delta$ and $x_{sup} - \delta$). In an analogous way, the shifting movement is accomplished by changing the original points to $x_{inf} + \delta$ and $x_{sup} + \delta$ (changing to minus if the movement is towards the opposite side of the image). Figure 3 illustrates the rotation movement and the shifting movement. The values allowed in both, shifting and rotation movements, are gradually diminished to avoid dramatic changes in the quality of the solutions.

To avoid repeated movements during a certain bounded period of time, TS stores each movement in a temporal memory, which is called *tabu list.* Each element in the tabu list contains one lane and its corresponding movement. A particular lane in the list may occur more than once, but the associated movement needs to be different. In this work, the size of the tabu list is bounded by the number of lanes in the particular image under treatment.

When it is not possible to find a better solution in the neighborhood of x, it is used the, so-called, *aspiration criterion*, which allows to search in the tabu list for a movement that improves the current state x. If that movement doesn't

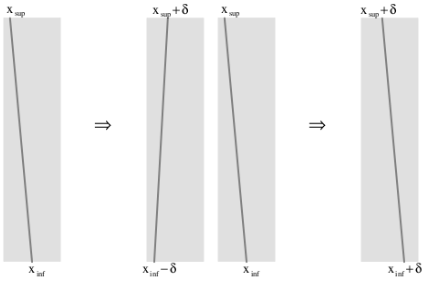


Fig. 3. The schema for rotating a line, and the schema for shifting a line

exist, then a movement with a higher residence time in the tabu list is chosen and, in this particular case, the inverse corresponding movement is applied to x; i.e., it is used to implement a backtracking strategy.

As previously mentioned, in spite of good results obtained by using genetic algorithms, the problem of local optimum is always present. By taking into account this issue, we decided to *hybridize* the procedure, that is to say to combine genetic algorithms with another strategy, in this specific work with tabu search, to let potential solutions avoid those local optimum points. The hybridization procedure considers the following strategy: the main objective of this process is to gradually improve individuals belonging to the population the genetic algorithm is working with. When the fitness measured during a certain number of iterations accomplished by the genetic algorithm doesn't vary; a reduced number of individuals is selected from current population. One of them is the best individual of the population, while the others are randomly selected. Each one of these individuals acts as an input for triggering a tabu search procedure.

Once the tabu search process is finished, the resulting individuals are reinserted into the genetic population, and the process continues with the genetic algorithm procedure, as before. The complete process is repeated several times depending on the quality of the genetic population and the stopping process condition. The latter is specified as a time condition (number of iterations) or as a specific fitness value.

4 Results

For testing, it is necessary to provide a set of images that consider most of the problematic situations: different slope in lanes, different bright, noise in images, missing lanes, and different number of lanes. According to this criteria, 13 images were chosen to carry out the final tests.

Parameters are variables that maintain a fixed value during a particular processing. While they cannot be defined *a priori*, they have to be experimentally determined. We carried out different tests for both, the genetic algorithm and the hybrid algorithm processing.

Table 1. Parameters for testing

Parameter	Value(s)
Population Size	150
Num. Generations	2000
Cross-over %	70
Mutation %	2
Elitism %	10
Seed Value	10 different values

Parameter	Value(s)
Population Size	20, 50
Num. Generations	2000
Cross-over %	70, 60
Mutation %	2
Elitism %	10
Seed Value	10 different values
Triggering Cond.	30 generations
Num. Iterations	30
Tabu List Size	Number of lanes

The set of parameters used for the genetic algorithm approach is summarized in the left part of Table 1; and the set of parameters used for the hybrid approach (genetic algorithm plus tabu search), is summarized in the right part of the table. Figure 4 illustrates fitness improvement for 13 testing images.

The elements that have an important influence on the execution time, are the number of lanes the image has, the population size and the mutation probability.

The execution time for the hybrid algorithm approach, is higher than the genetic approach, because in this case, each movement implies to evaluate one or more lines for each new template created in the neighborhood. This is one of the reasons why we reduced the population size during tests. An increasing population size needs a larger execution time. A reduced number of tests considering a higher number for the population size didn't produce better results.

As shown in Figure 4, in all cases, the hybrid algorithm produced a better fitness than the genetic algorithm. In most of the test cases, considering the two different population sizes, the values obtained are similar.

If we consider the evolution of fitness, for early generations (below 300 genetic generations), the genetic algorithm offers better results that those obtained by using the hybrid algorithm. This is likely due to the bigger population size. For a medium evolution period (between 300 and 500 genetic generations) the fitness values for the hybrid algorithm increase dramatically. In this particular group, fitness is similar in both, the genetic algorithm and the hybrid algorithm. When

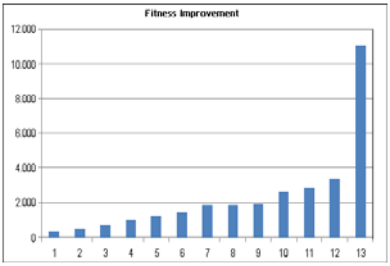


Fig. 4. Fitness improvement for different images

we consider a higher number of genetic generations (more that 500), values are clearly better when using the hybrid algorithm. On the other hand, as expected, the hybrid approach is much more time consuming than the genetic algorithm approach. Figure 5 shows the evolution while processing a particular image, the upper part (a) shows the original RAPD image, the middle part (b) shows the best individual, and the bottom part (c) shows the corrected image.

5 Analysis

The number of lanes and the population size, impact the execution time because of the increasing amount of data that needs to be processed. The mutation probability influence is related to the fact that the evaluation function evaluates a line each time a new line is created, the typical action when mutation is accomplished. In absence of mutation, each line keeps its evaluation while it is not modified.

The facts that the hybrid algorithm produced a better fitness and the similarity of results in spite of changing the population size, allow us to infer that the hybrid algorithm possibly reached an optimum value. For a particular image, the value obtained for the corresponding fitness is the same for the different population sizes and for the different seeds used for the random values that the genetic algorithm employs.

It is possible to say that when dealing with RAPD images, the hybrid algorithm works better than the genetic algorithm. One fact to remark is that this is true for both, large population sizes and small population sizes. This can be explained by considering that the hybrid approach acts from two different points

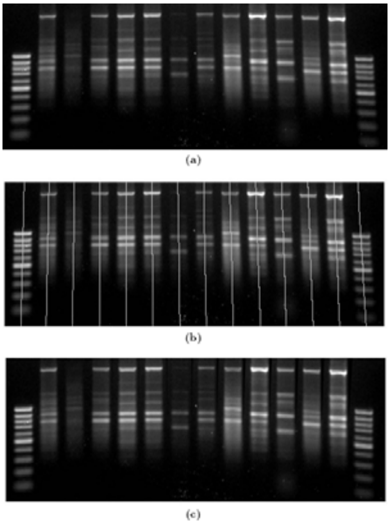


Fig. 5. The evolution in lane correction: a) the original image, b) the best individual, c) the corrected image

of view. From the first one, the genetic algorithm point of view acts as a global search process, while from the second point of view, tabu search acts as a local search process. Combining these two strategies leads to better results.

One hypothesis that can explain this fact is that the single genetic approach is good enough, and that there is no space for significant improvements concerning fitness. Despite the fact that the values of fitness have minor differences between both strategies (single and hybridized genetic algorithm), these differences are important for human experts, because they provide more reliability to their genetic conclusions.

The population size seems not to be important in terms of results. However there is a difference; a larger population size implies a higher genetic algorithm influence; on the contrary, a smaller population size reveals that the tabu search steps have a greater influence on the results. This is because the larger populations present a higher genetic variability, and genetic operations, consequently, possibly produce higher quality offsprings; while little size populations depend on tabu search movements quality to improve individuals, which tend to converge to local minimum values.

6 Conclusions

Experiments show that by using genetic algorithms and tabu search the final fitness value is slightly improved.

Another issue to remark is that the best performance for the hybrid algorithm is obtained after considering 500 genetic iterations. We believe that the reason behind this behavior is that at the beginning, the genetic algorithm can improve the population due to genetic operations, and so, the tabu search is not triggered until a local stability population is reached.

Results are slightly better with the hybrid approach, but it is necessary to pay a cost; that is the increasing execution time; in fact, the hybrid approach that considers 50 individuals takes an execution time similar to the single genetic approach that considers 150 individuals.

Some future work directions are mentioned in the following: the implemented solution depends strongly on a correct lane boundary detection. So, it is necessary to have a better lane detection method, probably automated. Currently, multilevel thresholding is being considered for band detection, but it could be appropriate for that purpose, as well. To increase the degree of parallelism when dealing with tabu search and to increase the population size are some pending issues that need to be taken into account.

Acknowledgment

This work has been partially supported by grant DIUC 209.093.014-1.0, University of Concepción, Chile.

References

1. Cao, W., Scoles, G., Hucl, P., Chibbar, R.: Philogenetic Relationships of Five Morphological Group of Hexaploid wheat Based on RAPD Analysis. Genome. 43, 724–727 (2000)
2. Casasoli, M., Mattioni, C., Cherubini, M., Villani, F. A Genetic Linkage Map of European Chestnut (Castanea Sativa Mill.) Nased on RAPD, ISSR and Isozime Markers. Theoretical Applied Genetics 102, 1190–1199 (2001)
3. Floreano, D., Mattiussi, C.: Bio-Inspired Artificial Intelligence. Theories, Methods, and Technologies. MIT Press, Cambridge (2008)
4. Glover, F., Laguna, M.: Tabu Search. Springer, Heidelberg (1997)
5. Groos, C., Gay, G., Perrenant, M., Gervais, L., Bernard, M., Dedryver., F., Charmet, G.: Study of the Relationships Between Pre-harvest Sprouting and Grain Color by Quantitative Trait Loci Analysis in the White X Red Grain Bread-wheat Cross. Theoretical Applied Genetics 104, 39–47 (2002)
6. Herrera, R., Cares, V., Wilkinson, M., Caligarip, D.: Characterization of Genetic Variations Between Vitis vinifera Cultivars from Central Chile Using RAPD and Inter Simple Sequence Repeat Markers. Euphytica 124, 139–145 (2002)
7. Pinninghoff, M.A., Contreras, R., Rueda, L.: An evolutionary approach for correcting random amplified polymorphism DNA images. In: Mira, J., Ferrández, J.M., Álvarez, J.R., de la Paz, F., Toledo, F.J. (eds.) IWINAC 2009. LNCS, vol. 5602, pp. 469–477. Springer, Heidelberg (2009)
8. Rueda, L., Uyarte, O., Valenzuela, S., Rodriguez, J.: Processing Random Amplified Polymorphism DNA Images Using the Radon Transform and Mathematical Morphology. In: Kamel, M.S., Campilho, A. (eds.) ICIAR 2007. LNCS, vol. 4633, pp. 1071–1081. Springer, Heidelberg (2007)
9. Saal, B., Struss, D.: RGA-and RAPD-derived SCAR Markers for a Brassica B-Genome Introgression Conferring Resistance to Blackleg Oil Seed in Oil Seed Rape. Theoretical Applied Genetics 111, 281–290 (2005)
10. Sudapak, M., Akkaya, M., Kence, A.: Analysis of Genetic Relationships Among Perennial and Annual Cicer Species Growing in Turkey Using RAPD Markers. Theoretical Applied Genetics 105, 1220–1228 (2002)
11. Tripathi, S., Mathish, N., Gurumurthi, K.: Use of Genetic Markers in the Management of Micropropagated Eucalyptus Germplasm. New Forests 31, 361–372 (2006)

Adaptive Algorithm-Based Fused Bayesian Maximum Entropy-Variational Analysis Methods for Enhanced Radar Imaging

R.F. Vázquez-Bautista[1], L.J. Morales-Mendoza[1], R. Ortega-Almanza[1], and A. Blanco-Ortega[2]

[1] FIEC-Universidad Veracruzana
Av. Venustiano Carranza s/n, Colonia Revolución, C.P. 93390, Poza Rica, Ver
{favazquez,javmorales,raortega}@uv.mx
[2] CENIDET-Ingeniería Mecatrónica
Cuernavaca, Morelos, C.P. 62490
andres.blanco@cenidet.edu.mx

Abstract. In this paper we address an adaptive computational algorithm to improve the Bayesian maximum entropy–variational analysis (BMEVA) performance for high resolution radar imaging and denoising. Furthermore, the variational analysis (VA) approach is aggregated by imposing the metrics structures in the corresponding signal spaces. Then, the formalism for combining the Bayesian maximum entropy strategy with the VA paradigm is presented. Finally, the image enhancement and denoising benefits produced by the proposed Adaptive Bayesian maximum entropy–variational analysis (ABMEVA) method are showed via simulations with real-world radar scene

Keywords: Bayesian Maximum Entropy, data fusion, adaptive algorithm, variational analysis.

1 Introduction

Recently, the Bayesian maximum entropy (BME) method was developed in [2], where the Bayesian estimation method for high resolution radar image formation [1], [13], [15] employs the maximum entropy (ME) information theoretical-based windowing of the resulting images. Moreover, the alternative approach to radar image enhancement and denoising was proposed in [3] where the variational analysis (VA) paradigm was applied in [4], [14] to control the image gradient flow over the sensing scene using the difference-form approximations of the partial differential equations (PDE) to formalize different image processing problems including image segmentation, enhancement and denoising [3], [7], [11]. This strategy is adapted for a particular remote sensing system model and robust a priori information about the noise's statistics and the desired image. The BME is associated in [6], [7],[10] with the spatial spectrum pattern (SSP) of the wavefield backscattered from the probing surface. As the SSP represents the power distribution in the RS environment, the power non-negativity constraint is incorporated implicitly in the BME strategy but that do not

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 154–163, 2010.

incorporate specific VA geometrical properties of the image, e.g. its gradient flow over the scene/frame [14]. However, the so called data fusion [12] strategy is a useful way to aggregate the BME statistical approach to optimize the VA-based radar image enhancement using a priori information [8]. The fused BMEVA approach [8] is presented and developed into a robust formalism without an efficient computational performance [10]. However, in recent years, several efforts have been directed toward the high performance computing because the computational load and implementation represents critical problem [16]. Based on the previous thing, the following problem arises: how to include the aggregate the BMEVA method for performing the combined statistical-descriptive enhancement of the radar images based on adaptive algorithm?

2 Inverse Problem Statement

Based on [1], [2], we define the model of the observation wavefield u by specifying the stochastic equation of observation of an operator form $u = Se + n; e \in \mathbf{E}$; $u, n \in \mathbf{U}$; S**:** E $\rightarrow$ U**,** in Hilbert signal spaces E and U with the metrics structures induced by the inner products, $[u_1, u_2]_\mathrm{U}$ and $[e_1, e_2]_\mathrm{E}$, respectively, where the Gaussian zero-mean random fields e, n, and u correspond to the initial scattered field, noise and observation wavefield, respectively. Now, recalling the experiment design (ED) theory-based projection formalism [2], one can proceed from the operator form equation of observation to its finite-dimensional vector form,

$$\mathbf{U} = \mathbf{S}\,\mathbf{E} + \mathbf{N}\,. \tag{1}$$

In which **E**, **N**, and **U** represent the zero-mean Gaussian vectors with the correlation matrices $\mathbf{R}_\mathrm{E} = \mathbf{R}_\mathrm{E}(\mathbf{B}\ )=$ diag{**B**}, $\mathbf{R}_\mathrm{N}$,and $\mathbf{R}_\mathrm{U} = \mathbf{S}\mathbf{R}_\mathrm{E}\mathbf{S}^+ + \mathbf{R}_\mathrm{N}$, respectively, where diag{**B**} is a diagonal K-by-K matrix with elements $B_k = <E_kE_k^*>$ and $<\cdot>$ defines the statistical averaging operator. Vector **B** is referred to as the spatial spectrum pattern (SSP) vector that represents the average brightness image of the remotely sensed scene, and matrix **S** defines the signal formation operator (SFO).

3 Bayesian Maximum Entropy and Variational Analysis

3.1 BME Analysis

The processing of the observation data **U** is used to obtain an unique and stable estimate $\hat{\mathbf{B}}$. However, because of the ill-posed nature of such the image reconstruction problem [2] the SFO, in general, is ill-conditioned or even singular. The ME principle [2], [5] provides the well-grounded way to alleviate the problem ill-poseness. Based on the ME paradigm, the whole image is viewed as a composition of a great amount of elementary discrete (speckles or pixels) with the elementary “pixel brightness”. Following the ME approach in [9], the a priori pdf $p(\mathbf{B})$ of the desired image is defined via maximization the entropy of the image probability that satisfies also the constraints imposed by the prior knowledge [2]. The vector **B** is viewed as an element of the K-D vector space $\mathrm{B}_{(K)} \ni \mathbf{B}$ with the squared norm imposed by the inner product $\|\mathbf{B}\|^2_{B(K)} = [\mathbf{B},\mathbf{MB}]$, where $\mathbf{M}$ is the positive definite metrics inducing matrix [2]. In addition, the physical factors of the experiment can be generalized imposing the physically obvious constraint bounding the average squared norm of the SSP [2],

$$\int_{\mathbf{B}_C} [\mathbf{B},\mathbf{MB}]p(\mathbf{B})d\mathbf{B} \le c_0 \ . \tag{2}$$

Thus, the a priori pdf $p(\mathbf{B})$ is to be found as a solution to the Lagrange maximization problem, with the Lagrange multipliers α, and λ. Such a problem is specified as follows

$$\begin{aligned} -\int_{\mathbf{B}_C} \ln p(\mathbf{B})p(\mathbf{B})d\mathbf{B} - \alpha\left(\int_{\mathbf{B}_C} [\mathbf{B},\mathbf{MB}]p(\mathbf{B})d\mathbf{B} - c_0\right) \\ - \lambda\left(\int_{\mathbf{B}_C} p(\mathbf{B})d\mathbf{B} - 1\right) \to \max_{p(\mathbf{B})} \end{aligned} , \tag{3}$$

for $\mathbf{B} \in B_C$, and $p(\mathbf{B}) = 0$ otherwise. The solution to (3) was derived in [2] that yields the Gibbs-type a priori pdf

$$p(\mathbf{B} \mid \alpha) = \exp\{-\ln \sum(\alpha) - \alpha[\mathbf{B},\mathbf{MB}]\} \ , \tag{4}$$

where $\sum(\alpha)$ represents the so-called Boltzmann statistical sum [2]. The log-likelihood [2] of the vector $\mathbf{B}$ is defined as

$$\begin{aligned} \Lambda(\mathbf{B} \mid \mathbf{U}) = \ln p(\mathbf{B} \mid \mathbf{U}) = -\ln \det\{\mathbf{SD}(\mathbf{B})\mathbf{S}^+ + \mathbf{R_N}\} \\ -[\mathbf{U},(\mathbf{SD}(\mathbf{B})\mathbf{S}^+ + \mathbf{R_N})^{-1}\mathbf{U}] \end{aligned} , \tag{5}$$

and BME strategy for image reconstruction (estimation of the SSP vector $\mathbf{B}$) is stated as follows,

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B},\alpha}\{-\Lambda(\mathbf{B} \mid \mathbf{U}) - \ln p(\mathbf{B} \mid \alpha)\} \ . \tag{6}$$

The BME estimate of the SSP is a solution to the problem (6) and is given by the nonlinear equation [2]

$$\hat{\mathbf{B}} = \mathbf{W}(\hat{\mathbf{B}})[\mathbf{V}(\hat{\mathbf{B}}) - \mathbf{Z}(\hat{\mathbf{B}})] \ . \tag{7}$$

Here, $\mathbf{V}(\hat{\mathbf{B}}) = \{\mathbf{F}(\hat{\mathbf{B}})\mathbf{UU}^+\mathbf{F}^+(\hat{\mathbf{B}})\}_{\mathrm{diag}}$ is a vector that has the statistical meaning of a sufficient statistics (SS) for the SSP estimator, operator $\mathbf{F}(\hat{\mathbf{B}}) = \mathbf{D}(\hat{\mathbf{B}})(\mathbf{I}+\mathbf{S}^+\mathbf{R}^{-1}{}_{\mathbf{N}}\mathbf{SD}(\hat{\mathbf{B}}))^{-1}\mathbf{S}^+\mathbf{R}^{-1}{}_{\mathbf{N}}$ is referred to as the SS formation operator, the vector $\mathbf{Z}(\hat{\mathbf{B}}) = \{\mathbf{F}(\hat{\mathbf{B}})\mathbf{R_N}\mathbf{F}^+(\hat{\mathbf{B}})\}_{\mathrm{diag}}$ is the shift or bias vector, and $\mathbf{W}(\hat{\mathbf{B}}) = (\mathbf{T}(\hat{\mathbf{B}})+2\hat{\alpha}\mathbf{D}^2(\hat{\mathbf{B}})\mathbf{M})^{-1}$ has the statistical meaning of a solution dependent (i.e., adaptive) window operator with the stabilizer $\mathbf{T}(\hat{\mathbf{B}}) = \mathrm{diag}\{\{\mathbf{S}^+\mathbf{F}^+(\hat{\mathbf{B}})\mathbf{F}(\hat{\mathbf{B}})\mathbf{S}\}_{\mathrm{diag}}\}$. Adaptation is to be performed by both the current SSP estimate, $\hat{\mathbf{B}}$, and the estimate of the normalizing constant $\hat{\alpha}$ determined from a solution of the equation $-\partial/\partial\alpha(\ln \Sigma(\alpha)) = [\hat{\mathbf{B}},\mathbf{M}\hat{\mathbf{B}}]$.

3.2 Variational Analysis

Based on the anisotropic diffusion in [4], the idea is to evolve from an original image $B(x, y)$, defined in a convex domain Ω, to a family of increasingly smooth images $B(x, y, t)$ derived from the solution. The diffusion algorithm reduces the noise from an image by modifying the image via a partial differential equation (PDE). Now, looking for a statistical interpretation of the Perona-Malik [4] anisotropic diffusion equation it is possible to make the robustification of the VA-based image model. The generalized robustified VA energy function is defined as [3]

$$VA(\mathbf{B}) = \int_{\Omega} \rho(\| \nabla \mathbf{B} \|) d\Omega \ . \tag{8}$$

over the image domain Ω, where

$$\rho(x) = \int g(x) x \, dx = \sigma^2 \log\left[1 + \frac{1}{2}\left(x^2/\sigma^2\right)\right] \ . \tag{9}$$

is the Lorentz function, and

$$g(x) = \rho'(x)/x \ . \tag{10}$$

is the auxiliary function that defines the relation between the different reconstructed images after applying the robust VA estimation method [4].

The VA approach assumes the minimization of (8) via gradient descendent flow using the calculus of variations as follows,

$$VA(\hat{\mathbf{B}}) = \arg\min_{\mathbf{B}} \int_{\Omega} \rho(\| \nabla \mathbf{B} \|) d\Omega \ . \tag{11}$$

In such the VA approach, the critical issue is the choice of the variational functional. Recall that in this study we follow the Lagrangian model given by (8).

4 Fused BMEVA Method

Now, the fused BMEVA method for image reconstruction presented in [8], [10] combine the VA and BME approaches and the formalism used the following strategies

$$\hat{\mathbf{B}} = \arg\min_{\mathbf{B},\alpha}\{-\Lambda(\mathbf{B} \mid \mathbf{U}) - \ln p(\mathbf{B} \mid \alpha)\} \rightarrow \text{Optimal BME} \ , \tag{12}$$

$$VA(\hat{\mathbf{B}}) = \arg\min_{\mathbf{B}} \int_{\Omega} \rho(\| \nabla \mathbf{B} \|) d\Omega \ \rightarrow \ \text{Optimal VA} \ . \tag{13}$$

It is important to understand that both the BME and VA approaches look for an enhanced reconstruction with edge preservation. Henceforth, the proposed fused BMEVA reconstruction strategy assumes the solution to the variational problem

$$\hat{\mathbf{B}}_{BMEVA} = \arg\min_{\mathbf{B},\alpha} \{-\Lambda(\mathbf{B} \mid \mathbf{U}) - \ln \ p(\mathbf{B} \mid \alpha) + \int_{\Omega} \rho(\| \nabla \mathbf{B} \|) d\Omega\} \tag{14}$$

The logarithm series expression is a viable mathematical tool to obtain the numerical approximation from the energy function by series. Recalling [8], the conventional gradient method [4] is used to solve (14), but in this case n=3 to minimize the third term on the second member in (14), as following,

$$\sigma^2 \log\left[1+\left(\frac{\|\nabla \mathbf{B}\|}{\sigma\sqrt{2}}\right)^2\right] = \sigma^2 \sum_{n=1}^{2} \frac{(-1)^{n+1}\left(\frac{\|\nabla \mathbf{B}\|}{\sigma\sqrt{2}}\right)^{2n}}{n} = \mathbf{B}^{+}\mathbf{Q}\mathbf{B} \ , \tag{15}$$

where

$$\mathbf{Q} = (1/2)\mathbf{L} + \tau_1\,\mathbf{LL} + \tau_2\mathbf{LLL} \tag{16}$$

is the composed weighting matrix and the regularization parameters: τ_1=-1/8σ^2 and τ_2=1/24σ^4, respectively. The matrix **L** represents the numerical approximation of the Laplacian operator [7].

Now, the solution to problem (14) can be expressed in a form of the nonlinear equation

$$\begin{aligned} F(\mathbf{B},\alpha) = \ln\det\{\mathbf{SD}(\mathbf{B})\mathbf{S}^{+} + \mathbf{R}_{\mathrm{N}}\} + [\mathbf{U}, (\mathbf{SD}(\mathbf{B})\mathbf{S}^{+} + \mathbf{R}_{\mathrm{N}})^{-1}\mathbf{U}] \\ + \ln\sum(\alpha) + \alpha[\mathbf{B}, \mathbf{MB}] + [\mathbf{B}, \mathbf{QB}] \ . \end{aligned} \tag{17}$$

$$\partial F(\mathbf{B},\alpha)/\partial\mathbf{B} = 0 \qquad \text{and} \qquad \partial F(\mathbf{B},\alpha)/\partial\alpha = 0 \, . \tag{18}$$

As it was detailed in [2], because of the nonlinearity, no unique regular method for solving (18) exists; however, one can represent the solution in a form convenient for the further analysis. To proceed in this direction, we follow the methodology proposed in [2] which yields the equation

$$\mathbf{TB} + \mathbf{Z} - \mathbf{V} + 2\alpha\mathbf{D}^2\mathbf{MB} + 2\mathbf{QB} = 0 \ . \tag{19}$$

Grouping the terms and replacing $\alpha \rightarrow \hat{\alpha}$ from (18) we obtain

$$\mathbf{TB} + \mathbf{Z} - \mathbf{V} + 2\alpha\mathbf{D}^2\mathbf{MB} + 2\mathbf{QB} = 0 \ . \tag{20}$$

Finally, solving (20) with respect to **B** and exposing the dependence of **T**(**B**), **D**(**B**), **V**(**B**)**,** and **Z**(**B**) on the solution we obtain

$$\hat{\mathbf{B}} = \mathbf{W}(\hat{\mathbf{B}})[\mathbf{V}(\hat{\mathbf{B}}) - \mathbf{Z}(\hat{\mathbf{B}})] \ , \tag{21}$$

where

$$\mathbf{W}(\hat{\mathbf{B}}) = (\mathbf{T}(\hat{\mathbf{B}}) + 2\hat{\alpha}\mathbf{D}^2(\hat{\mathbf{B}})\mathbf{M} + 2\mathbf{Q})^{-1} \ . \tag{22}$$

represents the adaptive spatial window operator.

5 The Adaptive Computational Algorithm

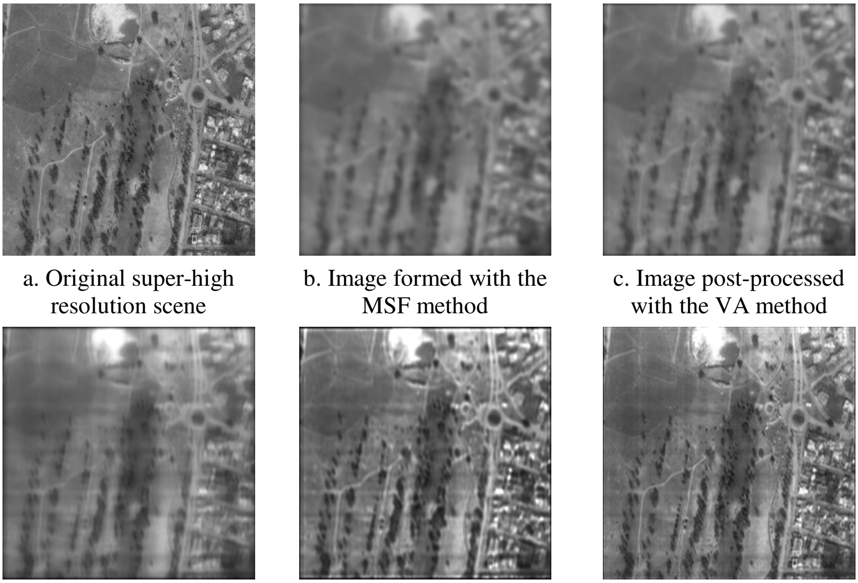
The derived BMEVA estimator (21) can be converted into an efficient iterative algorithm using the LMS iteration method [11]. Pursuing such the approach [11], we refer to the SSP estimate on the right-hand side in (21) as the current estimate $\hat{\mathbf{B}}^{(t)}$ at the *t*th iteration step, and associate the entire right-hand side of (21) with the rule for forming the estimate $\hat{\mathbf{B}}^{(t+1)}$ for the next iteration step (t+1) that yields

$$\hat{\mathbf{B}}^{(t+1)} = \hat{\mathbf{B}}^{(t)} + \gamma[\hat{\mathbf{B}}^{(t+1)} - \hat{\mathbf{B}}^{(t)}]. \tag{23}$$

Due to the performed regularized windowing (20), the iterative algorithm (23) converges in a polynomial time [8] regardless of the choice of the balance factor γ within the prescribed normalization interval, $0 \le \gamma \le 1$.

6 Simulations and Concluding Remarks

Now, the imagery results obtained by the new adaptive computational algorithm are enough qualitative evidence to discuss the ABMEVA computational efficiency. Figure 1 shows the original 256-by-256 simulated scene **B** and the computed results using the Match Spatial Filter (MSF), VA, BME , BMEVA and the ABMEVA algorithms for the exp(-bx^2) of the point spread functions (PSF), i.e. system 1: Gaussian “Bell”-type PSF. Moreover, the Figure 2 shows the results of the image enhancement



a. Original super-high resolution scene

b. Image formed with the MSF method

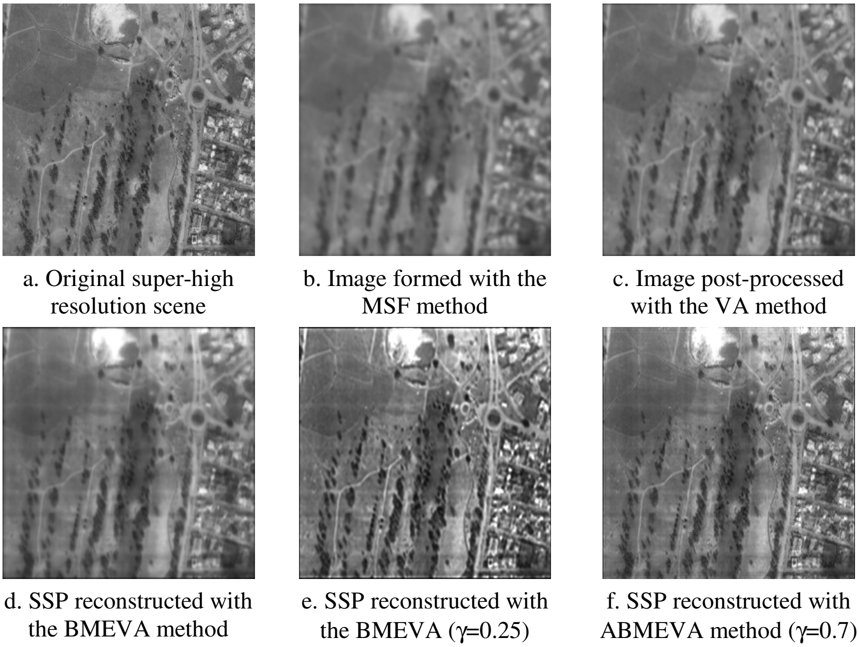
c. Image post-processed with the VA method

d. SSP reconstructed with the BMEVA method

e. SSP reconstructed with the ABMEVA (γ=0.6)

f. SSP reconstructed with ABMEVA method (γ=0.9)

Fig. 1. Simulation results for **B** scene (first system model)



a. Original super-high resolution scene

b. Image formed with the MSF method

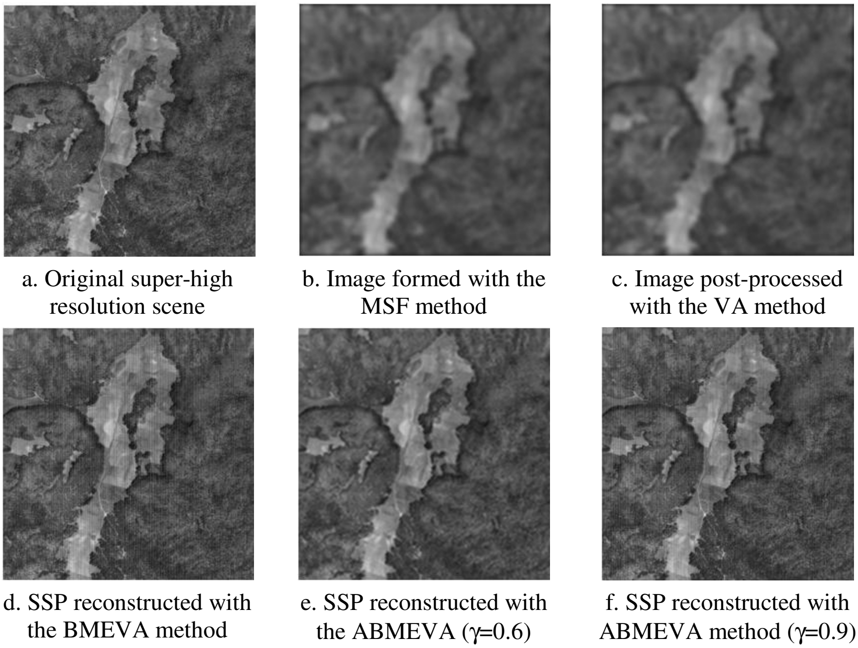
c. Image post-processed with the VA method

d. SSP reconstructed with the BMEVA method

e. SSP reconstructed with the BMEVA (γ=0.25)

f. SSP reconstructed with ABMEVA method (γ=0.7)

Fig. 2. Simulation results for **B** scene (second system model)



a. Original super-high resolution scene

b. Image formed with the MSF method

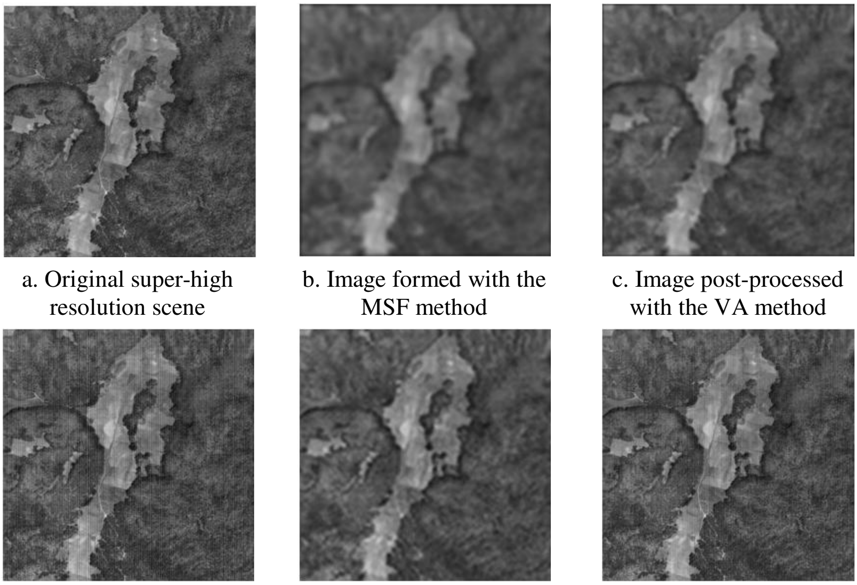
c. Image post-processed with the VA method

d. SSP reconstructed with the BMEVA method

e. SSP reconstructed with the ABMEVA (γ=0.6)

f. SSP reconstructed with ABMEVA method (γ=0.9)

Fig. 3. Simulation results for **B’** scene (first system model)



a. Original super-high resolution scene

b. Image formed with the MSF method

c. Image post-processed with the VA method

d. SSP reconstructed with the BMEVA method

e. SSP reconstructed with the BMEVA (γ=0.25)

f. SSP reconstructed with ABMEVA method (γ=0.7)

Fig. 4. Simulation results for **B'** scene (second system model)

Table 1. Scene **B**-based results for two different simulated SAR systems

SNR [dB]	*IOSNR* [dB] System 1 Reconstruction Method				*IOSNR* [dB] System 2 Reconstruction Method			
μ	*VA*	*BMEVA*	*ABMEVA* (γ=0.6)	*ABMEVA* (γ=0.9)	*VA*	*BMEVA*	*ABMEVA* (γ=0.25)	*ABMEVA* (γ=0.7)
10	0.635	2.120	8.301	9.711	1.335	3.025	11.926	13.575
15	0.638	2.625	8.322	9.753	1.351	3.128	11.941	13.581
20	0.638	3.142	8.354	9.773	1.356	4.335	11.978	13.590
25	0.640	4.430	8.369	9.781	1.358	5.498	11.982	13.603
30	0.642	4.732	8.375	9.800	1.360	6.133	11.993	13.615

using the sin(*ax*)/*ax*, i.e. system 2: sinc-type PSF. On the other hand, the Table 1 manifests the strong quantitative analysis to valid the computational performance based on input-output signal-noise ratio (IOSNR) metric [2]. Following this way, the Table 2, Figure 3, and Figure 4 show the results based on simulated scene **B'** to match the quantitative and qualitative analysis.

In summary, we may conclude that the proposed ABMEVA method provides the substantially improved image enhancement and reconstruction achieved due to performing the adaptive windowing in the flat regions with preserving the edge features. The new approach presents a new computational technique to improve the BMEVA

Table 2. Scene **B'**-based results for two different simulated SAR systems

SNR [dB]	*IOSNR* [dB] System 1 Reconstruction Method				*IOSNR* [dB] System 2 Reconstruction Method			
μ	*VA*	*BMEVA*	*ABMEVA* (γ=0.6)	*ABMEVA* (γ=0.9)	*VA*	*BMEVA*	*ABMEVA* (γ=0.25)	*ABMEVA* (γ=0.7)
10	1.020	12.454	7.393	10.268	0.724	3.948	6.293	10.382
15	1.132	12.625	7.444	10.465	0.751	4.339	6.445	10.662
20	1.129	12.957	7.834	10.628	0.779	4.613	6.608	10.873
25	1.166	13.032	8.102	10.831	0.822	4.903	6.868	11.241
30	1.171	13.073	8.123	10.870	0.837	5.249	7.017	11.388

performance. A key distinguish feature of the ABMEVA method is that the problem of image enhancement and denoising is solved in the framework of Bayesian estimation theory that incorporates the VA considerations through the fused reconstruction strategy. Finally, the proposed ABMEVA manifest a robust performance for different formation systems-based radar imaging.

References

1. Skolnic, M.I. (ed.): Radar Handbook, 2nd edn. McGraw-Hill, Boston (1990)
2. Shkvarko, Y.V.: Estimation of Wavefield Power Distribution in the Remotely Sensed Environment: Bayesian Maximum Entropy Approach. IEEE Trans. on Signal Processing 50(9), 2333–2346 (2002)
3. Ben Hamza, A., Krim, H., Unal, G.sB.: Unifying Probabilistic and Variational Estimation. IEEE Signal Processing Magazine 19, 37–47 (2002)
4. Black, M., Sapiro, G., Marimont, D.H., Hegger, D.: Robust Anisotropic Diffusion. IEEE Trans. Image Processing 7(3), 421–432 (1998)
5. Khuong Nguyen, M., Mohammad-Djafari, A.: Bayesian Approach with the Maximum Entropy Principle in Image Reconstruction from Microwave Scattered Field Data. IEEE Trans. Medical Imaging 13, 2 (1994)
6. Shkvarko, Y.V.: Estimation of Wavefield Power Distribution in the Remotely Sensed Environment: Bayesian Maximum Entropy Approach. IEEE Transactions on Signal Processing 50, 2333–2346 (2002)
7. Shkvarko, Y.V.: Unifying Regularization and Bayesian Estimation Methods for Enhanced Imaging with Remotely Sensed Data. Part I – Theory. IEEE Transactions on Geoscience and Remote Sensing 42, 923–931 (2004)
8. Vazquez-Bautista, R.F., Morales-Mendoza, L.J., Shkvarko, Y.V.: Aggregating the Statistical Estimation and Variational Analysis Methods in Radar Imagery. In: IEEE International Geoscience and Remote Sensing Symposium, IGARSS, Toulouse, France, vol. 3, pp. 2008–2010 (2003)
9. Morales-Mendoza, L.J., Vazquez-Bautista, R.F., Shkvarko, Y.V.: Unifying the Maximum Entropy and Variational Analysis Regularization Methods for Reconstruction of the Remote Sensing Imagery. IEEE Latin America Transactions 3, 60–73 (2005)
10. Shkvarko, Y., Vazquez-Bautista, R., Villalon-Turrubiates, I.E.: Fusion of Bayesian Maximum Entropy Spectral Estimation and Variational Analysis Methods for Enhanced Radar Imaging. In: Blanc-Talon, J., Philips, W., Popescu, D., Scheunders, P. (eds.) ACIVS 2007. LNCS, vol. 4678, pp. 109–120. Springer, Heidelberg (2007)

11. Cichocki, A., Amari, S.-i.: Adaptive Blind Signal and Image Processing. John Wiley & Sons, England (2002)
12. Solberg, A.H.S.: Data Fusion for Remote-Sensing Applications. In: Chen, C.H. (ed.) Signal and Image Processing for Remote Sensing, pp. 515–537. CRC Press, Boca Raton (2007)
13. Nadine, M.: Minimum Variance. In: Castanié, F. (ed.) Spectral Analysis: Parametric and Non-Parametric Digital Methods, ISTE USA, 1st edn., pp. 175–211 (2006)
14. Likas, A., Galatsanos, N.: Bayesian Methods based on Variational approximation for Blind Image Deconvolution. In: Campisi, P., Egiazarian, K. (eds.) Blind Image Deconvolution:Theory and Applications, pp. 141–168. CRC Press, Boca Raton (2007)
15. Joshi, M., Jalobeanu, A.: MAP Estimation for Multiresolution Fusion in Remotely Sensed Images Using an IGMRF Prior Model. IEEE Transactions on Geoscience and Remote Sensing 48(3), 1245–1255 (2010)
16. Plaza, A., Chang, C.I.: High-Performance Computer Architectures for Remote Sensing Data Analysis: Overview and Case Study. In: Plaza, A., Chang, C.I. (eds.) High Performance Computing in Remote Sensing, USA, pp. 9–41. Chapman & Hall/CRC (2008)

Iris Segmentation Using a Statistical Approach

Luis M. Zamudio-Fuentes[1], Mireya S. García-Vázquez[1],
and Alejandro A. Ramírez-Acosta[2]

[1] Centro de Investigación y Desarrollo de Tecnología Digital (CITEDI-IPN),
Avenida del Parque 1310, Tijuana, B.C. México 22510
[2] MIRAL. R&D, 1047 Palm Garden, Imperial Beach, 91932 USA
{zamudio,mgarciav}@citedi.mx, ramacos10@hotmail.com

Abstract. Eyelashes and reflections occluding the iris region are noise factors that degrade the performance of iris recognition. If these factors are not eliminated in iris segmentation phase, they are incorrectly considered as the iris region. Thus, produce false iris pattern information which decreases the recognition rate. In this paper a statistical approach is used to improve iris segmentation phase eliminating this noise from none constrain images, which is composed in three parts, finding the pupil and limbus boundary, reflection detection and eyelash detection. First an edge map is calculated using canny filter then the Circular Hough Transform is used to improve circle parameter finding. An intensity variation analysis is use to recognize a strong reflection. Eyelashes are classified in two categories, separable and multiple. Intensity variances are used to detect multiple eyelashes and an edge detector to localize separable eyelashes. The results show that statistics are useful to decide when is necessary applied the eyelash detector.

Keywords: Iris recognition, biometric, segmentation, eyelash detector.

1 Introduction

Iris recognition, is the most reliable biometric in terms of recognition and identification performance [1]. However, the performance of these systems is affected by inaccuracy segmentation [2, 3]. Indeed, the false iris region information decreases the recognition rate [4, 5 and 6]. It is therefore importance to identify the source of noise such as eyelashes and reflections for improving the quality of the segmentation and then the performance of the iris recognition. In the previous iris segmentation approaches [7, 8] this noise is not considered, just the inner and outer boundary of an iris is founded. The aim of this paper is to improve iris segmentation method using a statistical approach and decide when eyelashes and reflection detection is necessary to remove these noise from the image and improve the accuracy of the iris recognition.

The remainder of this paper is organized as follows. Section 2 reviews the reflection detection; in section 3 and 4 discuss inner and outer boundary detection of an iris and the eyelash detection model, respectively. Implementation, results and discussion are presented in section 5. Finally, in section 6, we draw discussion and give suggestions for future work.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 164–170, 2010.

2 Reflection Detection

Iris recognition system needs to avoid reflection that could degrade the recognition performance. However, strong reflection may be present if the illumination is not adapted or the subject uses contact lenses, also some jewelry can reflect in to the eye too. In this case, the value of intensity of a pixel with a strong reflection should be larger than a certain threshold [6]. A strong reflection can be recognized by a simple inequality, $f(x,y)>K_1$, where f(x,y) is a pixel in an image and K_1 is taken as 180, used in the following experiment after an histogram analysis of the grey scale intensity of the pixels.

3 Inner and Outer Boundary Detection of an Iris

Eye image contains pupil, iris, eyelids, eyelashes, sclera regions. However, for iris recognition the only area of interest is the iris region which has the patterns that are reported to remain unchanged over a life time [1], and they cannot be easily forged or modified. These patterns are delimited by the inner and outer boundary. The inner boundary is the circle which delimits de pupil and the iris region. On the other hand, outer boundary delimits the iris and the sclera. These two circles can be taken as two non concentric circles [9]. To define the each boundary and extract the iris region; the first step is to conduct edge detection to get the edge map of the eye images using a canny filter. The second step is to perform the circular Hough transform (CTH) [10]. The CTH is a "voting based" computational algorithm. For each boundary is necessary to obtain the coordinates of the center of the circle and the radius length. The CHT is used to transform a set of feature points in the image space into a set of accumulated votes in a parameter space. Then, for each feature point, votes are accumulated in an accumulator array for all parameter combinations. The array elements that contain the highest number of votes indicate the radius length and the coordinates of the center. The Circular Hough Transform [11, 12] has been implemented as follows:

1. An binarized edge map is calculated using a canny filter
2. Count each pixel in an edge map and obtain its position (x_i,y_i), where i is the total pixels in an edge map.
3. Set the radius range to find the boundary of interest.
4. Compute Circular Hough Transform.

```
   for pixeledge=1 to i.
   for r=1 to maximum radius wanted
   for y=1 to maximum row in an edge map
    Compute $x_k = \sqrt{r_j^2 - (y - y_i)^2} + x_i$
```

x_k represents the possible coordinate x of the center of the circle, r is the radius, y is the row in an edge map, y_i and x_i are the positions of a pixel in an edge map.

After x_k is calculated the accumulator array is increased as follows $ACC(y,x_k,r_j)=ACC(y,x_k,r_j)+1$
end all loops.

5. Obtain the maximum value in the accumulator array to get (x, y) coordinates which belong to the center and radius length r of the circle of interest.
6. Repeat from step 3 and modify the radius range to obtain the outer boundary

This algorithm requires a radius range where the radius of interest could be located and the position of the pixels which belong to an edge map. First, the inner boundary is found; secondly, the algorithm finds the outer boundary. For each position in an edge map this algorithm generates a circle for each value in the radius range. The point where the majority of the circles intersect, that point will be the center of the boundary of interest.

4 Eyelash Segmentation

Two classes of eyelashes are defined, separable and multiple eyelashes. Separable eyelashes are defined as the eyelashes that can be distinguished from other eyelashes and multiple eyelashes are the eyelashes that overlap in a small area [6].

4.1 Separable Eyelashes

Separable eyelashes can be distinguished from other eyelashes [4]; the pixels around separable eyelashes should not belong to others. Because of the intensity difference between iris pixels and eyelash pixels, a separable eyelash can be regarded as an edge in an image. Base on this property, a real part of Gabor Filter [13, 14, 15] is proposed to detect separable eyelashes, which, is in the spatial domain has the following general form.

$$G(x,u,\sigma) = e^{x^2/2\sigma^2}\cos(2\pi ux), \tag{1}$$

Where the frequency of the sinusoidal wave is $u,$ the standard deviation of the Gaussian envelope is σ and the parametric component which represent the real part of the Gabor filter is x. In fact, the filter works as an edge detector. If the resultant value of a point is smaller than a threshold, it is noted that this point belong an eyelash.

This approach has been implemented by the follow algorithm.

1. Gabor filter is applied after strong reflection detection (cf. section 2) and iris localization (cf. section 3).
2. Compute the gradient direction and quantify the result in four angles 0, 45, 90, 135 degrees.
3. Set the frequency of the sinusoidal u = ¼π. ½π, ¾π.
4. Calculate component x.
5. Obtain Gabor filter coefficients using equation (1).
6. Spatially convolve image with the filter to get the enhanced image.
7. Eliminate the pixels which belong to an eyelash.

This algorithm, use Gabor filter to enhance the pixels and eliminates the separable eyelash. The filter is divided in different frequency scales, to ensure that the same proportion of the spectrum is covered in both dimensions. In this experiment three different frequencies are used. The first frequency used was ¼π, then ½π and ¾π. The result is a set of filters [15] that covers one half of the frequency plane, the other half of the plane is not needed because the extra filters would have the same response as the existing ones.

4.2 Multiple Eyelashes

Many eyelashes overlap in a small area. Such that, the change of intensity variation in this area is almost zero. Thus, is necessary to obtain the variance of the intensity in this area and verify if is smaller than a threshold. It can be described [6] as:

$$\sum_{i=-N}^{N}\sum_{j=-N}^{N}\frac{(f(x+i,y+j)-M)^2}{(2N+1)^2} < K_2 \tag{2}$$

Where M is the mean of intensity in the small window; $(2N+1)^2$ is the window size and K_2 is a threshold.

5 Implementation and Results

In the experiments we test the segmentation system with 150 images from the MBGC NIR eyes still data base [16]. This contains 8590 eyes images. In this data set were acquired using an Iridian LG EOU 2200 camera.

The figure 1 shows a diagram with all the steps to obtain accurate iris segmentation. Once the eye image is obtained, the first step consist in verify if it has a strong reflection (cf. section 2). Then an edge map is calculated using canny filter to improve circle parameter finding. We use CHT to calculate iris/pupil circle parameter.

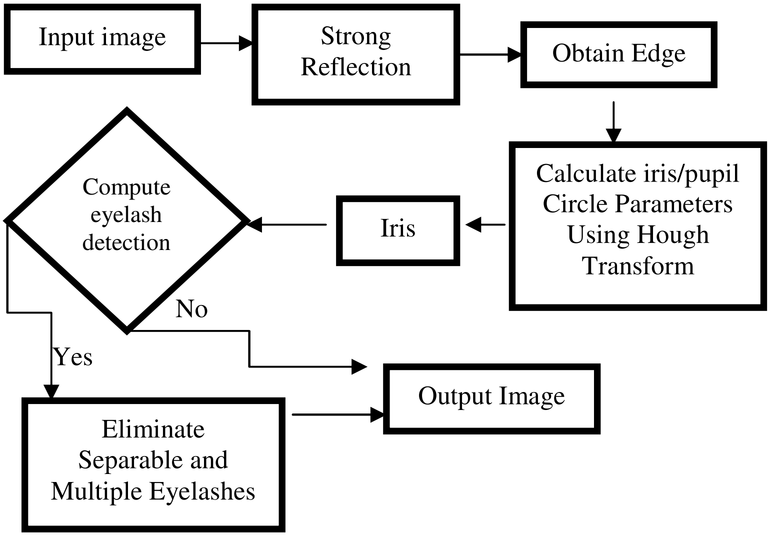

Input image
Strong Reflection
Obtain Edge
Calculate iris/pupil Circle Parameters Using Hough Transform
Iris
Compute eyelash detection
No
Yes
Output Image
Eliminate Separable and Multiple Eyelashes

Fig. 1. All steps to obtain accurate iris segmentation

The algorithm uses these parameters to extract the iris region (cf. section 3). However, after eye data base analysis, it shows that it is more probable to find eyelash just in the region near at the upper eyelid or lower eyelid than the region where the pupil exist. Thus it is necessary to decide when the Gabor filter should be used. To resolve this issue, this algorithm divides the extracted iris region in three blocks (upper, middle and lower).

After extracted iris region is divided, the algorithm calculates the mean μ_r and the standard deviation σ_r of the whole region. Then the algorithm calculates the mean μl_j and the standard deviation σl_j for each j block. To determine which block has eyelash occlusion is important to analyze the intensity of the pixels. Due to changes of intensities, we conclude that: if one block has eyelashes the difference between the pixels is significant. Such that, its mean and standard deviation are bigger than the statistical values of the whole region (condition 3) then the eyelash segmentation algorithm is computed.

$$\mu_r < \mu l_j \text{ and } \sigma_r < \sigma l_j \tag{3}$$

As a result we eliminated the eyelashes from the iris region without changing the original iris pattern. In other words, after removing the eyelashes region we obtain an accurate iris pattern free of any noise either from reflection or eyelash occlusion which will improve iris recognition performance. On the other hand, we compare our segmentation method versus Libor's Masek segmentation and eyelash detection method [17]. This algorithm eliminated the eyelashes by occluding them through a rectangle. The results show that our segmentation is accurate, it performed eyelash detection and it is 90% faster than Libor Masek. Some examples are shown in figure 2 which is distributed at the left row, is the original image, in the center row is the iris extracted from our algorithm and the right row is the iris extracted using Libor's method. Where images a) and b) shows that our algorithm finds the region of the eyelashes. The eyelashes issue increases when they are coated; due to the similitude of intensities between the pupil and eyelash pixels. From the image c) the algorithm could extract the iris region. The image d) shows the accuracy of our iris segmentation. However, Libor's method does not have that accuracy for this image. The Circular Hough Transform is the most intensive search as remarked in [8]. Even though, this algorithm is our first approach to iris recognition. This segmentation was

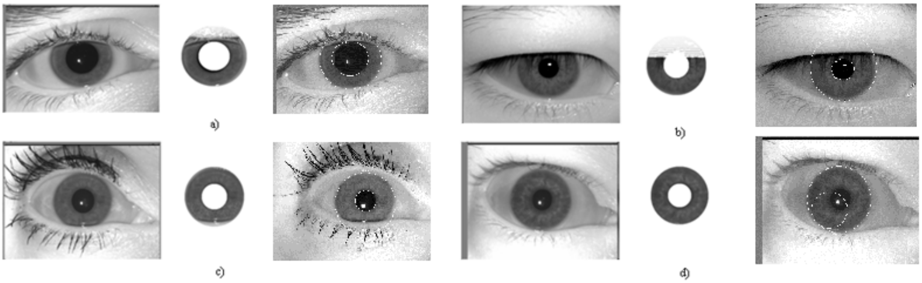


Fig. 2. Results of Iris image segmentation

accurate and fast enough to keep working on it. However, computing time of the CHT is one limitation that will be improved in further work.

Our eyelash detection is base on [4]. On the other hand, we introduce the three iris region blocks to statistical study of intensity variation of the pixels which improve the eyelash detection.

6 Discussions and Further Work

We have shown how to extract the iris region, eliminate the reflections and remove the eyelash occlusion using statistical values. Our method is fast, straightforward to implement, and accurately for a wide variety of images from MBGC NIR eye still data base. This method explains how a Circular Hough Transform should be implemented. On one hand, after analyze the data base; this article introduces a new way to detect where the eyelashes are by dividing the iris region in three iris blocks (upper, middle and lower). Then each block is statistically evaluated and classified if that block has or not eyelashes to detect eyelash. For each detected block, this method analyzes the Gabor filter that is used to eliminate the separable eyelashes and study how much is the chance of intensity variation to eliminate multiple eyelashes. Nevertheless, our method does have some limitations, and there are several avenues for future work.

The primary limitation of our method is that computed time to find the pupil's search region be improved. The first further work will perform a histogram analysis combined with morphologic models as thinning to enhance pupil and iris segmentation. When the pupil's radius is too small close to ten pixels the segmentation method could not be accurate because the Circular Hough Transform is a voting method to find circles boundary. Where in an edge map each pixel votes. If some edge is bigger than a small pupil edge then other circle boundary will be founded instead of the pupil boundary. Other further work is that we will improve our method to analyze iris segmentation from video using the MBGC NIR eye video data base instead of the MBGC NIR eye still data base.

Acknowledgment. This work was supported by IPN-SIP20100030.

References

1. Daugman, J.: Results from 200 Billion iris Cross-Comparisons. Technical Report. UCAM-CL-TR-635, ISSN 1476-2986, Number 635 (June 2005)
2. Zamudio, L.M., García, M.S., Colores, J.M.: Revisión de las Etapas de Adquisición y Pre-Procesamiento de Imagen en un Sistema de Reconocimiento Basado en Iris. VI Taller-Escuela de Procesamiento de Imágenes PI09 CIMAT. Agosto, Gto. México (2009)
3. Xu, G.Z., Zhang, Z.F., Ma, Y.D.: Automatic Iris Segmentation Base on Local Areas. In: The 18th International Conference on Pattern Recognition (ICPR'06). IEEE, Los Alamitos (2006)
4. Kong, Z.: Detecting Eyelash and Reflection For Accurate Iris Segmentation. International Journal of Pattern Recognition and Artificial Intelligence 17(6), 1025–1034 (2003)

5. Kang, P.: A Robust Eyelash Detection Based on Iris Focus Assessment. Pattern Recognition Letters 28, 1630–1639 (2007)
6. Yuan, W.H.: A Novel Eyelash Detection Method for Iris Recognition. In: Proceeding on the 2005 IEEE. Engineering in Medicine and Biology 27th Annual Conference Shanghai, China, pp. 6536–6539 (2005)
7. Lim, S., Lee, K., Kim, J.: Efficient iris recognition by characterizing key local variations. IEEE Transactions on Image Processing 13(6) (June 2004)
8. Otero-Mateo, N., Vega-Rodríguez, M.Á., Gómez-Pulido, J.A., Sánchez-Pérez, J.M.: A Fast and Robust Iris Segmentation Method. In: Martí, J., Benedí, J.M., Mendonça, A.M., Serrat, J. (eds.) IbPRIA 2007. LNCS, vol. 4478, pp. 162–169. Springer, Heidelberg (2007)
9. Chen, Y., Wang, J., Han, C., Wang, L., Adjouadi, M.: A robust segmentation approach to iris recognition based on video. In: 37th IEEE Applied Imagery Pattern Recognition Workshop, pp. 1–8 (2008)
10. Rizon, M., Yazid, H., Saad, P., Shakaff, A.Y.M., Sugisaka, A.R.S.M., Yaacob, S., Rozailan Mamat, M., Karthigayan, M.: Object Detection using Circular Hough Transform. American Journal of Applied Sciences 2(12), 1606–1609 (2005) ISSN 1546-9239
11. Hough Transform http://www.vislab.uq.edu.au/education/sc3/2000/hough/report/node3.html
12. Hough Tranform http://www.cis.rit.edu/class/simg782/lectures/lecture_10/lec782_05_10.pdf
13. Gabor filter. Multiresolution Design of Multiple Gabor Filters for Texture Segmentation, http://wws2.uncc.edu/tpw/diss/diss.html
14. Kovesi, P.: Gabor filter and Log-Gabor filter http://www.csse.uwa.edu.au/~pk/research/matlabfns/PhaseCongruency/Docs/convexpl.html
15. Christmas, B.: Designing complex Gabor Filters (November 16, 2007), http://www.ee.surrey.ac.uk/CVSSP/Ravl/RavlDoc/share/doc/RAVL/html/Gabor.pdf
16. Multi Biometric Grand Challenge MBGC, http://face.nist.gov/mbgc/
17. Masek, L., Kovesi, P.: MATLAB Source Code for a Biometric Identification System Based on Iris Patterns. The School of Computer Science and Software Engineering. The University of Western Australia (2003)

Adaboost Classifier by Artificial Immune System Model

Hind Taud[1], Juan Carlos Herrera-Lozada[2], and Jesús Álvarez-Cedillo[1]

[1] Centro de Innovación y Desarrollo Tecnológico en Cómputo
[2] Centro de Investigación en Computación
Instituto Politécnico Nacional
Av. Juan de dios Bátiz, Gustavo A. Madero, C.P. 07700, México, D.F.
{htaud,jlozada,jaalvarez}@ipn.mx

Abstract. An algorithm combining Artificial Immune System and AdaBoost called Imaboost is proposed to improve the feature selection and classification performance. Adaboost is a machine learning technique, which generates a strong classifier as a combination of simple classifiers. In Adaboost, through learning, the search for the best simple classifiers is replaced by the clonal selection algorithm. Haar features extracted from face database are chosen as a case study. A comparison between Adaboost and Imaboost is provided.

Keywords: Artificial immune system; Feature selection; Adaboost; Clonal selection algorithm; Haar Features.

1 Introduction

Machine learning is an active research topic in computer vision and pattern recognition research, which is applied in various fields such as the identity authentication; man-machine interface; virtual reality; content-based retrieval and many other aspects. Recently, Viola and Jones [1, 2] developed a method based on Adaboost classifier that performs a high detection rate. On the one hand, this method is considered to be one of the fastest systems and can be used to detect any object. On the other, Adaboost is a machine-learning algorithm that performs two tasks simultaneously: feature selection and forming a classifier using combination of these features. In Adaboost, to select the best features, a search is made of all the features. In the case of a large feature space such as Haar features, a reduction in this search step allows an increase in the performance of the classification.

In this article, we propose a new algorithm called Imaboost which is based on combining the Artificial immune system AIS model with Adaboost . The search step is replaced by the clonal selection algorithm. CLONALG is integrated with Adaboost to improve the feature selection for image classification. The objective is to present the Imaboost system but not to compare it in detail with the many other algorithms. Nonetheless, the results are compared with the original Adaboost algorithm. Haar features extracted from face database are chosen as a case study. Related works, AIS, Adaboost and Haar features are briefly described before the main lines of the algorithm are presented. Experiments and results are provided.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 171–179, 2010

2 Related Works

Viola and Jones Method contains three essential points: Haar features from the integral image, the Adaboost classifier and the cascade structure. The cascade allows non-object to be rejected quickly and subsequently quickens the detector. Features can be calculated extremely fast from the integral image. Adaboost is a machine-learning algorithm that performs two tasks simultaneously: feature selection and forming a classifier using a combination of these features. It is considered to be the key to a high detection rate. Feature selection is a step that follows feature extraction in a pattern recognition process. The aim of this step is to obtain the optimal feature subset from the input space that can achieve the highest accuracy results. Most of feature selection algorithms involve a combinatorial search through the whole space. Usually, heuristic methods, such as hill climbing, have to be adopted, because of the large number of features of input space [3].

Due to the high detection rate and real-time execution of the Viola and Jones approach, different investigations try to enhance the idea of boosting simple weak classifiers or to improve the response of the Adaboost classifier. Lienhart and Maydt [4] showed that extending the basic feature set yields detectors with lower error rates. Li and Zhang [5] described a variant of Adaboost called Floatboost for learning better classifiers. Zhang y al. [6] presented Z-Adaboost and Chang and Lee [7] proposed the Segment-Boost.

The use of Evolutionary Algorithms has received growing interest in the field of automatic learning. Genetic Algorithms are used as optimization procedures inspired by the mechanisms of natural selection. Within Adaboost, which is considered to be an optimization problem, genetic algorithms are used to find better classifiers as proposed by Treptow and Zell [8]. Zin et al. [9] extended the work of these authors by implementing GA inside the Adaboost to select features. Jang and Kim [10] introduced the employment of Evolutionary Pruning that reduces the number of weak classifiers. Chouaib et al. [11] presented a fast method combining genetic algorithm and Adaboost classifiers for feature selection. Li et al. [12] proposed dynamic Adaboost learning with feature selection based on a parallel genetic algorithm.

Just as the GA, the Artificial immune system (AIS) has been also applied successfully to a variety of optimization problems [13]. AIS is a computational intelligence paradigm inspired by the biological immune system, which has found an application in pattern recognition [14] and machine-learning [15].

3 Artificial Immune System

Artificial immune systems (AIS) are computational systems inspired by the principles and processes of the immune system. Formal definition is given by De Castro and Timmis [16]. The algorithms typically exploit different theories and processes, which allow the acquired immunity system to solve a specific problem. Common techniques are the clonal selection algorithm, negative selection algorithm, immune network algorithms and dendritic cell algorithm. The first is chosen for this investigation.

3.1 Clonal Selection Algorithm

Proposed by Castro and Von Zuben [17], CLONALG (CLONal selection ALGorithm) is an algorithm inspired by the clonal selection theory of acquired immunity. As described by these authors, the algorithm starts with an initial set of random solutions called population and iterates over a number of rounds (G) or generations until a specific stopping condition is reached (Fig. 1).

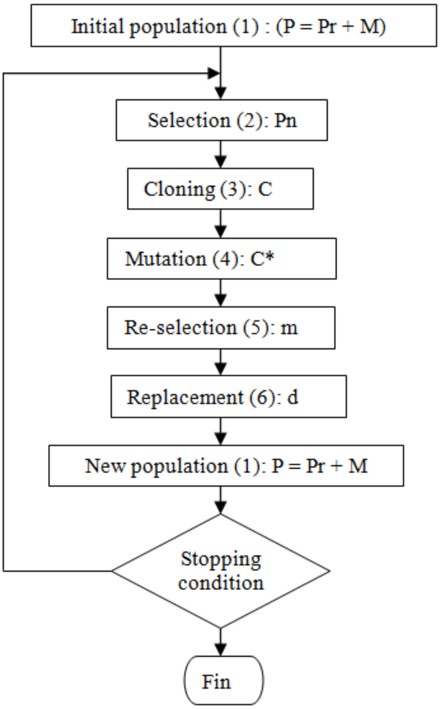


Fig. 1. One Clonal algorithm selection

The following provides the six steps composing CLONALG:

(1) Generate a set (P) of candidate solutions or antibodies, composed of the memory cells (M) and the remaining (Pr) population (P = Pr + M);

(2) Select the n best antibodies (Pn), based on an affinity measure;

(3) Clone these n best antibodies in proportion to their affinity; giving rise to a temporary set of clones (*C*);

(4) Apply a hypermutation to the temporary clones; the degree of mutation is inversely proportional to the affinity. A maturated antibodies is generated (*C**);

(5) Re-select the best elements from *C** to compose the memory set M. Some members of P can be replaced by other improved members of *C**;

(6) Replace d antibodies by novel ones to introduce the diversity concept. The probability to be replaced is inversely proportional to the affinity of the previous remaining (Pr) population.

4 Adaboost and Haar Features

4.1 Adaboost

The Adaboost was introduced by Freund and Schapire [18]. It is a machine learning technique, which combines weak classifiers in an iterative way to generate a final strong classifier through the learning process. A final classifier *H(x)* is a linear combination of the weak or simple classifiers $h_t\colon X \rightarrow \{0,1\}$

$$H(x) = \begin{cases} 1 & if\ g(x) \geq \text{threshold} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

$$\text{where } g(x) = \sum_{t=1}^{T} \alpha_t h_t\,(x) \tag{2}$$

Each weak classifier h_t describes a single feature f_t:

$$h_t(x) = \begin{cases} 1 & if\ \ p_t f_t(x) < p_t \theta_t \\ 0 & \text{otherwise} \end{cases} \tag{3}$$

where θ_t is a threshold and p_t is a parity to indicate the direction of the inequality.

Input : N training set (x_i, y_j), i = 1,2, …,N
with negative $(y_j = 0)$ and positive $(y_j = 1)$ examples.

• Initialize weights $w_{1,i} = \frac{1}{2m}, \frac{1}{2n}$ where *m* and *n* are the number of negatives and positives examples respectively

• For *t* = 1, . . . , *T:*

1) Normalize the weights, $W_{t,i} \leftarrow \frac{wtj}{\sum_{j=1}^{n} wt,j}$

2) For each feature *j* train classifier $h_{j,\,t}$ with error $\epsilon_j = \sum_i w_i |h(x_i) - y_i|$

3) Choose the weak classifier h_t with the lowest error ϵ_t

4) Update the weights: $w_{t+1,i} = w_{t,i} \beta_t^{1-e_i}$ with

$$e_i = \begin{cases} 0 & x_i \ \text{ is classified correctly} \\ 1 & \text{otherwise} \end{cases}$$

and $\beta_t = \frac{\epsilon_t}{1-\epsilon_t}$

• The final strong classifier is :$C(x) = \begin{cases} 1 & \sum_{t=1}^{T} \alpha_t h_t\,(x) \ \geq \frac{1}{2} \sum_{t=1}^{T} \alpha_t \\ 0 & \text{otherwise} \end{cases}$

where $\alpha_t = log \frac{1}{\beta_t}$

Fig. 2. Adaboost algorithm

Adaboost (Fig.2) iterates over a number of T rounds. In each round, the features space is scanned in order to train the weak classifiers and to find the threshold θ_t, which discriminates between positive and negative examples. This threshold is calculated as the mean values of features that results on the positive and negative examples [8]. For each feature, the error value $\epsilon_t$is estimated. The best feature with the lowest error is selected as the weak classifier for this iteration. All training examples are reweighted and normalized. The following iteration is executed and another weak classifier is selected. After T iterations, the resulting strong classifier is formed as a combination of all T weak classifiers.

4.2 Haar Features Extraction

Three kinds of feature are considered: two-rectangle, three-rectangle, and four-rectangle feature. The rectangular regions have the same size and shape and are horizontally or vertically adjacent. The value of each feature (Fig. 3) is the difference between the sum of the pixels within the white and black rectangular regions.

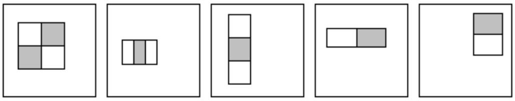


Fig. 3. Haar features

The possible positions and scales of the three basic feature types within a sub window size, for example of 24x24 pixels produce about 160,000 possible alternative features. In order to compute these features very quickly, the integral image representation or Summed-area table [19], is introduced. At the location (x, y) (Fig. 4(a)) the integral image $II(x,y)$ contains the sum of the pixels above and to the left of x, y:

$$II(x,y) = \sum_{x' \leq x, y' \leq y} I(x',y') \tag{4}$$

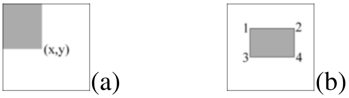


Fig. 4. Feature Estimation: (a) Integral image at point (x, y); (b) Four references to obtain the gray rectangular sum

The integral image can be computed from an image using a few operations per pixel. Therefore, Harr-like features can be calculated at any scale or location in constant time. Any rectangular sum can be estimated by four references (Fig. 4(b)). Two adjacent rectangular sums can be obtained from six array references, eight or nine references in the case of the three-rectangle, and four-rectangle features respectively.

5 Artificial Immune Adaboost: Imaboost

The aim in this article is to apply the artificial immune system in Adaboost to improve the feature selection step. The search over all features in step 2 (Fig. 2) is replaced by

CLONALG (Fig. 1). This algorithm is relatively low in complexity and requires a small number of user parameters such as the number of generations (G), population size (P), memory antibody size (M), selection antibody size (N), remainder replacement size (d), clonal factor (β), Re-select size (m) from the maturate antibodies.

The antibody representation has to be chosen to represent the features and the affinity to resolve the minimization problem. The antibody representation and the affinity are chosen as those given by Treptow and zell [8] and Zin et al. [9]. Every feature is encoded by a string of 5 integer variables (t, x, y, x´, y´) where t represents a type of feature (Fig. 3), (x,y) and (x´,y´) are the coordinate of the upper left and the lower right corner of the feature in the sub-window. The affinity is described as follows:

$$Aff = 1 - \epsilon \tag{5}$$

ϵ is the error function as estimated in Adaboost.

The different operations of the algorithm are directly or inversely proportional to the affinity. The rank based measure is achieved by sorting the set of selected antibodies in ascending or descending order by their affinity. The number of clones created for each antibody is calculated as follows:

$$c_i = \left[\frac{\beta . P}{i} + 0.5\right] \tag{6}$$

where β is a clonal scaling factor, P is the population size, and i is the antibody current rank where $i \in [0, N]$ and N is the number of selected antibodies. The total number of clones is then calculated as a summation of all c_i. The mutation is performed by creating a new type t and changing the corner positions of the feature by adding a random constant in the integer set {-3,…,3}. A probability rate $Pm \in [0,1]$ is used to defined the m and d size.

6 Experiments and Result

The Imaboost is compared with the standard Adaboost to test the response and performance of Imaboost in relation to features selection and classification. The face is chosen as a case study. The training and testing set are obtained from various sources. They consist of 3000 positive and 5000 negative images for the training set (Fig. 5). The testing set, which is different to the training set, consists of 2000 positive and 3000 negatives images. Gray images of size 24×24 are employed.

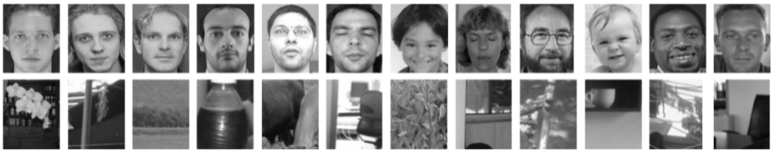


Fig. 5. Images face / non face set

The clonal algorithm parameters are chosen by testing different values: 50 for population size, 35 for memory antibody size, β=1 for the clonal coefficient. The mutation process decreases by 0.2% with each generation .The algorithm converges and stops when no better solution is found within the next 50 generations. If there is no convergence within the maximum number of generations, set at 300 generations, the algorithm is stopped as well.

With both algorithms, the training step is stopped when all the examples are labeled correctly. The experiments are carried out on an Intel Core 2 Quad Q9550 processor. Imaboost is run 27 times and the average results are taken. The result of applying Adaboost and Imaboost is summarized in Table 1, 2 and 3. Imaboost is able to find classifiers with a lower number of features with less training time compared to Adaboost. The average number of features is 160 for Imaboost compared to 209 for Adaboost. It represents 77% of the number of features selected by Adaboost. The mean time for the search for a weak classifier on the face set is 14.7 seconds for Imaboost compared to 44.8 seconds for Adaboost. The selection of a single feature is then 3 times faster in Imaboost than Adaboost. Reducing the features number and time per iteration implies a reduction of total training time for Imaboost.

Table 1. Features selection

	worst	Best	Average feature
Adaboost	209	209	209
Imaboost	172	148	160

Table 2. Training time

	Average time per iteration(s)	Average Total time (s)
Adaboost	44.8s	9363s
Imaboost	14.7s	2352s

Table 3. Rates on test set

	Classification rate %	False positive rates%
Adaboost	95.9	0.040
Imaboost	96.6	0.031

The learned classifiers are evaluated on the test set to compare classification and false positive rates as shown in Table 3. The learned classifiers use 150 features with Imaboost and 209 with Adaboost. Although a lower features number is used, Imaboost provides similar detection and false positive rates. Imaboost classifies 96.6% of the set correctly whereas Adaboost gives a classification rate of 95.9%.

7 Conclusion and Future Works

In this paper, an approach for feature selection and classification is presented, using a model of artificial immune system. A new combination of Adaboost and clonal algorithm is investigated in order to overcome the problem of feature selection in the huge search space. A comparison between Adaboost and Imaboost applied to a set of face is presented. Preliminary results show that Imaboost improves the performance of the classifier. The number of features and training time is reduced preserving the same classification rate. Moreover, a more thorough study of the performance of Imaboost, requires the performance of more experiments in different image sets.

In order to increase the detection speed, Viola and Jones use a cascade of various Adaboost. Improving each Adaboost implies improving the entire cascade. On the one hand future work can be directed to test the cascade of classifiers produced by Imaboost. On the other hand, in order to enhance the performance of Imaboost, a micro immune system with a reduced population size should be studied. A comparison between different evolutionary algorithms used with Adaboost should also be made.

Acknowledgments

We thank Anna Reid for improving the English grammar and style of our manuscript.

References

1. Viola, P., Jones, M.: Rapid object detection using boosted cascade of simple features. In: Proceedings IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Hawaii, vol. 1, pp. 511–518 (2001)
2. Viola, P., Jones, M.: Robust Real-Time Face Detection. International Journal of Computer Vision 57(2), 137–154 (2004)
3. Zheng, L., He, X.: Classification Techniques in Pattern Recognition. In: Proceedings of the 13th International Conference in Central Europe on Computer Graphics, Visualization and Computer Vision, Bory, pp. 77–79 (2005)
4. Lienhart, R., Maydt, J.: An extended set of haar-like features for rapid object detection. In: IEEE ICIP2, New York, vol. 1, pp. 900–903 (2002)
5. Li, S.Z., Zhang, Z.: FloatBoost Learning and Statistical Face Detection. IEEE Transactions on Pattern Analysis and Machine Intelligence 26(9), 1112–1123 (2004)
6. Zhang, W., Tong, R., Dong, J.: Z-AdaBoost: Boosting 2-Thresholded Weak Classifiers for Object Detection. In: IITA08 Second International Symposium on Intelligent Information Technology Application, Shanghai, vol. 2, pp. 839–844 (2008)
7. Chang, W.S., Lee, J.S.: Segment-Boost Learning for Facial Feature Selection. In: Proceedings of the Third International Conference on Convergence and Hybrid Information Technology, vol. 1, pp. 358–363 (2008)
8. Treptow, A., Zell, A.: Combining Adaboost Learning and Evolutionary Search to select Features for Real-Time Object Detection. In: CEC 2004 Congress on Evolutionary Computation, vol. 2, pp. 2107–2113 (2004)
9. Zin, Z.M., Khalid, M., Yusof, R.: Enhanced Feature Selections OF Adaboost training for face detection using genetic algorithm (gaboost). In: Proceedings of the Third IASTED International Association of Science and Technology For Development, Alberta, pp. 34–39 (2007)

10. Jang, J.S., Kim, J.H.: Evolutionary Prunning for Fast and Robust Face Detection. In: CEC 2006 IEEE Congress on Evolutionary Computation, Vancouver, pp. 1293–1299 (2006)
11. Chouaib, H., Ramos Terrades, O., Tabbone, S., Cloppet, F., Vincent, N.: Feature selection combining genetic algorithm and Adaboost classifiers. In: 19th International Conference on Pattern Recognition (ICPR), Tampa, pp. 1–4 (2008)
12. Li, R., Lu, J., Zhang, Y., Zhao, T.: Dynamic Adaboost learning with feature selection based on parallel genetic algorithm for image annotation. Knowledge-Based Systems 23(3), 195–201 (2010)
13. Tan, K.C., Goh, C.K., Mamun, A.A., Ei, E.Z.: An evolutionary artificial immune system for multi-objective optimization. European Journal of Operational Research 187(2), 371–392 (2008)
14. Carter, J.H.: The immune system as a model for pattern recognition and classification. Journal of the American Medical Informatics Association 7(1), 28–41 (2000)
15. Hunt, J.E., Cook, D.E.: Learning using an artificial immune system. Journal of Network and Computer Applications 19, 189–212 (1996)
16. De Castro, L.N., Leandro, N.: Timmis, Jonathan Artificial Immune Systems: A New Computational Intelligence Approach. Springer, Heidelberg (2002)
17. De Castro, L.N., Von Zuben, F.J.: The clonal selection algorithm with engineering applications. In: Workshop Proceedings of GECCO'00, Workshop on Artificial Immune Systems and their Applications, Las Vegas, pp. 36–37 (2000)
18. Freund, Y., Schapire, R.E.: A Short Introduction to Boosting. Journal of Japanese Society for Artificial Intelligence 14(5), 771–780 (1999)
19. Crow, F.C.: Summed-area tables for texture mapping. In: SIGGRAPH '84 Proceedings of the 11th annual conference on Computer graphics and interactive techniques, pp. 207–212. ACM Press, New York (1984)

Cost-Sensitive Neural Networks and Editing Techniques for Imbalance Problems

R. Alejo[1], J.M. Sotoca[1], V. García[1], and R.M. Valdovinos[2]

[1] Institute of New Imaging Technologies
Dept. Llenguatges i Sistemes Informátics, Universitat Jaume I
Av. Sos Baynat s/n, 12071 Castelló de la Plana (Spain)
[2] Centro Universitario UAEM Valle de Chalco, Universidad Autónoma del Estado de México
Hermenegildo Galena No.3, Col. Ma. Isabel, 56615 Valle de Chalco (Mexico)

Abstract. The multi-class imbalance problem in supervised pattern recognition methods is receiving growing attention. Imbalanced datasets means that some classes are represented by a large number of samples while the others classes only contain a few. In real-world applications, imbalanced training sets may produce an important deterioration of the classifier performance when neural networks are applied in the classes less represented. In this paper we propose training cost-sentitive neural networks with editing techniques for handling the class imbalance problem on multi-class datasets. The aim is to remove majority samples while compensating the class imbalance during the training process. Experiments with real data sets demonstrate the effectiveness of the strategy here proposed.

Keywords: Multi-class imbalance; backpropagation; cost function; editing.

1 Introduction

Neural networks have become a popular tool in Pattern Recognition, Machine Learning and Data Mining [1]. Although there are several kinds of neural networks, most attention has been focused on the use of Multilayer Perceptron (MLP) [2], or feed-forward networks trained with a backpropagation learning algorithm for supervised classification.

However, it is well known that in MLP, the nature of the Training Data Sets (TDS) has a major impact on the ability of the network to generalize[2]. One of the problems in the complexity of the TDS that most affects the neural networks is the class imbalance [3].

A two-class data set is said to be imbalanced when one of the classes (the minority one) is heavily under-represented with regard to the other class (the majority one) [4]. This problem is encountered in a large number of domains, and in certain cases, it has been observed that class imbalance may cause a significant deterioration in the performance attainable by standard learners because these are often biased towards the majority class [5].

Many works have addressed the class imbalance problem [6,5]. The most popular strategies for dealing with this problem can be grouped in three categories. One is to assign different costs to the classification errors [3]. The second technique is to

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 180–188, 2010.

make a resampling of the original TDS, over-sampling the minority class and/or under-sampling the majority class until the classes are approximately equally represented [4]. The third technique consists in internally biasing the discrimination-based process and compensate the class imbalance [7,8]. However, these techniques do not consider other complexities that might result from TDS. In this regard, several studies suggest that other problems such as the overlap between classes should be taken into account in classification tasks [9,10].

In this work, we present some preliminary results to explore two issues related with the Multi-Class Imbalance Problem. Initially, we remove majority samples from the overlap region, producing a local balance of the classes. For this, the only requirement is that all samples of the minority classes must be saved in the TDS. As downsizing of the majority classes can throw away significant information, an editing scheme is applied. Note that a global balance in the class sizes is not achieved. Subsequently, the backpropagation algorithm is modified to avoid that the minority classes be ignored in the learning process, and to accelerate the convergence of the neural network.

2 Multilayer Perceptron

The multilayer perceptron (MLP) neural network [11] usually comprises one input layer, one or more hidden layers, and one output layer. Input nodes correspond to features, hidden layers are used for computations, and output layers are related with the number of classes. A neuron is the elemental unit of each layer. It computes the weighted sum of its inputs, adds a bias term and drives the result thought a generally nonlinear (commonly a sigmoid) activation function to produce a single output.

The most popular training algorithm for MLP is the backpropagation strategy, which uses a set of training instances for the learning process. Given a feedforward network, the weights are initialized to small random numbers. Each training instance is sent through the network and the output from each unit is computed. The target output is compared with the output estimated by the network calculating the error, which is fed-back through the network.

To adjust the weights, the backpropagation algorithm uses a gradient descent to minimize the squared error. At each unit in the network starting from the output unit and moving to the hidden units, its error value is used to adjust the weights of its connections as well as to reduce the error. This process is repeated for a fixed number of times, or until the error is small.

2.1 The Backpropagation Algorithm and the Class Imbalance Problem

Empirical studies of the backpropagation algorithm [12] show that class imbalance problem generates unequal contributions to the mean square error (MSE) in the training phase. Clearly the major contribution to the MSE is produced by the majority class.

Let us consider a TDS with two classes ($m = 2$) such that $N = \sum_i^m n_i$ and n_i is the number of samples from class i. Suppose that the MSE by class can be expressed as

$$E_i(U) = \frac{1}{N} \sum_{n=1}^{n_i} \sum_{p=1}^{L} (d_p^n - y_p^n)^2 \,, \tag{1}$$

where d_p^n is the desired output and y_p^n is the actual output of the network for sample n. Then the overall MSE can be expressed as

$$E(U) = \sum_{i=1}^{m} E_i = E_1(U) + E_2(U) \, . \tag{2}$$

If $n_1 << n_2$ then $E_1(U) << E_2(U)$ and $\|\nabla E_1(U)\| << \|\nabla E_2(U)\|$, consequently $\nabla E(U) \approx \nabla E_2(U)$. So, $-\nabla E(U)$ it is not always the best direction to minimize the MSE in both classes.

Considering that the imbalance problem affects negatively in the backpropagation algorithm due to the disproportionate contributions in the MSE, it is possible to consider a cost function (γ) that balance the TDS class imbalance as follows:

$$\begin{aligned} E(U) &= \textstyle\sum_{i=1}^{m} \gamma(i) E_i = \gamma(1)E_1(U) + \gamma(2)E_2(U) \\ &= \tfrac{1}{N} \textstyle\sum_{i=1}^{m} \gamma(i) \sum_{n=1}^{n_i} \sum_{p=1}^{L} (y_p^n - F_p^n)^2 \, , \end{aligned} \tag{3}$$

where $\gamma(1)\|\nabla E_1(U)\| \approx \gamma(2)\|\nabla E_2(U)\|$ avoiding that the minority class be ignored in the learning process. In this work, the cost function is defined as

$$\gamma(i) = \|\nabla E_{max}(U)\| / \|\nabla E_i(U)\|, \tag{4}$$

where $\|\nabla E_{max}(U)\|$ corresponds to the largest majority class.

When a cost function is included in the training process, the data probability distribution is altered [13]. However, this cost function (Eq. 4) reduces its impact in the data distribution probability because the cost function value is diminished gradually. In this way, the class imbalance problem is reduced in early iterations, and later $\gamma(m)$ reduces its effect on the data distribution probability.

3 Edited Nearest Neighbor Rule

Wilson [14] developed the Edited Nearest Neighbor (ENN) algorithm in which the set of samples **S** starts out the same as TDS, and then each instance of the set **S** is removed if it does not agree with the majority of its k nearest neighbors (with k=3, typically). This method removes noisy instances as well as samples at the borderline, leaving smoother decision boundaries. Algorithmically, the ENN scheme can be expressed as follows:

1. Let $\mathbf{S} = \mathbf{X}$.
2. For each $\mathbf{x}_i$ in $\mathbf{X}$ do:
   - Discard $\mathbf{x}_i$ from $\mathbf{S}$ if it is misclassified using the k-NN rule with prototypes in $\mathbf{X} - \{\mathbf{x}_i\}$.

In this work, the ENN is applied only in the majority classes. The aim is to reduce the complexity in the overlap region maintaining all the minority samples. This technique can be seen as focused under-sampling.

4 Methodology

The experiments were carried out on three images real data sets (Cayo, Feltwell and Satimage). A brief summary is given in the Table 1. For each database, a 10–fold cross–validation was applied. The datasets were divided into ten equal parts, using nine folds as training set and the remaining block as test set.

Table 1. A brief summary of some basic characteristics of the databases

Dataset	Size	Attr.	Class	Class distribution
Cayo	6019	4	11	838/293/624/322/133/369/324/722/789/833/772
Feltwell	10944	15	5	3531/2441/896/2295/1781
Satimage	6430	36	6	1508/1531/703/1356/625/707

The *Accuracy* and *g-mean* are used as performance measure to evaluate the classifier. It is common to obtain measure criteria from the confusion matrix where real classes are in columns, whereas predicted ones appear in rows (Table 2). The table built in this way is a general vision assignment, where diagonal elements count the correctly assigned samples and elements out of the diagonal count the wrongly classified ones.

From the confusion matrix, we can define

$$Accuracy = \sum_{i=1}^{m} n_{ii}/N \, , \tag{5}$$

where N is the total number of samples.

$$Accuracy\ by\ class = n_{ii}/n_{i+} . \tag{6}$$

Other measure used is the geometric mean (*g-mean*) defined as

$$g\text{-}mean = (\prod_{i=1}^{m} n_{ii}/n_{i+})^{\frac{1}{m}} . \tag{7}$$

All the MLP were trained with the backpropagation algorithm in batch mode. This process has been repeated ten times and the results correspond to the average. The learning rate (η) was set to 0.1 and only one hidden layer was used. The number of neurons for the hidden layer was established to 7, 6 and 12 for Cayo, Feltwell and Satimage datasets respectively.

Table 2. Confusion matrix for a multi-class problem

	Real Classes				
Predicted Classes	1	2	$\cdots$	m	total (n_{i+})
1	n_{11}	n_{12}	$\cdots$	n_{1m}	n_{1+}
2	n_{21}	n_{22}	$\cdots$	n_{2m}	n_{2+}
$\vdots$	$\vdots$	$\vdots$		$\vdots$	$\vdots$
m	n_{m1}	n_{m2}	$\cdots$	n_{mm}	n_{m+}
total (n_{+j})	n_{+1}	n_{+2}	$\cdots$	n_{+m}	N

Summarizing the strategy proposed in this work consists of the following:

1. To edit the TDS with the ENN technique, removing only majority samples in the overlap region and producing a local balance of the classes (sec. 3).
2. To modify the backpropagation algorithm applying a cost-function (Eq. 4) to avoid that the minority classes would be ignored in the learning process, and accelerating the convergence of the neural network.
3. To train the MLP with the modified algorithm over the TDS edited.

5 Results and Discussion

In order to evaluate the possibilities of the proposed approaches here exposed, several experiments with imbalance data sets were developed. In Tables 3, 4, 5 and 6 the main results are detailed. In these experiments, we denote "Cost-MLP" the cost function with MLP and "TDS edited" the imbalanced training set edited.

Table 3 shows the percentage of samples eliminated after applying the edition algorithm in the majority classes. In the case of Cayo database, the classes 1, 3, 8, 9, 10 and 11 were considered as majority classes. On the other hand, in Feltwell database, only the class 3 was identified like minority class. For Satimage database, the classes 1, 2 and 4 were considered as majority classes. The experiments used different values of k in the edition process choosing the most suitable for each database: Cayo $k = 15$, Feltwell $k = 9$ and Satimage $k = 5$.

In the case of majority classes, the number of samples eliminated were significant (see Table 3). This important reduction of the size tends to improve the classification accuracy in the minority classes. On the other hand, it is possible observe that in some majority classes, the number of samples eliminated was minimum: classes 1 and 8 for Cayo, classes 2, 4 and 5 for Feltwell and class 2 for Satimage.

The information presented in Tables 4, 5 and 6 was organized as follows. The first column of each table indicates the strategy applied, i.e., if the TDS were edited or not, or if we use the modified algorithm or the standard algorithm. The second column indicates the class to which the results correspond. In the third column (the ratio), we show the proportion of class elements in relation with the total samples ($ratio = n_i/N$, where n_i is the elements number of class i and N the total samples in the TDS). The fourth column is the classification accuracy and the last one shows the classes with the level of confusion is greater thant 10% (the percentage of confusion appears in brackets).

Table 3. Percentage of samples eliminated after editing the TDS

Class	1	2	3	4	5	6	7	8	9	10	11	Total reduction
Cayo	10.74	0.00	43.73	0.00	0.00	0.00	0.00	5.54	13.42	48.92	48.19	21.40%
Feltwell	15.26	10.93	0.00	11.2	13.14							11.89%
Satimage	47.5	12.69	0.00	60.56	0.00	0.00						27.31%

Table 4. Results of the classification phase with the MLP on the Cayo data base

	Class	Ratio	Accuracy	% confusion (> 10 %)
MLP + TDS	C-01	0.14	89.74	
	C-02	0.05	51.20	C-03 (48.63)
	C-03	0.10	95.69	
	C-04	0.05	70.99	C-03 (12.61) C-08 (11.43)
	C-05	0.02	19.92	C-01 (50.30) C-03 (19.39)
	C-06	0.06	56.44	C-07 (31.90)
	C-07	0.05	95.40	
	C-08	0.12	98.55	
	C-09	0.13	87.56	C-10 (12.44)
	C-10	0.14	77.03	C-11 (21.80)
	C-11	0.13	89.40	C-10 (10.14)
Cost-MLP + TDS	C-01	0.14	88.10	
	C-02	0.05	51.37	C-03 (48.63)
	C-03	0.10	93.42	
	C-04	0.05	93.54	
	C-05	0.02	73.79	C-01 (14.39) C-03 (11.52)
	C-06	0.06	60.43	C-07 (30.82)
	C-07	0.05	95.31	
	C-08	0.12	94.86	
	C-09	0.13	87.56	C-10 (12.44)
	C-10	0.14	76.36	C-11 (22.99)
	C-11	0.13	91.89	
MLP + TDS edited	C-01	0.14	88.15	
	C-02	0.05	51.99	C-03 (48.01)
	C-03	0.10	92.84	
	C-04	0.05	91.43	
	C-05	0.02	51.97	C-01 (25.23) C-03 (15.45)
	C-06	0.06	58.99	C-07 (31.90)
	C-07	0.05	96.29	
	C-08	0.12	97.60	
	C-09	0.13	87.56	C-10 (12.44)
	C-10	0.14	76.56	C-11 (15.06)
	C-11	0.13	75.41	C-09 (16.02)
Cost-MLP + TDS edited	C-01	0.14	86.87	
	C-02	0.05	70.72	C-03 (29.28)
	C-03	0.10	78.31	C-02 (14.79)
	C-04	0.05	94.22	
	C-05	0.02	86.67	
	C-06	0.06	60.95	C-07 (31.39)
	C-07	0.05	95.74	
	C-08	0.12	95.24	
	C-09	0.13	87.56	C-10 (12.44)
	C-10	0.14	73.94	C-11 (24.63)
	C-11	0.13	94.70	

In Table 4, we observe in Cayo dataset that the classes 2, 5 and 6 are affected seriously by the imbalance problem. We point out that when the imbalance is compensated with the cost function, the accuracy of the minority classes is increased (except for the class 2) especially in the case of class 5.

When the TDS is edited, the global accuracy and the performance of the minority classes are improved. Nevertheless, in the case of overlapped classes (see class 2 in Table 4) the results presented are practically the same.

When the classes imbalance is compensated and the network is trained with the TDS edited, the accuracy of the class 2 increases significantly. On the other hand, the combination of both strategies improves the rate of recognition on the minority classes. The classes 6 and 7 do not increase your performance due to these classes are overlapped each other.

Table 5. Feltwell: Classification with MLP

	Class	Ratio	Accuracy	% confusion ($>$ 10 %)
MLP + TDS	C-01	0.35	99.07	
	C-02	0.24	81.97	C-03 (11.72)
	C-03	0.10	78.86	C-01 (10.70)
	C-04	0.15	83.91	C-01 (11.48)
	C-05	0.17	90.43	
Cost-MLP + TDS	C-01	0.35	98.58	
	C-02	0.24	80.85	C-03 (14.85)
	C-03	0.10	83.08	
	C-04	0.15	83.35	C-01 (10.74)
	C-05	0.17	88.92	C-01 (10.55)
MLP + TDS edited	C-01	0.35	97.63	
	C-02	0.24	73.62	C-03 (13.99) C-05 (11.17)
	C-03	0.10	81.48	C-04 (10.09)
	C-04	0.15	83.19	
	C-05	0.17	96.12	
Cost-MLP + TDS edited	C-01	0.35	97.45	
	C-02	0.24	69.70	C-03 (23.85)
	C-03	0.10	84.70	
	C-04	0.15	81.80	
	C-05	0.17	95.76	

Table 6. Satimage: Classification results with the MLP

	Class	Ratio	Accuracy	% confusion ($>$ 10 %)
MLP + TDS	C-01	0.23	90.87	
	C-02	0.23	98.83	
	C-03	0.11	90.71	
	C-04	0.20	97.71	
	C-05	0.11	2.37	C-01 (61.04) C-04 (33.03)
	C-06	0.12	70.25	C-01 (15.74)
Cost-MLP + TDS	C-01	0.23	81.89	C-05 (13.83)
	C-02	0.23	97.51	
	C-03	0.11	90.54	
	C-04	0.20	91.61	
	C-05	0.11	65.73	C-01 (19.91) C-04 (13.22)
	C-06	0.12	76.71	C-01 (13.50)
MLP + TDS edited	C-01	0.23	75.21	C-05 (18.66)
	C-02	0.23	98.18	
	C-03	0.11	91.43	
	C-04	0.20	88.46	C-05 (10.18)
	C-05	0.11	60.95	C-01 (25.45) C-04 (11.04)
	C-06	0.12	77.05	
Cost-MLP + TDS edited	C-01	0.23	71.47	C-05 (23.09)
	C-02	0.23	96.83	
	C-03	0.11	93.39	
	C-04	0.20	83.27	C-05 (15.57)
	C-05	0.11	84.08	
	C-06	0.12	81.14	

The results of Feltwell database are included in Table 5. The use of the TDS edited improves the classifier effectiveness on the minority class 3. However, there is a tendency for reducing the network effectiveness on the majority classes, especially class 2.

Satimage database (see Table 6) shows a similar tendency to Cayo and Feltwell. When the TDS is edited, the classification on the minority classes is increased. The combination of both approaches increases the accuracy of minority classes and compensate the classes imbalance. This strategy qualitatively improves the accuracy of the minority classes. For example, in class 5 the accuracy is 65.73% with the original TDS,

Table 7. Global performance of the classifier

Cayo	MLP TDS	Cost-MLP TDS	MLP TDS edited	Cost-MLP TDS edited
Accuracy	83.58(0.77)	85.15(0.27)	82.75(1.90)	84.79(0.48)
g-mean	70.17(6.28)	80.96(0.44)	76.50(2.92)	83.30(0.78)

Feltwell	MLP TDS	Cost-MLP TDS	MLP TDS edited	Cost-MLP TDS edited
Accuracy	89.38(0.95)	89.01(0.51)	87.99(1.21)	87.04(0.71)
g-mean	86.60(1.64)	86.79(0.75)	85.97(1.47)	85.33(0.98)

Satimage	MLP TDS	Cost-MLP TDS	MLP TDS edited	Cost-MLP TDS edited
Accuracy	82.26(0.31)	86.07(0.34)	83.66(0.34)	84.59(0.38)
g-mean	47.31(5.72)	83.34(0.95)	80.90(1.17)	84.70(0.36)

whereas when the network is trained with the TDS edited and with the modified algorithm its value reaches 84.08%.

On the other hand, analyzing the global values of accuracy and geometric mean (see Table 7), we can see that these measures obtain better results when the TDS is edited, even in cases where the imbalance is not compensated.

In Feltwell database, it is possible that the proposed strategy does not represent a significant improvement. Only when we apply cost-functions in training neural network, the results are similar to original dataset. The editing technique proposed obtains clearly worse results and it is not adequate for this database.

Summarizing we can say that the editing of TDS and the application of cost functions in the neural network training reduces the confusion between classes. However, when we give priority to minority classes, the majority classes are affected in the training process with a loss of accuracy in these classes.

6 Conclusion

In this work we propose a strategy based on combination of training cost-functions with editing technique in neural networks to deal with the class imbalance problem on multi-class datasets. This generates two effects: a) to compensate the class imbalance during the training process and b) to reduce the confusion of the minority classes in the overlap region. With the edition of the majority classes it is possible to reduce the confusion between the minority and majority classes.

The modification of the training algorithm including a cost function increases the recognition rate of less represented classes, accelerating the convergence of the network.

However, we have seen in some situations that the proposed editing technique has not been adequate. Thus, it is interesting the use of new strategies to reduce the confusion region taking into account both the imbalance and the representativeness of the data.

Acknowledgment

This work has been partially supported by the Spanish Ministry of Science and Education under project CSD2007-00018, UAEMCA-114 and SBI112 from the Mexican SEP, the 2703/2008U from the UAEM project and P1.1B2009-45 (Fundació Caixa-Castelló).

References

1. Jain, A., Mao, J., Mohiuddin, K.: Artificial neural networks: A tutorial. Computer 29(3), 31–44 (1996)
2. Foody, G.: The significance of border training patterns in classification by a feedforward neural network using back propagation learning. International Journal of Remote Sensing 20(18), 3549–3562 (1999)
3. Zhou, Z.H., Liu, X.Y.: Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Transactions on Knowledge and Data Engineering 18, 63–77 (2006)
4. Japkowicz, N., Stephen, S.: The class imbalance problem: a systematic study. Intelligent Data Analysis 6, 429–449 (2002)
5. He, H., Garcia, E.: Learning from imbalanced data. IEEE Trans. on Knowl. and Data Eng. 21(9), 1263–1284 (2009)
6. Visa, S.: Issues in mining imbalanced data sets - a review paper. In: Artificial Intelligence and Cognitive Science Conference, pp. 67–73 (2005)
7. Anand, R., Mehrotra, K., Mohan, C., Ranka, S.: Efficient classification for multiclass problems using modular neural networks. IEEE Transactions on Neural Networks 6(1), 117–124 (1995)
8. Bruzzone, L., Serpico, S.: Classification of imbalanced remote-sensing data by neural networks. Pattern Recognition Letters 18, 1323–1328 (1997)
9. Visa, S., Ralescu, A.: Learning imbalanced and overlapping classes using fuzzy sets. In: Workshop on Learning from Imbalanced Datasets(ICML'03), pp. 91–104 (2003)
10. Prati, R., Batista, G., Monard, M.: Class imbalances versus class overlapping: An analysis of a learning system behavior. In: Monroy, R., Arroyo-Figueroa, G., Sucar, L.E., Sossa, H. (eds.) MICAI 2004. LNCS (LNAI), vol. 2972, pp. 312–321. Springer, Heidelberg (2004)
11. Bishop, C.M.: Neural Networks for Pattern Recognition, January 1996. Oxford University Press, USA (1996)
12. Anand, R., Mehrotra, K., Mohan, C., Ranka, S.: An improved algorithm for neural network classification of imbalanced training sets. IEEE Transactions on Neural Networks 4, 962–969 (1993)
13. Lawrence, S., Burns, I., Back, A., Tsoi, A., Giles, C.L.: Neural network classification and unequal prior class probabilities. In: Orr, G.B., Müller, K.-R. (eds.) NIPS-WS 1996. LNCS, vol. 1524, pp. 299–314. Springer, Heidelberg (1998)
14. Wilson, D.R., Martinez, T.R.: Reduction techniques for instance-based learning algorithms. Machine Learning 38(3), 257–286 (2000)

Designing RBFNNs Using Prototype Selection

Ana Cecilia Tenorio-González, José Fco Martínez-Trinidad,
and Jesús Ariel Carrasco-Ochoa

National Institute for Astrophysics, Optics and Electronics
Luis Enrique Erro No. 1, Sta. María Tonanzintla, Puebla, México,
C.P. 72840
{catanace17,fmartine,ariel}@ccc.inaoep.mx

Abstract. Performance and accuracy of a neural network are strongly related to its design. Designing a neural network involves topology (number of neurons, number of layers, number of synapses between layers, etc.), training synapse weights, and parameter selection. Radial basis function neural networks (RBFNNs) could additionally require some other parameters, for example, the means and standard deviations if the activation function of neurons in the hidden layer is a Gaussian function. Commonly, Genetic Algorithms and Evolution Strategies have been used for automatically designing RBFNNs In this work, the use of prototype selection methods for designing a RBFNN is proposed and studied. Experimental results show the viability of designing RBFNNs using prototype selection.

Keywords: Neural networks, RBFNN, prototype selection, supervised classification.

1 Introduction

Supervised classification is one of the most important problems into Pattern Recognition. Given a dataset, described by a set of attributes, for which the true class-label is known (training set). The supervised classification problem consists in designing a classifier using the training set to automatically classify unseen data. Neural networks [17, 18] are among the most widely used supervised classifiers.

Neural networks are typically formed by a set of parallel processing units (neurons) interconnected among them. Neurons are organized in layers depending of their function in the network (input, hidden or output). Each neuron has associated an activation (transfer) function used for combining its inputs and weights to produce an output value. Among the different types of neural networks, radial basis function neural networks (RBFNNs) have been widely used due to its reduced number of layers and its good results in supervised classification problems [2, 3, 5-8, 18]. A RBFNN is a three-layer feed-forward network that uses radial basis functions (normally a Gaussian), as transfer functions for the hidden layer neurons. A typical RBFNN is illustrated in Figure 1. The number of nodes in the input layer is determined by the number of attributes used for describing the data in the problem to solve. In the hidden layer, the processing is made using a nonlinear transfer function

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 189–198, 2010

(commonly a Gaussian). In the output layer there is a neuron for each class in the problem. The neurons in the output layer perform a simple weighted sum, for combining the outputs in the hidden layer, producing a linear output. During the training, the weights between the internal and the output layers are adjusted (weights between the input and the internal layers of a RBFNN are equal to 1.0, which can be interpreted as no weights). When an unseen object is presented to the network the class associated to the neuron with the highest value in the output layer is assigned to the object.

Generally, the automatic design of a RBF neural network has been done using Genetic Algorithms and Evolution Strategies [1 -7, 14, 18]. In this paper, we present a new method for designing RBFNNs for multiclass problems, based on prototype selection. Our method designs a RBFNN reducing the number of neurons in the hidden layer without affecting the classification accuracy.

The rest of the paper is organized as follows: Section 2 presents a review of the most recent methods for designing RBFNNs. Section 3 introduces the proposed method for designing RBFNNs for multiclass problems, based on prototype selection. Section 4 shows the experimental results, and Section 5 presents our conclusions and future work.

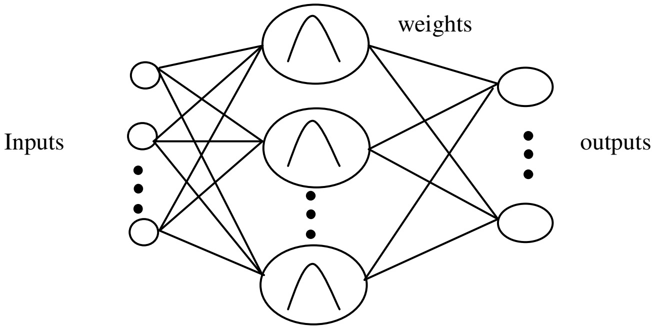


Fig. 1. RBFNN structure

2 Related Work

Most of the methods for automatic design of a RBFNN are based on Genetic Algorithms or Evolution Strategies. In [1], a review of evolutionary methods for designing neural networks is presented. In this work, the importance of a good design of a neural network is established; several evolutionary algorithms for modifying the network topology as well as network parameters are described and compared. Examples of evolutionary methods for automatic design of RBFNNs are [2-7]. From these, we have chosen the most recent [5, 7] for comparing the method proposed in this work.

In [5] a co-operative co-evolutionary algorithm (Co-CEA) is used for modifying the topology and parameters (means and standard deviations for internal neurons) of a RBFNN. First, DRSC (Decaying Radius Selection Clustering) is used to compute the initial hidden neurons based on the training set, and the cluster distribution is used for initializing the standard deviations. Then, the neurons of the hidden layer are clustered using a modification of the *k-means* algorithm. Later, each cluster is divided in overlapped subsets, which will be the individuals of a population that will evolve for each cluster. After each generation, the best individual (subset of neurons) from each cluster is selected to build the hidden layer of a RBFNN Θ^* (at the beginning a representative of each cluster is selected for Θ^*). The fitness of an individual is based on: a) the accuracy, over a validation dataset, of the RBFNN obtained by replacing in Θ^* the best individual of the corresponding cluster by the evaluated individual; and, b) how many objects, misclassified by most of the RBFNNs built for all other individuals, can be correctly classified by the RBFNN of (a). Using this fitness, a roulette wheel selection is used to select individuals for the next generation; and elitist selection is used for updating Θ^* after each generation. After a predetermined number of iterations, the last Θ^* is selected as the final estimation of the RBFNN. An extension of this work was presented in [6], where the RBFNN is designed using elliptical basis functions. In [6], a co-operative co-evolutionary algorithm is also used for designing a RBFNN but, instead of using DRSC, a random initialization of the hidden layer neurons is done.

In [7], a method for designing RBFNNs for two class problems using symbiosis is presented. Symbiosis is done by means of co-evolution of two populations. For the first population, an evolutionary method is used for obtaining a set of parameters for EvRBF[14], which, for each individual of the first population, evolves a second population for designing a RBFNN. The accuracy of the best RBFNN found by EvRBF is used as fitness of the corresponding individual of the first population. Each individual of the second population (EvRBF) is a set of neurons for the hidden layer of a RBFNN. EvRBF uses special crossover and mutation operators for evolving this population in order to modify the topology (size) and parameters (means and standard deviations) of a RBFNN. Tournament selection is done for each generation, using, as fitness, the accuracy of the RBFNN built with the neurons of an individual.

There are other methods for designing RBFNNs, which are not based on evolutionary techniques. In [9], an approach based on clustering is proposed for designing a generalized RBFNN, using clusters obtained by analyzing the input data and their expected output. In this work, the size and parameters of the network are obtained. First, the output space is clustered using *k-means*, and for each group, a fuzzy set is created. These fuzzy sets are used for clustering the input data using a context-based fuzzy clustering algorithm. Using the clusters of the input data, a generalized RBFNN is built. In [8], fuzzy clustering is used for obtaining the means of the neurons in the hidden layer of a RBFNN; nevertheless the topology of the network is not modified.

As it can be seen, most of the methods for designing RBFNNs are based on genetic algorithms or evolution strategies. Those methods that follow a different approach have been designed for a variation of the RBFNNs or only modify the parameters of a RBFNN, but not its topology. For this reason, in this paper we compare our proposed method only against methods based on genetic algorithms and evolution strategies.

3 Proposed Method

In this paper, we address the automatic design of a RBFNN through prototype selection. As we can observe in the related work, for the automatic design of a RBFNN, the majority of the works choose some instances in the training set as neurons for the hidden layer. Following this idea, for building the hidden layer, we could use one neuron for each instance in the training set; however to proceed in this way could negatively affect the network performance, since usually in the training set there are some instances which are redundant or noise. Since, the main goal of instance selection is to choose a subset of instances from the training set such that the selected subset of instances does not contain superfluous instances and, at the same time, this subset would produce as high classification accuracy as possible, we propose to use prototype selection for designing the hidden layer of a RBFNN. The hypothesis is that a RBFNN designed in this way can have similar or better results in accuracy than those results obtained by the networks designed by evolutionary approaches.

In figure 2, we show the main steps of our method, the prototype selection methods are applied to the training data in order to choose some representative instances. Then these instances are used to create the neurons in the hidden layer of the RBFNN. Finally, the network designed in this way is trained and evaluated.

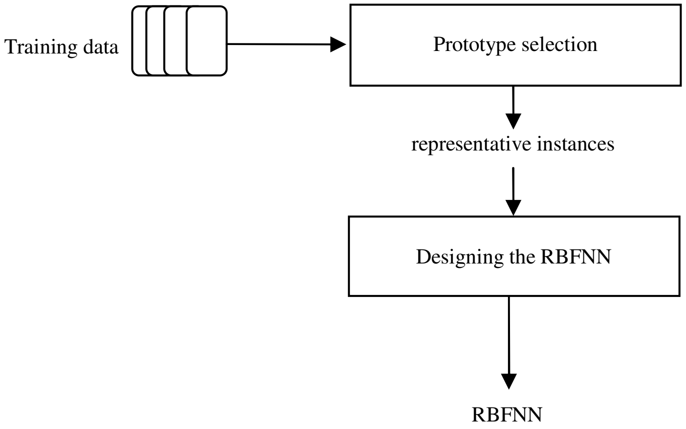


Fig. 2. Proposed Method for designing a RBFNN

The prototype selection methods that we will test for designing the hidden layer of a RBFNN are OSC, CLU and PSR [10, 11], these methods are based on clustering and they are the fastest methods for prototype selection [10]. Additionally, we include the DROP3 method which is not as fast as the others but is one of the best prototype selection methods [12].

For our study, we used a RBFNN similar to the one reported in [13], this network uses in the hidden layer a neuron for each instance in the training set. For our method we use a neuron for each instance selected by the prototype selection method. In the same way as in [13] we use as vector of means for each neuron the values of its respective instance. The standard deviation is the same for all neurons in the hidden layer. For computing the standard deviation the amount of instances in the training and the number of attributes used for describing them are used in the same way as in [13]. The network is trained using the fast training algorithm *steepest descent* [13], using negative gradient. This algorithm repeats the training a certain maximum number of times while the accuracy threshold or the maximum number of iterations is not reached.

4 Experiments

In this section we present a comparison of the size of the network (number of neurons in the hidden layer), accuracy and runtime needed by our method, for designing and training a RBFNN, against those results obtained by methods based on evolutionary approaches.

Since the authors of original implementations refuse to provide their programs, in order to compare our proposed method against previous approaches for designing RBFNNs, we implement two methods using genetic algorithms and one more using evolution strategies, based on [5] and [7, 14], respectively.

For the genetic approach, GA-RBFNN was implemented based on [5]. In this method, each individual of the population represents a RBFNN configuration (including number of neurons, means and standard deviations). The population is evolved and the best individual is selected. As fitness function, the accuracy of a RBFNN built using the individual (as it was described in section 3) over a validation dataset, is used. The one-point crossover operator is applied using a random crossover-point, and the mutation probability is obtained in a random way at the beginning of the algorithm and it remains the same along all generations. A second variant of GA-RBFNN, called GAS-RBFNN was implemented in a similar way, but the individuals represent a subset of training objects that will be used to build the hidden layer of a RBFNN.

For the evolution strategy, EvRBF [7, 14] was implemented using tournament selection and elitist replacement. The crossover operator consists in interchanging a random number of neurons of the hidden layer, and the mutation operator modifies the means and standard deviation of some neurons of this layer. Additionally, a random number of neurons can be added or eliminated. The fitness function is evaluated as an average, among the training and validation datasets, of the difference between the obtained and the expected outputs of the RBFNN built using an individual.

4.1 Databases

For our experiments we used twelve databases taken from the UCI repository [15]. The characteristics of these databases are shown in the Table 1.

4.2 Experiments

For our experiments, the prototype selection methods OSC, PSR and CLU were implemented in Matlab [19] and we used the author implementation in C of DROP3

Table 1. Databases used in the experiments

Databases	#Attr	#Class	#Inst
Glass	9	7	214
Pima	8	2	768
Yeast	8	10	1484
Lymphography	18	4	148
Primary-tumor	17	22	339
Soybean	35	19	307
Waveform	21	3	5000
Wines	13	3	178
Zoo	16	7	101
Sonar	60	2	208
Ionosphere	34	2	351
Breast-cancer	9	2	286

[12]. The RBFNN was implemented in Java 6. The methods based on genetic algorithms were implemented into the package JGAP [20] (in Java). The Evolution stetegy was implemented using the Keel platform [16]. All the methods for designing a RBFNN were executed in a PC having a Mobile AMD Sempron processor at 2 Ghz with 1GB in RAM using Windows Vista.

For all the experiments, we used k-fold cross validation with k=3. For each fold, we tried to keep the original proportion of the instances in the classes. Thus, two folds, i.e., the 66% of the objects in the database were used for training and the remaining 33% for testing.

It is important to highlight that we included the evaluation of the RBFNN built without applying prototype selection (see column Orig in tables 2, 3 and 4), i.e., using the 66% of the objects in the database as neurons in the hidden layer of the network, and the remaining 33% of the objects as testing.

For the proposed method based on prototype selection, the objects selected by OSC, PSR, CLU and DROP3 were used as neurons in the hidden layer. Then the RBFNN was trained, for adjusting the weights, using the 66% of the objects in the database. And finally the RBFNN was evaluated using the remaining 33% of objects.

For the evolutionary methods, the RBFNN was designed and trained using the 66% of the objects in the database. And finally the RBFNN was evaluated using the remaining 33% of objects.

4.3 Results

The number of neurons in the hidden layer of the RBFNNs built by each method is shown in table 2. In the penultimate row (average 1) of this table we show the average size of the designed RBFNNs excluding the two largest datasets (*Yeast* and *Waveform*) because, after 50 hours, the evolutionary methods could not build a RBFNN for these datasets. In the last row (average 2) of table 2, we show the average size of the RBFNNs designed by the methods that could design RNFNNs for all datasets. We can see that the smallest networks were designed applying the Evolution Strategies based method (ES in table 2) followed by the networks designed applying the prototype selection method DROP3 (Drop3 in table 2). However, the accuracies obtained by the networks designed by the Evolution Strategies based method were the

Table 2. Number of neurons in the hidden layer

Databases	Orig	OSC	PSR	Drop3	CLU	ES	GA	GAS
Glass	142	70	82	31	43	7	46	65
Pima	512	169	263	72	154	32	331	106
Yeast	989	421	637	169	297	61	n. a.	n. a.
Lymphography	98	43	55	28	29	4	89	33
Primary-tumor	226	121	147	31	68	3	121	144
Soybean	204	126	165	50	61	8	137	184
Waveform	3333	650	1803	494	1000	n. a.	n. a.	n. a.
Wines	118	44	70	21	35	2	95	61
Zoo	67	36	43	15	20	4	56	34
Sonar	139	44	70	34	41	9	65	116
Ionosphere	234	101	115	17	70	7	154	209
Breast-cancer	191	71	98	26	57	9	147	113
Average 1	**521.1**	**158**	**295.7**	**82.3**	**156.3**	**n. a.**	**n. a.**	**n. a.**
Average 2	**193.1**	**82.5**	**110.8**	**32.5**	**57.8**	**8.5**	**124.1**	**106.5**

Table 3. Accuracy

Databases	Orig	OSC	PSR	Drop3	CLU	ES	GA	GAS
Glass	35.52	35.52	35.51	35.52	35.52	32.75	52.80	36.90
Pima	72.66	73.17	72.65	71.62	71.62	65.75	69.27	65.90
Yeast	59.23	58.89	58.56	58.96	59.02	29.84	n. a.	n. a.
Lymphography	78.94	78.92	78.25	80.28	78.27	46.89	76.30	76.92
Primary-tumor	44.23	43.35	38.05	27.93	21.8	20.59	32.13	42.76
Soybean	87.28	87.29	71.66	67.08	71.65	11.16	77.48	85.65
Waveform	86.76	86.60	86.80	86.70	86.86	n. a.	n. a.	n. a.
Wines	66.86	66.86	67.43	66.86	66.85	63.34	70.24	66.86
Zoo	84.10	86.06	84.07	83.09	88.08	61.11	94.09	85.09
Sonar	90.39	86.05	80.29	84.13	84.13	68.27	77.43	87.99
Ionosphere	87.46	82.62	82.62	88.88	86.61	66.66	80.91	88.03
Breast-cancer	72.36	73.06	72.36	74.11	73.76	68.20	73.43	72.71
Average 1	**72.15**	**71.53**	**69.02**	**68.76**	**68.68**	n. a.	n. a.	n. a.
Average 2	**71.98**	**71.29**	**68.29**	**67.95**	**67.83**	**50.47**	**70.41**	**70.88**

worst. The accuracies obtained by the networks designed applying the prototype selection method DROP3 were better but they were not the best.

Table 3 shows the accuracy of the RBFNN built by each method for each dataset. In the penultimate row (average 1) of this table we show the average accuracy of the designed RBFNNs excluding the two largest datasets (*Yeast* and *Waveform*) because, after 50 hours, the evolutionary methods could not build a RBFNN for these datasets. In the last row (average 2) of table 3, we show the average accuracy of the RBFNNs designed by the methods that could design RNFNNs for all datasets. Among the method for designing a RNFNN, we can see in table 3 that the best results in accuracy were obtained when the network was designed without applying prototype selection. The second best results in accuracy were obtained by our proposed method using the prototype selection method OSC. The networks designed by the genetic algorithm based methods obtained accuracies close to the obtained by our method, however genetic algorithm based methods require a lot of time for designing the network. Besides, the time needed by the method that does not apply prototype selection is clearly longer than the time needed by our proposed method.

Table 4 shows the runtimes for designing the RBFNN by each method for each dataset. In the last row of table 4, we show the total runtime needed for each method for designing the RBFNNs for all datasets. In table 4, the columns ES, GA and GAS have some cells where appear >50, it means that we stop the method when it reached 50 hours. In the columns ES, GA and GAS instead of the total time appears >58 h, >117 h and >112 h respectively since for some databases these methods were stopped when they reached 50 hours, thus for computing the total we used in those databases 50 hours. The method that needed the least amount of time was our method using the prototype selection method DROP3. The second best method was our method using the prototype selection method OSC. However, it is important to remark that, among these methods the method using OSC is the one that obtains the best accuracies.

Taking into account all the above remarks, we can conclude that our method using OSC is the best option for designing a RBFNN since this method designs networks with the best accuracies, which have a size (number of neurons) lesser than the size of the networks designed by genetic algorithm based methods (the most used methods for designing RBFNNs). Additionally, our method requires much shorter time for designing a RBFNN than genetic algorithm based methods.

Table 4. Runtime

Databases	Orig	OSC	PSR	Drop3	CLU	ES	GA	GAS
Glass	00:01:51	00:01:01	00:01:00	00:00:26	00:00:36	00:08:00	00:18:56	00:24:40
Pima	00:09:43	00:03:20	00:04:37	00:01:19	00:02:45	01:10:02	01:48:05	01:45:55
Yeast	08:20:36	3:01:18	04:34:18	00:59:53	01:50:25	05:24:02	>50 h	>50 h
Lymphography	00:00:25	00:00:15	00:00:12	00:00:10	00:00:12	00:05:01	00:08:41	00:07:05
Primary-tumor	00:22:30	00:19:10	00:14:02	00:03:18	00:06:53	00:07:00	04:40:51	03:40:50
Soybean	00:14:05	00:09:09	00:11:40	00:03:24	00:04:13	00:17:00	02:55:11	04:23:54
Waveform	41:29:44	05:28:20	18:02:35	03:47:23	08:42:28	>50 h	>50 h	>50 h
Wines	00:00:28	00:00:14	00:00:18	00:00:09	00:00:12	00:04:00	00:12:20	00:12:10
Zoo	00:00:21	00:00:11	00:00:15	00:00:08	00:00:10	00:03:01	00:09:30	00:9:51
Sonar	00:00:26	00:00:11	00:00:15	00:00:10	00:00:12	00:13:03	06:45:06	00:30:28
Ionosphere	00:01:26	00:00:18	00:00:44	00:00:10	00:00:29	00:27:00	00:26:29	00:52:11
Breast-cancer	00:00:53	00:00:21	00:00:29	00:00:10	00:00:18	00:09:01	00:18:53	00:21:48
Total time	**50:42:28**	**9:03:48**	**23:10:25**	**4:56:40**	**10:48:29**	**>58 h**	**>117 h**	**>112 h**

5 Conclusions

In this paper, we propose a method for the automatic design of a RBFNN based on prototype selection.

Based on the experimental results we can conclude that the proposed method, using OSC as prototype selection method, is the best option for designing a RBFNN since this method designs networks with the best accuracies, which have a size (number of neurons) smaller than the size of the networks designed by genetic algorithm based methods (the most used methods for designing RBFNNs). Additionally, our method requires much shorter time for designing a RBFNN than genetic algorithm based methods.

As future work, we will include in our method some strategies to simultaneously adjust the parameters of a RBFNN.

References

1. Yao, X.: Evolving artificial neural networks. Proceedings of the IEEE 87(9), 1423–1447 (1999)
2. Tian, J., Li, M., Chen, F.: A Cooperative Coevolution Algorithm of RBFNN for Classification. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 809–816. Springer, Heidelberg (2007)
3. Qin, Z., Chen, J., Liu, Y., Lu, J.: Evolving RBF Neural Networks for Pattern Classification. In: Hao, Y., Liu, J., Wang, Y.-P., Cheung, Y.-m., Yin, H., Jiao, L., Ma, J., Jiao, Y.-C. (eds.) CIS 2005. LNCS (LNAI), vol. 3801, pp. 957–964. Springer, Heidelberg (2005)
4. Chen, Y., Yang, B., Zhou, J.: Automatic Design of Hierarchical RBF Networks for System Identification. In: Yang, Q., Webb, G. (eds.) PRICAI 2006. LNCS (LNAI), vol. 4099, pp. 1191–1195. Springer, Heidelberg (2006)
5. Tian, J., Li, M., Chen, F.: Improving multiclass pattern recognition with a co-evolutionary RBFNN. Pattern Recognition Letters 29(4), 392–406 (2008)
6. Tian, J., Li, M., Chen, F.: A hybrid classification algorithm based on co-evolutionary EBFNN and domain covering method. Neural Computing & Applications 18(3), 293–308 (2009)
7. Parras-Gutierrez, E., Rivas, V.M., Del Jesus, M.J.: Automatic Neural Net Design by Means of a Symbiotic Co-evolutionary Algorithm. In: Corchado, E., Abraham, A., Pedrycz, W. (eds.) HAIS 2008. LNCS (LNAI), vol. 5271, pp. 140–147. Springer, Heidelberg (2008)
8. Mu, S., Tian, S., Yin, C.: A Novel Radial Basis Function Neural Network Classifier with Centers Set By Cooperative Clustering. International Journal of Fuzzy Systems 9(4), 205–211 (2007)
9. Pedrycz, W., Park, H.S., Oh, S.K.: A granular-oriented development of functional radial basis function neural networks. Neurocomputing 72(1-3), 420–435 (2008)
10. Olvera-Lopez, A., Carrasco-Ochoa, J.A., Martinez-Trinidad, J.F.: Object Selection Based on Clustering and Border Objects. Computer Recognition Systems 2, Advances in Soft Computing 45, 27–34 (2007)
11. Lumini, A., Nanni, L.: A clustering method for automatic biometric template selection. Pattern Recognition 39(3), 495–497 (2006)
12. Wilson, D.R., Martinez, T.R.: Reduction Techniques for Instance-Based Learning Algorithms. Machine Learning 38(3), 257–286 (2000)
13. Looney, C.G.: Pattern Recognition Using Neural Networks. In: Theory and Algorithms for Engineers and Scientists. Oxford University Press, Oxford (1997)
14. Rivas, V.M., Merelo, J.J., Castillo, P.A., Arenas, M.G., Castellano, J.G.: Evolving RBF neural networks for time-series forecasting with EvRBF. Information Sciences 165(3-4), 207–220 (2003)
15. Blake, C., Keogh, E., Merz, C.J.: UCI repository of machine learning databases, Department of Information and Computer Science, University of California, Irvine, CA (1998), http://www.ics.uci.edu/~mlearn/MLRepository.html
16. Alcalá-Fdez, J., Sánchez, L., García, S., del Jesus, M.J., Ventura, S., Garrell, J.M., Otero, J., Romero, C., Bacardit, J., Rivas, V.M., Fernández, J.C., Herrera, F.: KEEL: A Software Tool to Assess Evolutionary Algorithms to Data Mining Problems. Soft Computing 13(3), 307–318 (2009)

17. Haykin, S.: Neural Networks: a comprehensive foundation, 2nd edn. Prentice Hall, Englewood Cliffs (2005)
18. Konar, A.: Computational Intelligence: principles, techniques, and applications. Springer, Heidelberg (2005)
19. The MathWorks Inc., Natick (1994-2008), `http://www.mathworks.com`
20. Meffert, Klaus, et al.: JGAP - Java Genetic Algorithms and Genetic Programming Package, `http://jgap.sf.net`

A Learning Social Network with Recognition of Learning Styles Using Neural Networks

Ramón Zatarain-Cabada[1], M.L. Barrón-Estrada[1], Viridiana Ponce Angulo[1], Adán José García[1], and Carlos A. Reyes García[2]

[1] Instituto Tecnológico de Culiacán, Juan de Dios Bátiz s/n, Col. Guadalupe,
Culiacán Sinaloa, 80220, México
[2] Instituto Nacional de Astrofísica, Óptica y Electrónica (INAOE)
Luis Enrique Erro No. 1, Sta. Ma. Tonanzintla, Puebla, 72840, México
{rzatarain,lbarron,vponce,rcabada}@itculiacan.edu.mx,
kargaxxi@inaoep.mx

Abstract. The implementation of an adaptive learning social network to be used as an authoring tool, is presented in this paper. With this tool, adaptive courses, intelligent tutoring systems and lessons can be created, displayed and shared in collaborative and mobile environments by communities of instructors and learners. The Felder-Silverman model is followed to tailor courses to the student's learning style. Self Organizing Maps (SOM) are applied to identify the student's learning style. The introduction of a social learning network to create, view and manage adaptive intelligent tutoring systems, and a novel method to identify the student's learning style, are the contributions of this paper.

Keywords: Adaptive mobile learning, Social learning networks, Authoring tools, Learning Styles, SOM.

1 Introduction

New technologies as Web 2.0 applications have appeared during the last recent years. These novel technologies besides the retrieval of information allow to implement and to hold its control. Moreover, users can perform *harnessing Collective Intelligence [1]* by adding more value to the information as they make use of it. Several well known Web 2.0 applications are, **YouTube**, **Wikipedia**, **Yahoo! Answers**, and **Digg.** As a matter of fact, the distinguishing technologies of Web 2.0 are Social Network Sites. A study identifying social learning networks and analyzing seven e-learning 2.0 systems (EctoLearning, Edu 2.0, eLearningCommunity *2.0, LearnHub, LectureShare, Nfomedia, Tutorom (Eduslide)),* was presented by Ivanova in [2].

According to Murray et al [3] by 2003 over twenty authoring tools for Intelligent Tutoring Systems had been developed. Authoring tools are classified according to the type of tutoring system they help to produce; for example the author tool SIMQUEST [4] produces “simulation-based learning” systems, IRIS [5] creates “multiple knowledge types” systems, and InterBook [6] generates Intelligent/adaptive Hypermedia.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 199–209, 2010

A common denominator in all the named tools is that they are designed only for authors without taking in count the learner.

With the idea of giving learners a more protagonist role, many efforts have been done recently to capture the way students learn in order to model their learning style [7, 8, 9, 10]. The ILSQ questionnaire is used to calculate learning styles with the support of Bayesian Networks, Linear Temporal Logic, or Neuro-Fuzzy Systems. Neither of these reported systems are authoring tools.

On the other side, some authoring tools have been developed with the capability of implementing mobile applications; among them are MyLearning [11], Test Editor [12], or mediaBoard [13]. In these systems, PocketPC orientation, quiz editing or game-based learning are some of the approaches taken. All of them are author tools without the capability to be adapted to the student learning style, and cannot be interchanged among different operating system platforms.

Trying to overcome the particular limitations inherent to each kind of the described tools, we have developed an adaptive learning social network tool named Zamná. The tool is implemented with the regular capabilities of social networks, like new users register, profiles and communities' creation, etc, and with the capabilities of intelligent learning, dynamically adapting, and visualization in mobile devices. A new methodology to identify learning styles, based on Self-Organizing Maps is also introduced. This process is executed whenever a tutoring system is displayed, either in our system or in a mobile device. In general, our system is integrate with components coming from different domains like; Social Learning Networks, Mobile Learning, Adaptive or Web-based Systems, Artificial Neural Networks, and Intelligent Tutoring Systems.

This paper is organized as follows: Section 2 is devoted to explain the Zamná Architecture. In Section 3 we describe the Predictive Engine, and the neural network training and testing. In Section 4 some resulting products are shown. And in Section 5 we present our Conclusions and propose some Future work.

2 Learning Styles and Zamná Architecture

A Learning Style model categorizes both the ways in which students learn, and how teachers teach students. There are several proposals on how to approach this problem; to implement Zamná we followed the Felder-Silverman model. The Felder-Silverman model was proposed by Richard Felder and Linda Silverman [15] and includes four dimensions or categories. The four included dimensions are perception (sensory/intuitive), processing (active/reflexive), submission of entry (visual/verbal), and understanding (sequential/global). With Zamná the learning material can be built and adapted to the identified learning style of each learner.

In Figure 1 we describe the architecture of the Web 2.0 tool Zamná. As can be noticed from the diagram, users have access to Zamná through any browser. For any access the user workspace will compose several components, which are; a news section, a user profile, an inbox part, a section of courses, communities, documents,

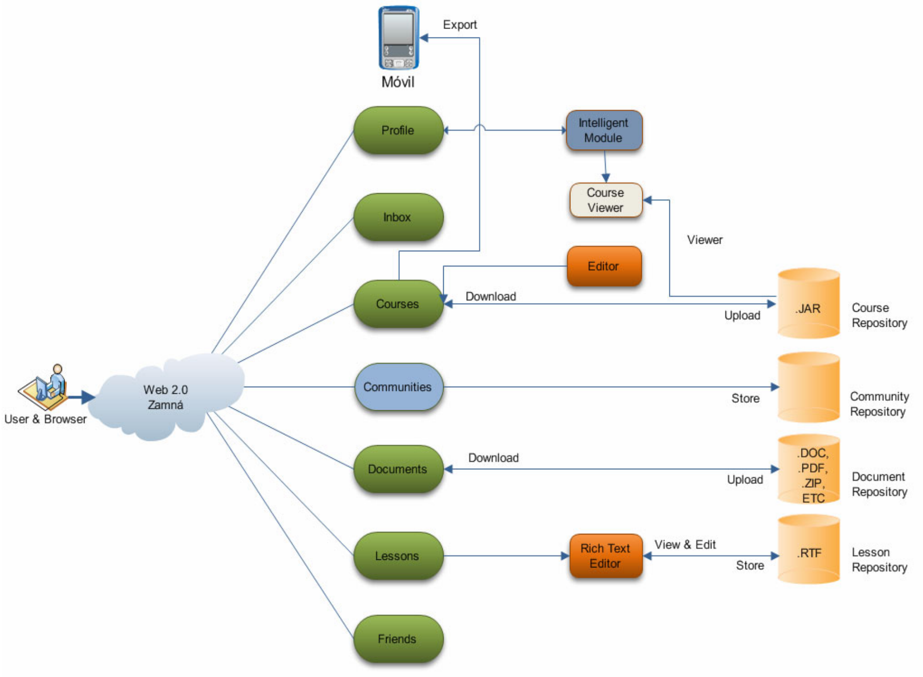


Fig. 1. Zamná Architecture

lessons and friends. To compose the component *profile* Zamná makes use of the intelligent module to identify the user's learning style. The intelligent module makes use of a course viewer for displaying the contents of a course according to the learning style identified by the intelligent module. The course materials are stored in data bases or repositories from where they are downloaded to be exported and studied on any mobile device. Another component is Co*mmunities* which is composed of small sets of networks, each of which is a community focused on a particular area of knowledge for specific purposes. Te communities are also stored in an assigned repository. The same description holds for the component *Lessons*.

2.1 How to Build a Course or ITSs

To implement an adaptive intelligent system, three steps have to be followed, as shown in Figure 2. Step1. A tree structure of the adaptive or intelligent tutoring system should be designed by the main instructor(s). On the tree structure, the designer inserts examinations in the form of quizzes (multiple selection and choice). Quizzes are important elements which provide adapting capabilities to the produced tutorials.

Step 2. The tree structure is filled up with the corresponding domain contents, and other learning resources. During the creation, the instructor or teacher compose the

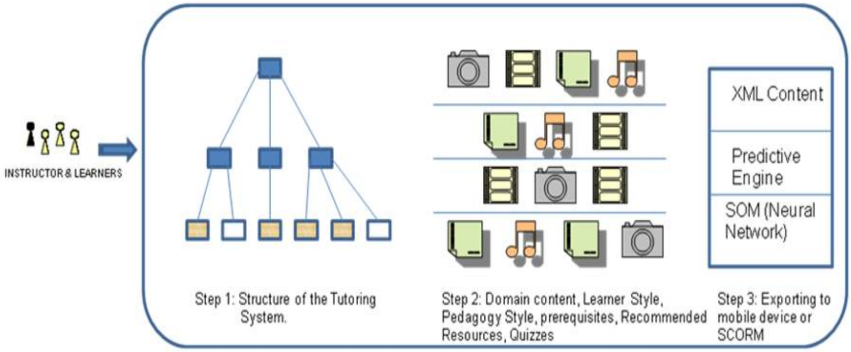

INSTRUCTOR & LEARNERS
XML Content
Predictive Engine
SOM (Neural Network)
Step 1: Structure of the Tutoring System.
Step 2: Domain content, Learner Style, Pedagogy Style, prerequisites, Recommended Resources, Quizzes
Step 3: Exporting to mobile device or SCORM

Fig. 2. Building a Tutoring System

tutoring system by the introduction of learning objects like text fragments, images, audio/video clips, and by defining learning styles, prerequisites, tags and quizzes. Further on, more learning resources can be included by the same learners, who can also recommend resources they find regularly on the web. Step 3. This step is for saving/exporting packages containing: the learning resources or contents (an XML file), a predictive engine for navigation purposes**,** and a SOM Neural Network for the classification of learning styles.

3 The Predictive Engine

The predictive engine is designed to dynamically identify the student´s learning style every time a tutoring system is run. After the corresponding learning style is identified, and in combination with the student profile, the course contents are selected by an interpreter in the form of learning objects. At any time, the learning style can be adapted as a result of the evaluation applied to the student.

3.1 Identifying Learning Styles

Based on the nature of our problem, for the selection of the learning classifier we needed one with; unsupervised learning abilities, good proved performance, and fast training potential. One model with such capabilities is the Kohonen´s Self-Organizing Maps SOM, which is the one we choose. The identification of the student´s learning styles is a pedagogy discipline, and we have thought that SOM will do the role of an always available pedagogue performing such a task.

3.2 SOM Configuration and the Input Layer

The input data are provided as input signals, which are part of the training data space. The signals are vectors with three components; two are vectors as well, and the other is a scalar value. A description of signal X is described by Eq. 3.1, where the vector d_{FS} represents the student's learning style, which is identified by the application of the questionnaire titled *Learning Styles Inventory* proposed by Felder-Soloman [14]. Vector, d_c, is the learning style used to previously design the learning material which will be studied by the student. And component p stands for the student´s performance, having a learning style d_{FS} and studying a course designed with the configuration for learning styles d_c. Stated in another way, p is taken as the grade of the student in a course offered in a learning style d_c where the student has a learning style orientation $d_{FS..}$ Values $d_{FS,}$ d_c, *and p* were obtained from 47 students in three different courses, previously to the network training.

$$X = [\mathrm{d_{FS}}\ \mathrm{d_c}\ \mathrm{p}] \tag{3.1}$$

A detailed description of vectors d_{FS} and d_c is given by Eq. 3.2. As can be seen, both vectors have three elements, each of which represents a dimension of the learning styles identified by Felder-Silverman. The value of each element represents the student's proclivity toward a specific learning style. For the processing of these two vectors plus the p value, the SOM´s topology was implemented as follows; there are 7 nodes in the input layer, while the Kohonen layer is formed by 1600 neurons which are organized in a lattice composed by 40x40 neurons.

$$\mathrm{d_c} = \mathrm{d_{FS}} = [\mathrm{c_1}\ \mathrm{c_2}\ \mathrm{c_3}] \tag{3.2}$$

The SOM designed is then trained to identify learning styles of students. In Eq. 3.3, the structure of the output vector as obtained from the neural network after training or testing is described. The output from the SOM neural network represents the student's learning style.

$$D = [\mathrm{d_c}\ \mathrm{p}] \tag{3.3}$$

For our experiments the programming language used for the neural network implementation was Java.

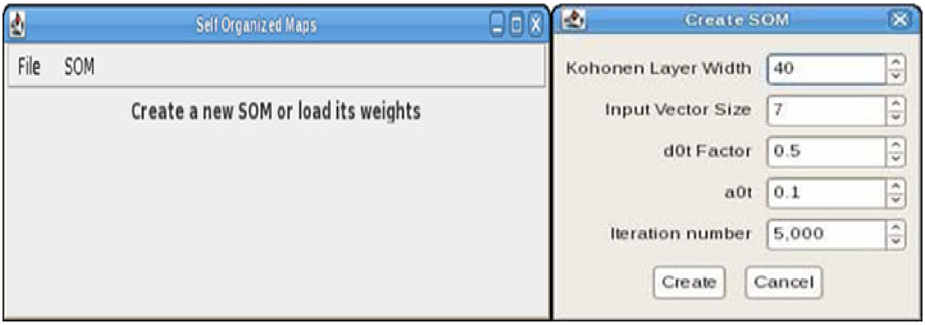
3.3 A Framework for Training and Testing the SOM

In order to be able to export the trained SOM to mobile devices or web-based learning systems supported by an interpreter to provide intelligent tutoring, a framework allowing the creation, training and testing/validating self-organizing maps is implemented. The framework and the interpreter were developed in a standard and free platform that is Java. From the beginning of this research, another important goal was to have a framework to allow us to implement universal neural networks to be adapted to recognize diverse learning styles from different learning models or theories. As a first effort we are here testing the model proposed by Felder-Silverman, and we are currently working with the multiple intelligences model developed by Gardner.

The experimental training procedure is performed in two stages. The first stage is devoted to the collection of teaching materials. To start testing the system we collected material for three high school courses with the help of high school teachers. These courses were *basic computing* (history, parts of a computer, computer technology and resources for exchanging information with the PC, etc.), w*ind energy* (wind energy story, basic constitution of a wind turbine, wind turbine types, etc.) and *photography* (Introduction, purpose, the film compartment, the shutter, etc.). The teaching material to be added to the intelligent tutorial system was prepared in eight different versions, by following the Felder-Silverman theory. Each version corresponds to each learning style described by the theory. They are the result of the combination of visual-verbal, sensitive-intuitive and sequential-global dimensions.

At this step, we did not consider dimension processing, mainly due to the limitations associated to the nature of mobile and electronic learning. For our study, one version of the course was randomly given to each selected student. Then, they had to study the provided materials during a period of approximately 40 minutes. After the study period the students had to answer an examination, on the studied subject, composed of 20 multiple choice questions. For the same course all the selected students were evaluated with the same assessment. From the results obtained we calculated the performance *p* of each student. To identify and register initial learning styles the test *Questionnaire Learning Styles Inventory* was also applied to each student. The initial learning style of any student can change, with the use or, while the student is using an intelligent tutoring system (through the mobile). The expected change is to the student´s actual learning style which should be identified by the ITS through the learning process. For the initial testing we obtained data from 47 randomly selected high school students. The second stage is for the SOM training with the input data obtained in the previous step (student tests). The number of iterations for training the neural network, heuristically selected, was set to 5000.

Two other values, heuristically determined also, were; initial learning rate, with a value of 0.1 and the size of the initial neighborhood, with a fixed value of 20.



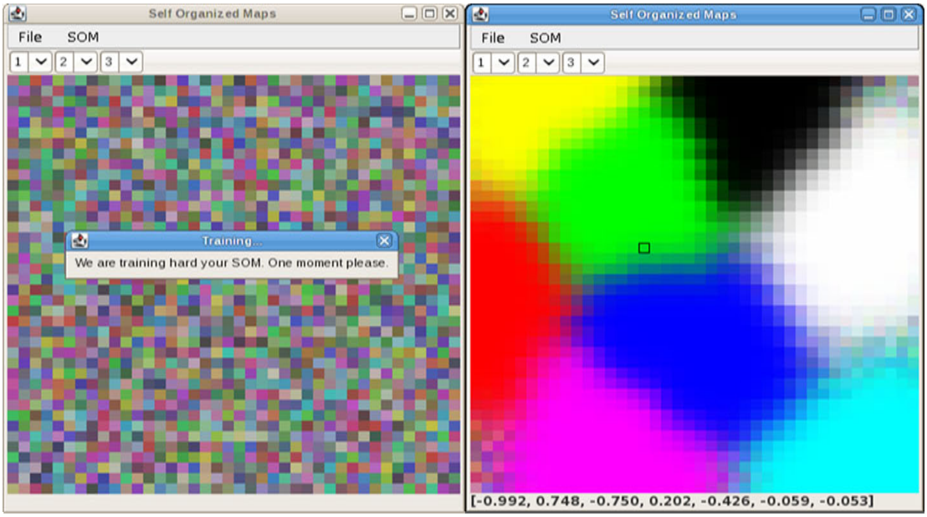
Create a new SOM or load its weights

Kohonen Layer Width 40
Input Vector Size 7
d0t Factor 0.5
a0t 0.1
Iteration number 5,000

Fig. 3. Initial and Dialog Window in the Framework

The 40x40 neurons in the *Kohonen layer* of the SOM are shown as a small colored rectangle in the left side of Figure 4.While on the right side of the same figure, the eight zones identified, at the end of training, in the Kohonen layer and corresponding to the eight combinations of learning styles are shown.

After the SOM has been trained we apply a validation process, for which we use 7 parameter vectors, where 4 parameters are used to identify learning styles (three d_c items to define the style of learning of the material provided to the student, and p which represents the student performance with that teaching material). Another 3 parameters will contain the outcome response from the neural network. Table one shows some values for the input vectors with different parameters, along with expected values for each vector, used during training, and obtained results, which are the actual output of the network for each input vector.



We are training hard your SOM. One moment please.

```
[-0.992, 0.748, -0.750, 0.202, -0.426, -0.059, -0.053]
```

Fig. 4. The Kohonen Layer before and after Training

Table 1. Test Results in the training process of the neural network

INPUT VECTORS				EXPECTED RESULTS			OBTAINED RESULTS		
-1	+1	-1	+0.6	-.8181	-.2727	-.0909	-.6315	-.1447	-.3863
-1	-1	-1	+0.8	-1.000	+.0909	+.2727	-.4896	-.1071	-.0176
-1	+1	+1	+0.4	-.4545	-.0909	-.2727	-.0919	+.1137	-.4328
+1	+1	-1	+0.4	-.8181	+.8181	+.2727	-.3933	-.0844	-.4320
+1	+1	+1	+0.3	+.0909	-.0909	-.4545	-.2765	-.0365	-.4807
+1	-1	-1	+0.3	-.8181	-.0909	-.0909	-.7461	+.0429	+.2300
-1	-1	+1	+0.2	-.4545	+.0909	-.2727	-.7392	-.1687	-.3382
-1	+1	+1	+0.4	-.4545	-.0909	-.2727	-.0919	+.1137	-.4328
+1	+1	+1	+0.3	-.0909	+.2727	+.0909	-.3956	-.4314	+.4301
+1	+1	-1	+0.5	-.2727	+.2727	-.6363	-.5336	+.1870	-.3271
+1	-1	+1	0.0	-.2727	+.4545	-.2727	-.3730	+.0481	+.2212
-1	+1	+1	+0.8	+.0909	+.0909	-.0909	-.3133	-.0021	-.3617

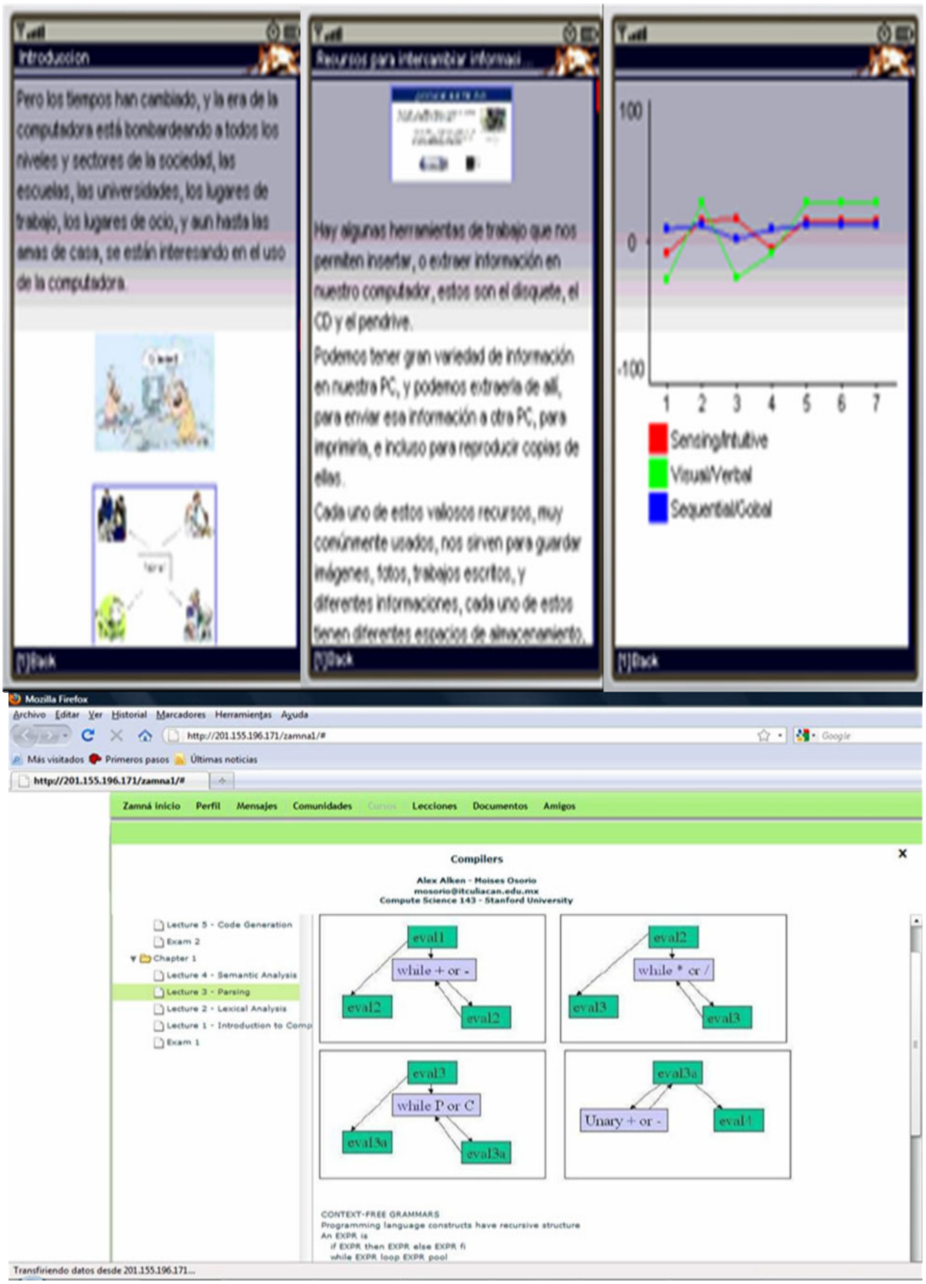
3.4 Results Analysis

Each of the three elements in vector d_c, which is part of the input vector, represents one of the scales of the Felder-Silverman model. These scales in the input vector are codifies as follows: the first element corresponds to the scale *Visual/Verbal*, the second to the scale *Sensitive/Intuitive* and the third to the scale *Sequential/Global.* For example, from the first input vector shown in Table 1, we obtain a learning style configuration corresponding to an intelligence of the type *Visual-Intuitive-Sequential* (-1 +1 -1). In the output vectors the first element is Visual/Verbal scale, the second one is Sensitive/Intuitive scale and the third one is Sequential/Global scale. Similarly, when selecting the first vector of the set of expected results, (-0.8181 -0.2727 -0.0909), it is possible to infer the learning style that the neural network reports as a result (Visual-Sensitive-Sequential). From the results obtained it is possible to carry out an analysis at different levels. For example, if we analyze the hard numbers of the results, which indicate the resulting size for each scale, it can be observed that 16.66% of the results obtained are consistent with the expected results on all scales, and 66.66% in at least two of the three scales. The same values can be interpreted in the domain of the Felder and Soloman [14] learning styles classification. In this case, the numerical values are substituted by linguistic labels associated to the preferences in each dimension (*strong* with a difference between -1 to -.666, *weak* with a difference between -.666 to -.333 and *almost nonexistent* with a difference between -.333 to 0).

According to the results in Table 1, we can notice that 83.3% of the output results are consistent with the expected ones. The distance with a perfect 100 % score is due to considering that no matter which dimension of the scale the student has a preference for, if it is balanced, the preference might vary from one or another dimension. Thus a student with a balanced degree of preference for the visual dimension of the scale Visual/Verbal, as time goes on he can easily switch to a balanced preference for the other dimension (Verbal) of the same scale. Felder and Solomon described the meaning of the adjectives strong, moderate and balanced, related to the degree of student preference for one dimension of each scale.

4 SOM and Intelligent Tutoring Systems

The SOM network was tested with the production of some courses to be displayed on cell phones or on the social network Zamná. Figure 5 presents several pictures of a small tutoring system for the topic *Eolic Energy* displayed in a cell phone (first three pictures) and a *compiler course* displayed in the Zamná Social Network Site (fourth picture). Two of the mobile phones display a sample of the learning material for a course in computer networks and the third shows a trace of the students' learning styles (in three and 7 stages) along the course.



Introduccion
[1]Back
Recursos para intercambiar informaci...
[1]Back
100
0
-100
1
2
3
4
5
6
7
🟥 Sensing/Intuitive
🟩 Visual/Verbal
🟦 Sequential/Global
[1]Back
Mozilla Firefox
Archivo Editar Ver Historial Marcadores Herramientas Ayuda
http://201.155.196.171/zamna1/#
Google
Más visitados Primeros pasos Últimas noticias
http://201.155.196.171/zamna1/#
Zamná inicio
Perfil
Mensajes
Comunidades
Cursos
Lecciones
Documentos
Amigos
Lecture 5 - Code Generation
Exam 2
Chapter 1
Lecture 4 - Semantic Analysis
Lecture 3 - Parsing
Lecture 2 - Lexical Analysis
Lecture 1 - Introduction to Comp
Exam 1
eval1
while + or -
eval2
eval2
eval2
while * or /
eval3
eval3
eval3
while P or C
eval3a
eval3a
eval3a
Unary + or -
eval4
Transfiriendo datos desde 201.155.196.171...

Pero los tiempos han cambiado, y la era de la computadora está bombardeando a todos los niveles y sectores de la sociedad, las escuelas, las universidades, los lugares de trabajo, los lugares de ocio, y aun hasta las amas de casa, se están interesando en el uso de la computadora.

Hay algunas herramientas de trabajo que nos permiten insertar, o extraer información en nuestro computador, estos son el disquete, el CD y el pendrive.

Podemos tener gran variedad de información en nuestra PC, y podemos extraerla de ahí, para enviar esa información a otra PC, para imprimirla, e incluso para reproducir copias de ellas.

Cada uno de estos valiosos recursos, muy comúnmente usados, nos sirven para guardar imágenes, fotos, trabajos escritos, y diferentes informaciones, cada uno de estos tienen diferentes espacios de almacenamiento.

Compilers

Alex Alken - Moises Osorio
mosorio@itculiacan.edu.mx
Compute Science 143 - Stanford University

CONTEXT-FREE GRAMMARS
Programming language constructs have recursive structure
An EXPR is
  if EXPR then EXPR else EXPR fi
  while EXPR loop EXPR pool

Fig. 5. Intelligent Courses for Computer Networks and Compilers

5 Conclusions and Future Work

The results from the performed evaluations of our designed tool show that the current version behaves as expected, showing a strong potential as an ITS authoring/learning tool. At present, our system continues being tested, evaluated and improved. We plan to test the site with groups of students from diverse levels and different areas of study. In addition, we want to test the courses created for mobile devices by following their impact and results of studying actual courses. In this case we plan to first analyze the forms to provide the learning material that is most suitable to be managed in these mobile devices. The IP address of Zamná site is http://201.155.196.171/zamna/.

Acknowledgments. The work described in this paper is fully supported by a grant from the DGEST (Dirección General de Educación Superior Tecnológica) in México.

References

1. O'Reilly, T.: What is Web 2.0, `http://oreilly.com/pub/a/oreilly/tim/news/2005/09/30/what-is-web-20.html`
2. Ivanova, M.: Knowledge Building and Competence Development in eLearning 2.0 Systems. In: I-KNOW'08, Graz, Austria, September 3-5, pp. 84–91 (2008)
3. Murray, T., Blessing, S., Ainsworth, S.: Authoring Tools for Advanced Technology Learning Environments. Kluwer Academic Publishers, Dordrecht (2003)
4. Jong, T., de Limbach, R., Gellevij, M., Kuyper, M., Pieters, J., Joolingen, W.R.: Cognitive tools to support the instructional design of simulation-based discovery learning environment: the SIMQUEST authoring system. In: Plomp, T., van den Akker, J., Nieveen, N., Gustafson, K. (eds.), pp. 215–224. Kluwer Academic Publishers, The Netherlands (1999)
5. Arruarte, A., Fernández, I., Ferrero, B., Greer, J.: The IRIS Shell: How to build ITSs from Pedagogical and Design Requisites. International Journal of Artificial Intelligence in Education 8, 341–381 (1997)
6. Brusilovsky, P., Schwarz, E.: Web-based education for all: A tool for developing adaptive courseware. Computer Networks and ISDN Systems 30(1-7), 291–300 (1998)
7. Carmona, C., Castillo, G., Millán, E.: Designing a Bayesian Network for Modeling Student's Learning Styles. In: Díaz, P., Kinshuk, Aedo, I., Mora, E. (eds.) ICALT 2008, pp. 346–350. IEEE Computer Society, Los Alamitos (2008)
8. Graf, S., Kinshuk, Liu, T.: Identifying Learning Styles in Learning Management Systems by Using Indications from Students' behavior. In: Díaz, P., Kinshuk, Aedo, I., Mora, E. (eds.) ICALT 2008, pp. 482–486. IEEE Computer Society, Los Alamitos (2008)
9. Limongelli, C., Sciarrone, F., Vaste, J.: LS-PLAN: An Effective Combination of Dynamic Courseware Generation and Learning Styles in Web-based Education. In: Nejdl, W., Kay, J., Pu, P., Herder, E. (eds.) AH 2008. LNCS, vol. 5149, pp. 133–142. Springer, Heidelberg (2008)
10. Zatarain-Cabada, R., Barrón-Estrada, M.L., Sandoval, G., Osorio, M., Urías, E., Reyes-García, C.A.: Authoring Neuro-fuzzy Tutoring Systems for M and E-Learning. In: Aguirre, A.H., Borja, R.M., Reyes-García, C.A. (eds.) MICAI 2008. LNCS (LNAI), vol. 5317, pp. 789–796. Springer, Heidelberg (2008)

11. Attewell, J.: Mobile technologies and learning: A technology update and mlearning project summary. Learning and Skills Development, `http://www.m-learning.org/reports.shtml`
12. Romero, C., Ventura, S., Hervás, C., De Bra, P.: An Authoring Tool for Building Both Mobile Adaptable Tests and Web-Based Adaptive or Classic Tests. In: Wade, V.P., Ashman, H., Smyth, B. (eds.) AH 2006. LNCS, vol. 4018, pp. 203–212. Springer, Heidelberg (2006)
13. Attewell, J.: From Research and Development to Mobile Learning: Tools for Education and Training Providers and their Learners, `http://www.mlearn.org.za/CD/papers/Attewell.pdf`
14. Felder, R.M., Solomon, B.A.: Index of Learning Styles Questionnaire, `http://www.engr.ncsu.edu/learningstyles/ilsweb.html`
15. Felder, R.M., Silverman, L.K.: Learning and Teaching Styles in Engineering Education. Engineering Education 78, 674–681 (1988)

On-line Signature Verification Based on Modified Dynamic Time Warping and Wavelet Sub-band Coding

Juan Carlos Sánchez-Diaz[1], Juan Manuel Ramírez-Cortes[1],
Rogerio Enriquez-Caldera[1], and Pilar Gomez-Gil[2]

[1] Department of Electronics; National Institute of Astrophysics, Optics and Electronics.
Luis Enrique Erro No. 1 Tonantzintla, Puebla. 72840. Mexico
[2] Department of Computer Science, National Institute of Astrophysics,
Optics and Electronics. Luis Enrique Erro No. 1 Tonantzintla, Puebla. 72840. Mexico
karnaught@hotmail.com, jmram@inaoep.mx, rogerio@inaoep.mx,
pgomez@inaoep.mx

Abstract. This paper presents an on-line signature biometric system based on a modified Dynamic Time Warping (DTW) algorithm applied to the signature wavelet coefficients. The modification on DTW relies on the use of direct matching points information (DMP) to dynamically adapt the similarity measure during the matching process, which is shown to increase the verification success rate. The wavelet analysis is done using a sub-band coding algorithm at global and local level. The use of wavelet coefficients showed a considerable reduction in processing time and an improvement in the equal error recognition rate (EER). The system was tested using a locally constructed database. A comparison of the ROC curves obtained in each case is presented.

Keywords: Signature, online, verification, dynamic time warping, wavelet.

1 Introduction

Automatic personal identification and verification based on biometrics has received extensive attention in past years. Biometric identification refers to identifying an individual based on physiological or behavioral characteristics. It associates/disassociates an individual with a previously determined identity based on how one is or what one does. Identification can be in the form of verification, which entails authenticating a claimed identity, or recognition, which entails determining the identity of a given person from a database of persons known to the system [1].

A biometric system aims to provide automatic recognition of an individual based on features or characteristics unique to each human being. Biometric systems are based on several modalities, such as iris, face, ear shape, hand-shape, fingerprints, palm prints [2-5], or dynamical features like gait, on-line signature verification [6-7], or combination of them [8]. Requirements, strengths, and weaknesses of each modality have been widely reported in the literature.

Among the different existing forms of biometrics, signature-based verification has the advantage that signature analysis requires no invasive measurements and it is

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 210–219, 2010

widely accepted since signature has long been established as the most popular mean for personal verification in a variety of contexts, including commerce applications, banking transactions, legalization of contracts, and others. Signature is a behavioral biometric, which means that it is not based on physical properties of the individual, such as face, hand-shape, or fingerprint. A signature may change over time and it is not as unique or difficult to forge as iris patterns or fingerprints, however, acceptance by the public makes it more suitable for certain lower-security authentication needs. Moreover, PDA and other portable digital devices are capable of providing support to get information about specific characteristics from signatures

Signature verification is split into two categories according to the available input data. Offline signature verification takes as input the image of a signature and is useful in automatic verification of signatures found on bank checks and documents. Online signature verification uses signatures that are captured by pressure-sensitive tablets that extract dynamic properties of a signature in addition to its shape [9]. The signature can be regarded as a series of rapid movements, which are dependent on the properties of human neuromuscular system. Mapping of the personal hand cadence and movement during the writing is highly difficult to forge. Dynamic features include the number and order of the strokes, the overall speed of the signature, the pen pressure at each point, cadence, etc., and make the signature more unique and more difficult to forge. As a result, online signature verification could be more reliable than offline signature verification in most cases. Various approaches have been proposed to solve the online signature verification problem: Multilayer perceptron neural networks [10], Hidden Markov Models [11], neurofuzzy systems [12], wavelet transform followed by discrete cosine transform for dimensionality reduction [13], fusion of methods, such as dynamic time warping and Hidden Markov Models [14], or dynamic time warping improved by incorporating the use of Fourier descriptors [15].

In this work, an on line signature verification with a feature extraction based on discrete wavelet transform is presented. The matching is performed by a modified dynamic time warping algorithm (MDTW), which operates on the approximation coefficients obtained through a wavelet sub-band coding algorithm. Two type of signature analysis are allowed by applying the matching at the stroke level (local analysis) or using the whole signature at once (global analysis).

2 System Description

System description can be summarized using the block diagram of Figure 1. In order to minimize the fluctuations of place, size, and rotation of the signature, some preprocessing operations were included in the first block. The rotation normalization was implemented using the Hotelling transform, which performs an alignment of the signature with its main axis through a matrix transformation formed by the eigenvectors of the covariance matrix using the X-Y signature position data. Once this tasks are performed, magnitude and phase information from the normalized data are used as input function to the recognition system. If a stroke-based analysis is performed, signature splitting is carried out in this block using the pen-up feature of the digitizing tablet. In the next block, feature extraction is performed using time-scale decomposition up to the specified level, based on the wavelet sub-band coding algorithm. A modified dynamic time warping (MDTW) algorithm is then applied on the wavelet coefficients. This operation consists of a matching/warping operation between vectors, which simultaneously finds a dissimilitude value between them. In the last stage, a matching decision block takes the dissimilitude value from the feature extraction blocks and compares it against a threshold value which is dynamically calculated based on statistical data information. If the dissimilitude value obtained from MDTW is less than threshold value, signature matching is decided to be positive. The template block temporarily stores the template signature, statistical data information, and dissimilitude values found in the training stage, for their posterior use during the verification process.

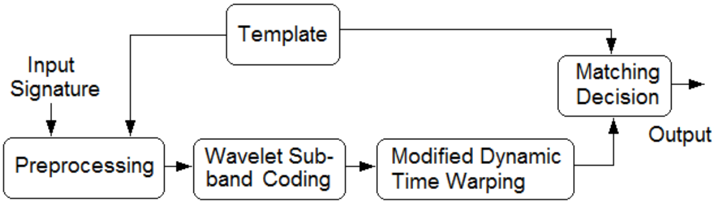


Fig. 1. System block diagram

3 Wavelet Sub-band Coding

The Discrete Wavelet Transform (DWT) is used to analyze the temporal and spectral properties of non-stationary signals. The DWT is defined by the following equation [16]:

$$W(j,k)=\sum_{j}\sum_{k} f(x)2^{-j/2}\psi(2^{-j}x-k) \tag{1}$$

The set of functions $\psi_{j,k}(n)$ is referred to as the family of wavelets derived from $\psi(n)$, which is a time function with finite energy and fast decay called the mother wavelet. The basis of the wavelet space corresponds then, to the orthonormal functions obtained from the mother wavelet after scale and translation operations. The definition indicates the projection of the input signal into the wavelet space through the inner product, then, any function $f(x)\ \epsilon\ \mathrm{L}^2(R)$ can be represented in the form:

$$f(x)=\sum_{j,k} d_j(k)\psi_{j,k}\ , \tag{2}$$

where $d_j(k)$ are the wavelet coefficients at level j. The coefficients at different levels can be obtained through the projection of the signal into the wavelets family as:

$$\langle f,\psi_{j,k}\rangle = \sum_l d_l \langle f,\phi_{j,k+l}\rangle \tag{3}$$

$$\langle f,\phi_{j,k}\rangle = \frac{1}{\sqrt{2}}\sum_l c_l \langle f,\phi_{j-1,2k+l}\rangle \tag{4}$$

The DWT analysis can be performed using a fast, pyramidal algorithm described in terms of multirate filter banks [17]. The DWT can be viewed as a filter bank with octave spacing between filters. Each sub-band contains half the samples of the neighboring higher frequency sub-band. In the pyramidal algorithm the signal is analyzed at different frequency bands with different resolution by decomposing the signal into a coarse approximation and detail information. The coarse approximation is then further decomposed using the same wavelet decomposition step. This is achieved by successive high-pass and low-pass filtering of the time signal, and a down-sampling by two as defined by the following equations:

$$c_j(k) = \sum_m h(m-2k)c_{j+1}(m) \tag{5}$$

$$d_j(k) = \sum_m g(m-2k)c_{j+1}(m) \tag{6}$$

Figure 2 shows a one-level filter bank. Signals $c_j(k)$, and $d_j(k)$ are known as approximation and detail coefficients, respectively. This process may be executed iteratively forming a wavelet decomposition tree up to any desired resolution level.

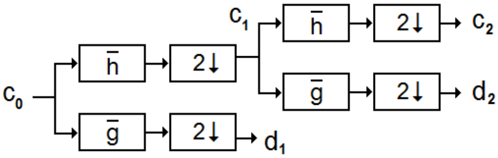


Fig. 2. Two-level discrete wavelet filter bank scheme

In this work, the approximation coefficients were used as input to the modified dynamic time warping algorithm, which is described in the following section. Different experiments were made using several wavelets and decomposition levels to obtain the best system performance, as described in the results section.

4 Modified Dynamic Time Warping Algorithm

Dynamic Time Warping was introduced by Kruskal and Liberman in the context of speech recognition [18], as a computational technique to make a matching between two time series, which may have different number of samples, providing a normalization and alignment of both sequences. DTW can distort the time axis by compressing it at some places and expanding it at others, as required. The main objective is the optimization of a function cost used to travel from one point to another, giving an optimal matching path based on some constraints. Minimization of the function cost is described as:

$$D(T_x, T_y) = \min_{\phi_x,\phi_y} \textstyle\sum_{k=1}^{T} d\left(\phi_x(k), \phi_y(k)\right) m(k) \tag{7}$$

Where $d\left(\phi_x(k), \phi_y(k)\right)$ is a dissimilitude value between both time sequences in the warping trajectory T, and $m(k)$ is a local weighting factor. Normalization of this measure is done by dividing this value between a global weight factor, which corresponds to the number of points in the warping trajectory, as expressed in equation 8. Details can be checked in reference [18].

$$D_v = \frac{D(T_x, T_y)}{T} \tag{8}$$

In this work we propose some modifications to the classical algorithm. The first one is the incorporation of a warping diagonal deviation used as complementary normalization factor. This value is given by equation 9, and expresses the ratio of the number of points in the warping trajectory and the sum of the number of points on the original data vectors to be matched. As the two data sequences are more dissimilar, the warping path separates from the diagonal and the coefficient tends to one.

$$D_N = \frac{T}{T_x + T_y} \qquad 9)$$

This dissimilitude measurement was also enhanced using information about direct matching points (DMP). DMPs are matched points unambiguously defined between data sets. Figure 3 shows an example of a matching segment between two data sequences, with the DMPs plotted in solid lines.

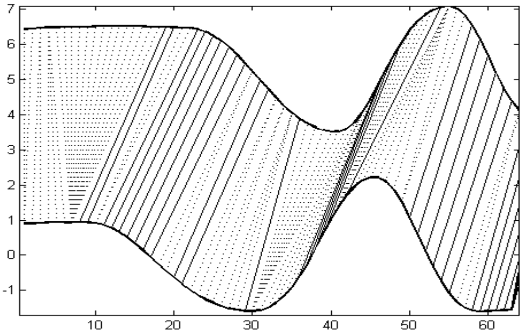


Fig. 3. Example of DMPs between two trajectories

Using these particular points, we propose an additional matching coefficient defined as:

$$C_{dmp} = 1 - \frac{\sum DMP}{T} \ , \tag{10}$$

where DMP stands for direct matching points, and T is the number of points in the warping trajectory. This coefficient is used as an additional weight factor in the dissimilitude measurement of DTW. The dissimilitude value V_{MDTW} obtained through the modified dynamic time warping is finally defined as:

$$V_{MDTW} = (D_v)(D_N)(C_{dmp}) = \frac{\left(D(T_x,T_y)\right)(1-\sum dmp)}{T(T_x+T_y)} \tag{11}$$

This value is used to make a decision on whether the signature corresponds to the template defined by the user in each case, by comparing it to a threshold value α_{th} defined in equation 12.

$$\alpha_{th} = \overline{Tm} + K\sigma_{Tm} \ , \tag{12}$$

where $\overline{Tm}$ is template mean value, σ_{Tm} is the template standard deviation value, and K is a weight factor used to control the trade-off between false rejections and false acceptances. K is defined by the administrator of the biometric system according to the need for security, which depends on some risk evaluation. During the system evaluation process this parameter assumes the range which allows the system to give both, FAR and FRR values from 0 to 100%.

$$matching = \begin{cases} positive \ if \ V_{MDTW} \leq \alpha_{th} \\ negative \ if \ V_{MDTW} > \alpha_{th} \end{cases} \tag{13}$$

Figure 4 show an example of warping trajectory (white line) on a dissimilitude matrix with values represented in gray levels.

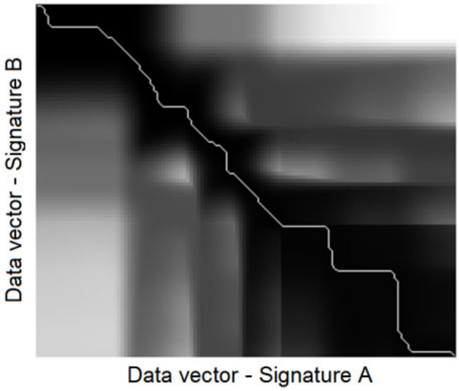
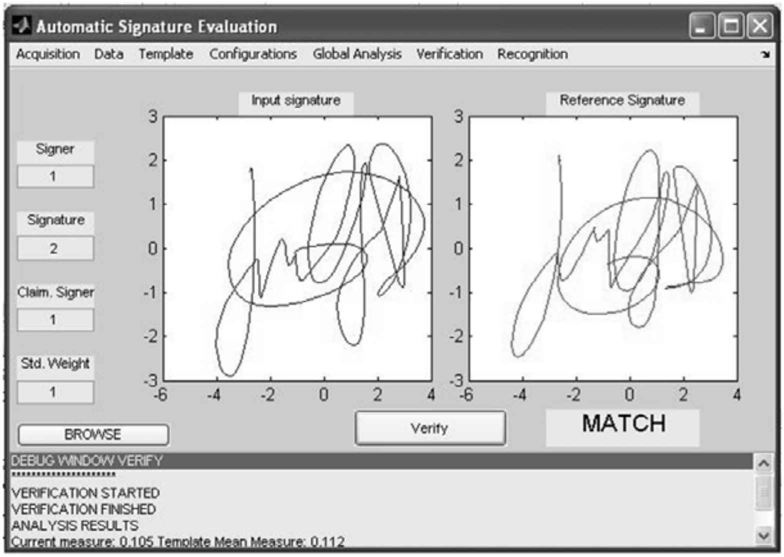


Fig. 4. Warping trajectory on a dissimilitude matrix obtained from two data vectors

5 Experimental Setup

On-line signature data acquisition was accomplished using a digitizing tablet Genius G-PEN 340 with a 3X4 inches active area and maximum data transmission rate of 100 points per second. For the described experiments a local signature database consisting of 1000 genuine signatures from 50 signers and 240 skilled forgeries from 12 subjects, was generated. The biometric system is accessed through a graphical user interface developed in MATLAB. Figure 5 shows the main screen of the application. Two modalities can be used: The first one consists of the signature analysis at stroke-level or using the whole signature. The second one is the application of the proposed modified dynamic time working algorithm directly to the raw data, or to the wavelet coefficients. From the combination of these modalities, four type of analysis can be performed. The graphical user interface allows the following tasks: Organize signature data files, set the number of signers and signatures to be used, generate templates, set the analysis type, perform a global analysis between enrolled signatures, perform signature verification and perform signature recognition. Figure 5 shows the GUI window in the verification mode:



Signer: 1

Signature: 2

Claim. Signer: 1

Std. Weight: 1

MATCH

```
**********************
VERIFICATION STARTED
VERIFICATION FINISHED
ANALYSIS RESULTS
Current measure: 0.105 Template Mean Measure: 0.112
```

Fig. 5. Main screen of the graphical user interface in verification mode

6 Results

Evaluation of the system performance was done through the ROC plot (receiver operation characteristic) for several cases. ROC is a plot of the false acceptance ratio (FAR) versus the false rejection ratio (FFR) [19]. The equal error rate (EER) is obtained from the point in which FAR and FFR assume the same value. Figure 6 shows the ROC curve obtained from an experiment which was done to check the effect of using the modified dynamic time warping algorithm vs. the classical DTW. The test was done using the raw data corresponding to the whole signature, i.e., without using wavelet decomposition. From this plot it can be seen that MDTW showed an EER of 12.28% approximately.

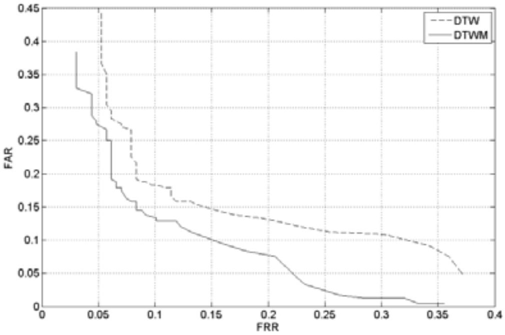


Fig. 6. Classical DTW vs Modified DTW ROC curves

A second experiment was done in order to characterize the system performance when the wavelet decomposition was incorporated. Figure 7 shows the ROC plot obtained using a different wavelet mother in each case, with a 3-level wavelet decomposition. This figure shows that the best result was obtained using the Coifflet-3 wavelet.

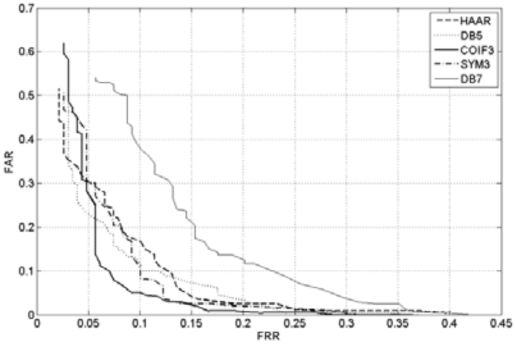


Fig. 7. ROC curves obtained using different wavelets

Figure 8 shows the improvement in performance obtained when the level-3 wavelet decomposition is incorporated into the biometric system. The plot shows an EER=7.46% for that case. In both cases, the proposed modified dynamic time warping algorithm is applied. The wavelet decomposition helped also to obtain a reduction in the dimensionality of data, which impacted on getting an improvement in the execution time.

Finally, figure 9 shows a comparison of ROC curves obtained when the system is tested using the stroke-based analysis. The four ROC plots correspond to the cases described as follows. AT1: MDTW applied to the whole signature. AT2: MDTW applied to the whole signature after 3-level wavelet decomposition. AT3: MDTW applied at stroke level. AT4: MDTW applied at stroke level and 3-level wavelet decomposition applied to each stroke.

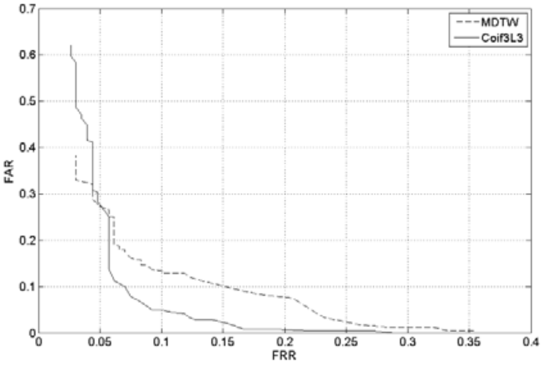


Fig. 8. ROC curves obtained using MDTW direct analysis vs MDTW-wavelet analysis

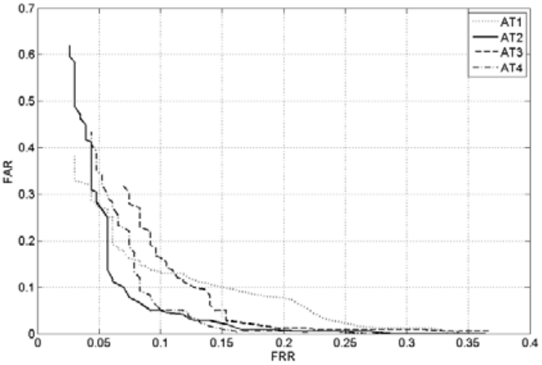


Fig. 9. ROC curves obtained using the MDTW algorithm in four different cases

7 Conclusions

This paper presented a signature-based biometric system using a modified dynamic time warping algorithm and wavelet decomposition. The described modification on the DTW when compared to the classical algorithm, provided an improvement in the system performance of 3.07% in average, as represented in the corresponding ROC curves. A further incorporation of a wavelet-based decomposition gave an additional improvement in the system performance, as well as a dimensionality reduction, which provided a considerable decreasing in execution time with an estimated factor of 23, when it was compared with the execution time without the wavelet decomposition. The best obtained results using both techniques showed in average a combined EER=7.46%. Further experiments using larger databases are currently in progress.

References

1. Jain, A.K.: Handbook of Biometrics. Springer, Heidelberg (2008)
2. Bowyer, K.W., Hollingsworth, K., Flynn, P.J.: Image understanding for iris biometrics: A survey. Computer Vision and Image Understanding 110(2), 281–307 (2008)
3. Chellappa, R., Sinha, P., Jonathon Phillips, P.: Face Recognition by Computers and Humans. Computer 43(2), 46–55 (2010)
4. Nanni, L., Lumini, A.: A multi-matcher for ear authentication. Pattern Recognition Letters 28, 2219–2226 (2007)
5. Ramírez-Cortes, J.M., Gómez-Gil, P., Sánchez-Pérez, G., Prieto-Castro, C.: Shape-based hand recognition approach using the pattern spectrum. Journal of Electronic Imaging 18(1) (2009)
6. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of fingerprint recognition, 2nd edn. Springer, Heidelberg (2009)
7. Kong, A., Zhang, D., Kamel, M.: A survey of palmprint recognition. Pattern Recognition 42(7), 1408–1418 (2009)
8. Kozik, R., Choras, M.: Combined Shape and Texture Information for Palmprint Biometrics. Journal of Information Assurance and Security 5, 058–063 (2010)
9. Impedovo, D., Pirlo, G.: Automatic signature verification: The state of the art. IEEE Trans. Syst. Man, Cybern. C 38(5) (September 2008)
10. McCabe, A., Trevathan, J., Read, W.: Neural Network-based Handwritten Signature Verification. Journal of Computers 3(8) (August 2008)
11. Fierrez, J., Ortega-Garcia, J., Ramos, D., Gonzalez-Rodriguez, J.: HMM-based on-line signature verification: Feature extraction and signature modeling. Pattern Recognition Letters 28(16), 2325–2334 (2007)
12. Nanni, L., Lumini, A.: A novel local on-line signature verification system. Pattern Recognition Letters 29(5), 559–568 (2008)
13. Khalid, M., Mokayed, H., Yusof, R., Ono, O.: Online Signature Verification with Neural Networks Classifier and Fuzzy Inference. In: 2009 Third Asia International Conference on Modelling & Simulation, pp. 236–241 (May 2009)
14. Nanni, L., Maiorana, E., Lumini, A., Campisi, P.: Combining local, regional and global matchers for a template protected on-line signature verification system. Expert Systems with Applications 37(5), 3676–3684 (2010)
15. Yanikoglu, B., Kholmatov, A.: Online Signature Verification Using Fourier Descriptors. EURASIP Journal on Advances in Signal Processing (2009) Article ID 260516
16. Priestley, M.B.: Wavelets and time-dependent spectral analysis. Journal of Time Series Analysis 17(1), 85–103 (2008)
17. Pinsky, M.A.: Introduction to Fourier Analysis and Wavelets. Graduate Studies in Mathematics, vol. 102. American Mathematical Society, Providence (2009)
18. Kruskal, J.B., Liberman, M.: The symmetric time-warping problem: from continuous to discrete. In: Sanko, D., Kruskal, J.B. (eds.) Time Warps, String Edits, and Macromolecules: The Theory and Practice of Sequence Comparisons, pp. 125–160. Addison-Wesley, Reading (1983)
19. Dunstone, T., Yager, N.: Biometric System and Data Analysis: Design. Evaluation and Data Mining. Springer, New York (2009)

New Dissimilarity Measures for Ultraviolet Spectra Identification

Andrés Eduardo Gutiérrez-Rodríguez[1], Miguel Angel Medina-Pérez[1],
José Fco. Martínez-Trinidad[2], Jesús Ariel Carrasco-Ochoa[2],
and Milton García-Borroto[1,2]

[1] Centro de Bioplantas. Carretera a Morón km 9, Ciego de Ávila, Cuba
[2] Instituto Nacional de Astrofísica, Óptica y Electrónica, Luis Enrique Erro No. 1,
Sta. María Tonanzintla, Puebla, México, C.P. 2840
{andres,migue,mil}@bioplantas.cu,
{fmartine,ariel}@ccc.inaoep.mx

Abstract. Ultraviolet Spectra (UVS) analysis is a frequent tool in tasks like diseases diagnosis, drugs detection and hyperspectral remote sensing. A key point in these applications is the UVS comparison function. Although there are several UVS comparisons functions, creating good dissimilarity functions is still a challenge because there are different substances with very similar spectra and the same substance may produce different spectra. In this paper, we introduce a new spectral dissimilarity measure for substances identification, based on the way experts visually match the spectra shapes. We also combine the new measure with the Spectral Correlation Measure. A set of experiments conducted with a database of real substances reveals superior results of the combined dissimilarity, with respect to state-of-the-art measures. We use Receiver Operating Characteristic curve analysis to show that our proposal get the best tradeoff between false positive rates and true positive rates.

Keywords: Ultraviolet Spectra, Ultraviolet Spectra Comparisons Functions, Substance Identification, Dissimilarity Measures.

1 Introduction

Ultraviolet Spectra (UVS) represent, for a given substance, the relation between ultraviolet light absorbance and light wavelength. Due to UVS are unique for each pure substance, they are frequently used for substance identification in different areas such as medicine, geology, criminalistics, and industrial applications [1–4].

Identifying substances by UVS is a challenge because there are different substances with very similar spectra shape (Fig. 1). Additionally, different concentrations of the same substance produce different spectra, dilated or contracted, according to Lambert-Beer Law [5] (Fig. 2).

In this paper, we focus on substance identification (mainly drugs, medicines, poisons, pesticides and other organic substances) by ranking its spectrum according to its dissimilarity values against the spectra in a database. These substances

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 220–229, 2010.

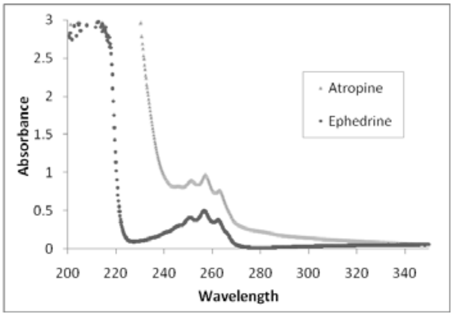


Fig. 1. Ultraviolet spectra of two different substances

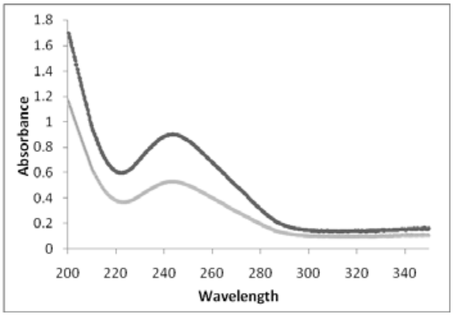


Fig. 2. Two spectra of Prednisone at different concentrations

are generally free of impurities or they have a predominant active chemical, which is involved in the identification process. In general, the quality of the dissimilarity function is the most important factor in the identification results.

In order to design a good UVS comparison function, we must take into account the application domain [1]; some of the most used properties are: agreements in the amplitudes of the signal, the shape of the spectrum, and a unique configuration of peaks that may change slightly. For substance identification, it is necessary a qualitative analysis of its spectrum. This leads to an empirical comparison of the unknown spectrum details with other known spectra. These details are maxima, minima and inflection points of the spectrum. Usually, experts visually match spectra based on their shape [6].

There are several UVS comparison functions [4, 7–12], but none of them effectively compare spectra shapes. The measures that compare spectrum absorbance values fail in comparing spectra of the same substance at different concentration, because these values can differ considerably. A normalization of the absorbance values introduces false positive matches when the spectrum has close absorbance values but differ in monotony or concavity. One attempt to overcome this problem is addressed in [1]. This measure compares spectra using the first derivatives, but ignores the changes in the curve concavity.

In this paper, we propose a new dissimilarity measure to compare ultraviolet spectra: Derivatives Sign Differences (DSD). DSD compares the spectral monotony and concavity, using the first and second derivatives of the 2D-shape of the spectrum at each wavelength. This way, the new measure can effectively match spectra of the same substance at different concentrations.

We compare DSD with several dissimilarity measures for substances identification, using a database containing 206 spectra of 103 substances (two spectra from each substance). This database was created by forensic experts in State Forensic Laboratory of Ciego de Avila. In order to evaluate the performance of the comparison functions we make use of Receiver Operating Characteristics (ROC) curves [13]. The experimental results show a good performance of the new measure, compared with eight UVS dissimilarities proposed in the literature. Moreover, we combine DSD with the Spectral Correlation Measure (SCM) [8], showing that this combination outperforms both single measures.

2 UVS Dissimilarities

There are several dissimilarity functions to compare mass spectra, infrared spectra, ultraviolet spectra, multi and hyper spectral images [11]. In this section, we briefly review some of the most cited spectral dissimilarities.

Perhaps, the most popular UVS measure is the Spectral Angle Mapper (SAM) [7]. SAM is primarily introduced for comparing hyperspectral image data; it compares two spectra by finding the angle between their absorbance tuples. Equation 1 shows SAM transformed to dissimilarity.

$$\mathrm{SAM}(s,t) = 1 - \left(\frac{\sum_{l=1}^{L} s_l t_l}{\sqrt{\sum_{l=1}^{L} s_l^2 \sum_{l=1}^{L} t_l^2}} + 1 \right) /2 \tag{1}$$

The tuple $s = (s_1, s_2, \ldots, s_l)$ represents a spectrum, where each s_l is the ultraviolet light absorbance for the corresponding light wavelength value w_l.

The main drawback of SAM is that the angle between tuples of two spectra of the same substance at different concentrations may be very different from zero. Van der Meer [8] proposes the Spectral Correlation Measure (SCM), which overcomes this limitation by standardizing the data, i.e. centralizing them using the mean of s and t [14].

$$\mathrm{SCM}\,(s,t) = 1 - \left(\frac{L \sum\limits_{l=1}^{L} s_l t_l - \sum\limits_{l=1}^{L} s_l \sum\limits_{l=1}^{L} t_l}{\sqrt{\left[L \sum\limits_{l=1}^{L} s_l^2 - \left(\sum\limits_{l=1}^{L} s_l \right)^2 \right] \left[L \sum\limits_{l=1}^{L} t_l^2 - \left(\sum\limits_{l=1}^{L} t_l \right)^2 \right]}} + 1 \right) /2 \tag{2}$$

The Spectral Information Divergence (SID) [9] measures the information divergence between the probability distributions generated by two spectra. To do so, SID models spectra as random variables by defining $p_k = s_k / \sum_{l=1}^{L} s_l,\ k = 1, \ldots, L$ so that $p = (p_1, \ldots, p_L)$. Then, it compares spectra taking into account the relative entropy between them:

$$\mathrm{SID}\,(s,t) = \sum_{l=1}^{L} p_l \log \frac{p_l}{q_l} + \sum_{l=1}^{L} q_l \log \frac{q_l}{p_l} \tag{3}$$

In a similar way as SAM, SID is introduced for comparing hyperspectral images.

SID-SAM is a combination of SID with SAM to enhance the spectral discriminatory probability [10]. The authors formulate this combination in two versions (Equation 4 and Equation 5).

$$\mathrm{SID} - \mathrm{SAM1}\,(s,t) = \mathrm{SID}\,(s,t) \sin\left(\arccos\left(1 - 2\mathrm{SAM}\,(s,t)\right)\right) \tag{4}$$

$$\mathrm{SID} - \mathrm{SAM2}\,(s,t) = \mathrm{SID}\,(s,t) \tan\left(\arccos\left(1 - 2\mathrm{SAM}\,(s,t)\right)\right) \tag{5}$$

Another fusion of spectral dissimilarities is the Spectral Similarity Scale (SSS) [11], which combines a modification of SCM with the Euclidean distance (Equation 6).

$$SSS\,(s,t) = \sqrt{1/L \sum_{l=1}^{L} (s_l - t_l)^2 + (1 - r)^2} \tag{6}$$

where:

$$r = \frac{\sum_{l=1}^{L} \left(s_l - 1/L \sum_{l=1}^{L} s_l\right) \cdot \left(t_l - 1/L \sum_{l=1}^{L} t_l\right)}{\sqrt{\sum_{l=1}^{L} \left(s_l - 1/L \sum_{l=1}^{L} s_l\right)^2 \cdot \sum_{l=1}^{L} \left(t_l - 1/L \sum_{l=1}^{L} t_l\right)^2}} \tag{7}$$

A significant drawback in the use of the Euclidean distance is that it is unbounded, because the range of values increases as the number of light wavelengths increases. Moreover, comparing spectra of the same substance at different concentrations (Fig. 2) would return high dissimilarity values. Robila and Gershman [12] propose the Normalized Euclidean Distance (NED) to overcome these limitations (Equation 8). The main difference with the Euclidean distance is that NED normalizes each spectrum dividing the absorbance values by the average absorbance.

$$\mathrm{NED}\,(s,t) = \sqrt{\sum_{l=1}^{L} \left(\frac{s_l}{1/L \sum_{l=1}^{L} s_l} - \frac{t_l}{1/L \sum_{l=1}^{L} t_l} \right)^2} \tag{8}$$

Fig. 3 shows the effects of normalizing spectra from Fig. 2. Notice that both spectra have now similar absorbance values. Nevertheless, there are examples where this type of normalization does not perform correctly. For example, Fig. 4 shows two spectra of the same substance after normalization with a clear difference between absorbance values for almost all wavelength values. In this case, NED erroneously returns a high dissimilarity value.

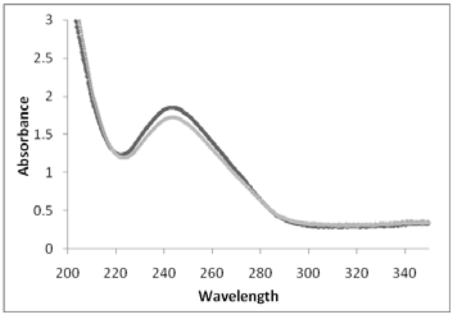


Fig. 3. Spectra from Fig. 2 normalized using the average of the absorbance values

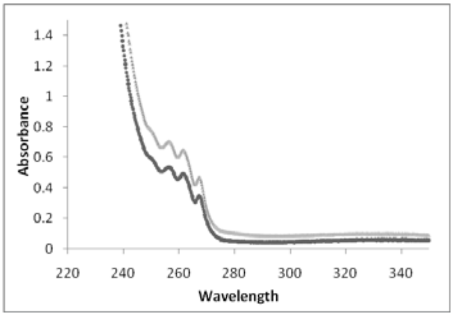


Fig. 4. Two spectra of Ampicillin after normalization

Paclík and Duin [1] propose to incorporate the difference between shapes of spectra into the dissimilarity measure. For that purpose, they use the spectra first derivative $(s'_1, \ldots, s'_L)$ for each light wavelength value. They compute the derivatives over the spectra normalized to unit area and improved by a Gaussian filter (Equation 9).

$$\mathrm{PD}\,(s,t) = \sum_{l=1}^{L} |s'_l - t'_l| \tag{9}$$

This measure has the same problem related to normalization discussed earlier. Moreover, PD ignores the information of spectral concavity to compare spectral shapes.

As we have shown in this section, there is no best spectral dissimilarity measure for all type of spectra. A good spectral dissimilarity for certain problems might be a bad measure in a different domain. In substances identification, experts visually compares UVS taking into account the similarity between spectral shapes. Based on this, we propose a measure that captures spectral shape with the aim of obtaining better results in spectra comparisons.

3 New Dissimilarity Measures for UVS Identification

The basic idea of Derivatives Sign Differences (DSD) is to count the points where either the monotony or the concavity of the spectra differs. Therefore, the lower value returned, the lesser spectral difference.

3.1 Derivatives Sign Differences Measure

Given two spectra $s = (s_1, \ldots, s_L)$ and $t = (t_1, \ldots, t_L)$, DSD computes the dissimilarity between s and t as follows:

```
1. Compute and smooth the first derivative value tuples from spectra $s$ and $t$:
   $s' = (s'_1, \ldots, s'_L)$, $t' = (t'_1, \ldots, t'_L)$
2. Compute and smooth the second derivative value tuples from $s'$ and $t'$:
   $s'' = (s''_1, \ldots, s''_L)$, $t'' = (t''_1, \ldots, t''_L)$
3. Set $count = 0$
4. For each $l = 1, \ldots, L$
     – If $(sign\,(s'_l) \neq sign\,(t'_l) \vee sign\,(s''_l) \neq sign\,(t''_l))$ then $count = count + 1$
5. Return $count/L$
```

The normalization at step 5 returns a value in the [0, 1] interval. A median filter smoothes the derivatives tuples, using a window width of five values. We choose the window width experimentally, but small variations of this value attain similar results.

Notice in Fig. 5 (a) the decreasing monotony of a spectrum in light wavelength interval [280, 320]. Fig. 5 (b) shows a zoom of the interval [305, 310] where dark dots represent the spectrum points, and line segments represent the slope directions at each point. Opposite to what we would expect, a zoom of this interval shows heterogeneous slope directions. That is why we apply a median filter over the derivative values tuples (steps 1 and 2), achieving homogeneous

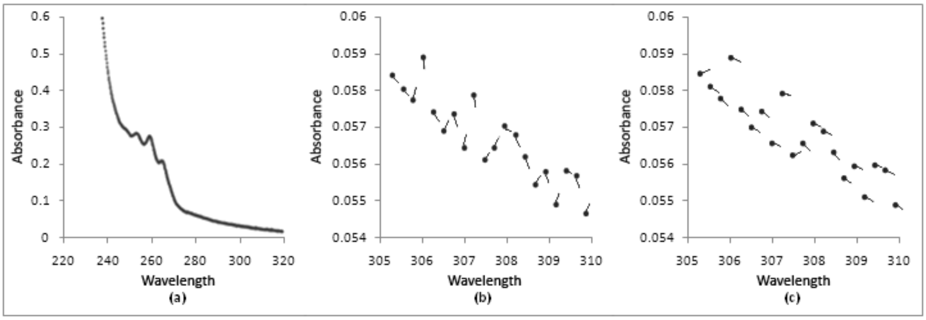


Fig. 5. Result of applying a median filter over the first derivatives values tuple. (a) Original spectrum. (b) Slope directions in a zoomed portion of the spectrum. (c) Homogeneous directions after applying a median filter.

slope directions, as shown in Fig. 5 (c). As we can see in the experimental section, this procedure improves the accuracy of the results.

Due to spectra derivatives are pre calculated and stored before searching for a query spectrum, the time complexity of DSD is $O(n)$, being n the total wavelength values. DSD effectively matches spectra from the same substance at different concentration. The proposed measure does not compare absorbance values but the signs of first and second derivatives tuples. Thus, DSD correctly matches different spectra from the same substance because these spectra do not differ in monotony and concavity.

In the experimentation, the proposed measure DSD outperforms all other dissimilarities except SCM. As SCM does not clearly outperform DSD, we decided to combine them to improve their results.

3.2 Fusion of DSD and SCM Dissimilarities

DSD and SCM compare spectra in different manners. DSD takes into account monotony and concavity of spectra, while SCM determines the correlations between the spectra tuples. We decided to combine them in order to benefit from their advantages and reducing their limitations. To do this, we use a combination scheme that returns small spectra difference if one of these measures indicates it, regardless of the result of the other measure.

Multiplying the results of two dissimilarity measures, both defined on the [0, 1] interval, if one of them returns a value close to zero, it takes precedence over the other; the final value will also be close to zero, indicating small difference. That is why we use the product rule [15] to combine DSD with SCM. This rule is one of the most used schemas for comparison functions combination.

$$\mathrm{DSD} - \mathrm{SCM}\,(s,t) = \mathrm{DSD}\,(s,t) \cdot \mathrm{SCM}\,(s,t) \tag{10}$$

Using the product rule we attempt to favor the best result of both dissimilarities achieving lower values when both dissimilarities agree in a good result.

4 Experimental Results

We tested the proposed dissimilarity measure on a database containing 206 ultraviolet spectra from 103 substances. For each substance, we extracted two spectra at different concentrations (Fig. 2) with the Spectrometer Cintra 101 [6]. The wavelength values ranges from 200 nm to 350 nm. The measurement interval was 0.5 nm to emphasize spectra details, and the average speed was 500 nm/min.

As our goal is to identify a query spectrum with its corresponding substance, for each query spectrum we sort all the spectra in the database in increasing order according to their dissimilarity with the query. A good measure always returns the correct substance in the first positions. That is why, in order to evaluate the performance of our dissimilarity, we build a ROC curve using a leave one out [16] sampling.

A ROC curve [13] is a useful tool in the evaluation of comparison functions for objects identification, this curve measures the tradeoff between correct and false identification rates. Moreover, it has become a standard in areas such as fingerprint identification [17], which is analogous to substance identification.

In terms of ROC curve, a dissimilarity measure is better than another one if it has higher true positive rates for most false positive rates.

Our first experiment shows how DSD with smoothed first and second derivatives outperforms the same dissimilarity without smoothing (Fig. 6).

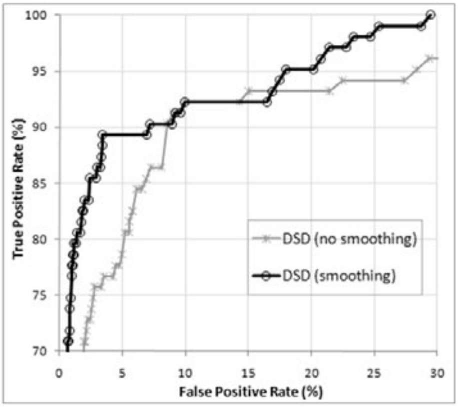


Fig. 6. The ROC curves of DSD smoothing derivatives versus DSD without smoothing

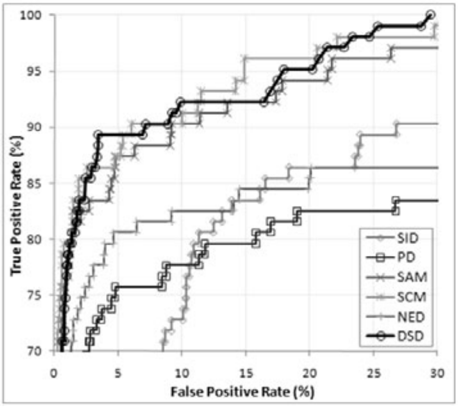


Fig. 7. ROC curves of DSD and all single dissimilarities

We compare DSD against the single dissimilarities reviewed in section 2, see Fig. 7. Notice that DSD clearly outperforms every other dissimilarity except SCM. However, SCM does not clearly outperform DSD.

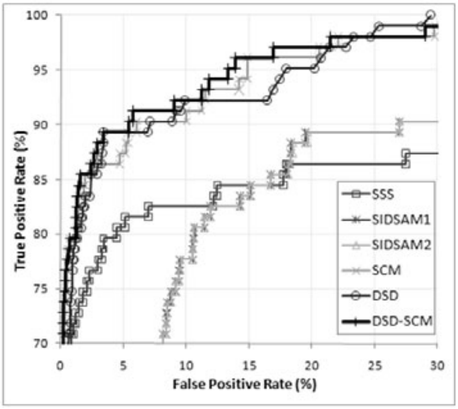


Fig. 8. ROC curves of combining DSD and SCM and three compound UVS dissimilarities

Our final experiment shows how the combination of DSD with SCM is superior to all individual measures and other composite measures proposed in the literature, as shown in Fig. 8.

The combination of DSD with SCM effectively takes the advantage of both measures and reduces its limitations. It returns low dissimilarity values if two spectra are dissimilar according to DSD or SCM. In addition, notice that combinations reviewed do not outperform its component measures.

5 Conclusions

In this paper, we introduce a new dissimilarity measure to compare UVS spectra. We create DSD based in the way experts visually compare spectra shape for substance identification. DSD allows better discrimination between UVS from different substances than most of the measures reviewed do. The combination of DSD with SCM clearly outperforms all single and combined measures analyzed in this paper. As future work, we plan to investigate our approach in similar problems using other wavelengths like visible and near infrared.

References

1. Paclík, P., Duin, R.P.W.: Classifying spectral data using relational representation. In: International Workshop on Spectral Imaging, pp. 31–34 (2003)
2. Demir, B., Ertürk, S.: Improved classification and segmentation of hyperspectral images using spectral warping. Int. J. Remote Sens. 29, 3657–3663 (2008)
3. Paclík, P., Leitner, R., Duin, R.P.W.: A study on design of object sorting algorithms in the industrial application using hyperspectral imaging. J. Real-Time Image Proc. 1, 101–108 (2006)

4. Van der Meer, F.: The effectiveness of spectral similarity measures for the analysis of hyperspectral imagery. Int. J. Appl. Earth Obs. Geoinf. 8, 3–17 (2006)
5. Ingle, J.D.J., Crouch, S.R.: Spectrochemical Analysis. Prentice Hall, New Jersey (1988)
6. GBC UV-Visible Cintra 101/202/303/404 Spectrometer Operation Manual. GBC Scientific Equipment Pty Ltd., Australia, GBC part number: 01-0831-01 (2005)
7. Kruse, F.A., Lefkoff, A.B., Boardman, A.B., Heidebrecht, K.B., Shapiro, A.T., Barloon, P.J., Goetz, A.F.H.: The Spectral Image Processing System (SIPS) - interactive visualization and analysis of imaging spectrometer data. Remote Sens. Environ. 44, 145–163 (1993)
8. Van der Meer, F., Bakker, W.: Cross correlogram spectral matching (CCSM): application to surface mineralogical mapping using AVIRIS data from Cuprite, Nevada. Remote Sensing Environ. 61(3), 371–382 (1997)
9. Chang, C.-I.: An information-theoretic approach to spectral variability, similarity, and discrimination for hyperspectral image analysis. IEEE Trans. Inf. Theory 46, 1927–1932 (2000)
10. Yingzi, D., Chein, I.C., Hsuan, R., Chein-Chi, C., James, O.J., Francis, M.D.A.: New hyperspectral discrimination measure for spectral characterization. Opt. Eng. 43, 1777–1786 (2004)
11. Homayouni, S., Michel, R.: Hyperspectral image analysis for material mapping using spectral matching. In: Proceedings of the XX ISPRS Congress, Proceedings volume IAPRS, Istanbul, July 12-23, vol. XXXV (2004)
12. Robila, S.A., Gershman, A.: Spectral matching accuracy in processing hyperspectral data. In: International Symposium on Signals, Circuits and Systems (ISSCS 2005), pp. 163–166 (2005)
13. Fawcett, T.: An introduction to ROC analysis. Pattern Recognit. Lett. 27, 861–874 (2006)
14. de Carvalho, O.A., Meneses, P.R.: Spectral Correlation Mapper (SCM): An Improvement on the Spectral Angle Mapper (SAM). In: NASA JPL AVIRIS Wkshp. (2000)
15. Kuncheva, L.I.: Combining pattern classifiers: methods and algorithms. Wiley Interscience, Hoboken (2004)
16. Devroye, L., Gyorfi, L., Lugosi, G.: A Probabilistic Theory of Pattern Recognition. Springer, New York (1996)
17. Maltoni, D., Maio, D., Jain, A.K., Prabhakar, S.: Handbook of Fingerprint Recognition, 2nd edn. Springer, London (2009)

Third Degree Volterra Kernel for Newborn Cry Estimation

Gibran Etcheverry[1], Efraín López-Damian[2], and Carlos A. Reyes-García[3]

[1] DIFUS-USON, Encinas y Rosales S/N, 83000, Hermosillo, Mexico
gibran.etcheve@difus.uson.mx

[2] FIME-CIIDIT-UANL, Mechatronics Department, PIIT, 66600, Apodaca, Nuevo León, Mexico
efrain.lopezdm@uanl.edu.mx

[3] INAOE, Department of Computer Science, Enrique Erro 1, 72840, Tonantzintla, Mexico
kargaxxi@ccc.inaoep.mx

Abstract. Newborn cry analysis is a difficult task due to its nonstationary nature, combined to the presence of nonlinear behavior as well. Therefore, an adaptive hereditary optimization algorithm is implemented in order to avoid the use of *windowing* nor *overlapping* to capture the transient signal behavior. Identification of the linear part of this particular time series is carried out by employing an Autorregresive Moving Average (ARMA) structure; then, the resultant estimation error is approched by a Nonlinear Autorregresive Moving Average (NARMA) model, which realizes a Volterra cubic kernel by means of a bilinear homogeneous structure in order to capture burst behavior. Normal, deaf, asfixia, pain, and uncommon newborn cries are inspected for differentation.

1 Introduction

The newborn cry represents the heart-breathing coordination activity with the muscular larynx. The cry is an acoustics event that contains information about the central nerves system functioning as well. It is one of the most important natural ways of communication that a newborn has to interact with his/her environment; through crying, the newborn babies express their needs and basic feelings like hungry or pain. Crying is the only non visual option that neonatologists or pediatricians have for disease or malaise recognition.

The Golub [1] crying model is divided into four parts. The first part is the glottal system that is in charge to develop the pressure under the glottis necessary to drive the vocal folds. The second part is the source of sound located at the larynx. This sound source can be described mathematically, in frequency domain, as a periodic source or like noisy source. All these sources can work in an isolated or simultaneous way. Both acoustics sources are originated in the vocal folds. The periodic source is produced by the vocal folds vibration. The noise is seen as a turbulence and is the result of the forced air passing through a small opening

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 230–239, 2010.

between the vocal folds from the lungs. The third part is composed by the vocal and nasal tract in the larynx, these elements act like an acoustics filter where its transfer function characteristics change due to their form and length. The four part is the radiation that describes the filtering of sound between the infant mouth and a microphone located at a certain distance to obtain this acoustic signal [2].

Nonstationary signals are those which present a time variable change of their statistical properties and frequency content [3]. For instance, we can find them in the areas of geology on seismic events [4] and of biomedicine over Heart Rate Variability (HVR) studies [5], electroencephalogram (EEG) characterization [6], newborn cry [2] and children speech analysis [7], among others.

In the case of crying, the first studies started back in 1838 with Gardiner, using music notes and in 1855 Darwing studied the local anatomy and physiology of speech development and its relation with the emotion expression. Babies cannot satisfy their necessities by themselves, they completely depend on the attentions of adults. The cry is the natural way of communication to express their different emotional and physiological states.

The widespread presence of these kind of signals has required to study them deeply in order to understand their behavior; for example, some of them present an intermittent sequence of frequency content and chaotic behavior in time.

Newborn cry classification is based on linear features extraction in the area of Artificial Intelligence (AI) [2]. In the case of speech, Linear Predictive Coefficients (LPC) analysis has been carried out followed by a quadratic degree Volterra filtering of the obtained residual for speech coding [8][9], analysis and synthesis [10].

In this work, the hereditary computation of the signal correlation function, gives us the possibility to determine the time variyng parameters between windows of time automatically without windowing nor overlapping. This adaptive task is carried out by an autorregressive moving average (ARMA) model[11] in order to approach the linear part of five kinds of newborn cries: normal, deaf, asfixia, pain, and uncommon. Thus, the resulting estimation error is projected onto a cubic degree Volterra kernel by means of a particular bilinear structure [12]. A good estimation performance of newborn cry signals is obtained due to burst modeling by nonlinear approximation.

2 Adaptive Hereditary Computation

Adaptive hereditary computation has its origin in the area of systems identification by employing an ARMA model, in order to fit a stochastic realization to some measured output $y_\tau, \tau = 1, ..., t$ of an unknown system. The one-step ahead predictor therefore implemented, estimates the model coefficients without using a non linear optimization technique, as it is the case for gradient based or Gauss-Newton techniques [11]. Instead of it, recomputation of $\tau = 1, \ldots, T$ past samples is developed by using the parameters obtained at time t ; hence, the correlation function is time variyng and up to date, avoiding in this way calculation errors due to *incomplete* correlation terms obtained with standard estimation methods.

2.1 ARMA Form

The one-step ahead predictor in ARMA form is written as follows:

$$\hat{y}_t = \sum_{i=1}^{n} a_i \hat{y}_{t-i} + \sum_{j=1}^{n} b_i \tilde{y}_{t-j} \tag{1}$$

where $\tilde{y}_t = y_t - \hat{y}_t$ and a_i,b_i are the autorregresive and input coefficients, respectively.

2.2 Hereditary Computation

As it was stated in the begining of this section, the transient optimization approach can be developed to the price of hereditary computation of the model coefficients, presenting a linear t-growing memory of size nt, with n as the system dimension or delay. Hence, the ARMA form having these characteristics is written as:

$$\hat{y}^t_\tau = \sum_{i=1}^{n} a^t_i \hat{y}^{t-i}_{\tau-i} + b^t_i \tilde{y}^{t-i}_{\tau-i}, \quad \forall \tau = 1, \ldots, t. \tag{2}$$

In order to obtain the model parameters, it is necessary to employ an evaluation criterium of how well (2) performs; this criterium consists on minimizing the mean square error (MSE) between the time series y_t and the predictor $\hat{y}^t_\tau$:

$$J^t_T = E^t_T[(y_\tau - \hat{y}^t_\tau)^2] = \frac{1}{T} \sum_{\tau=t-T+1}^{t} (y_\tau - \hat{y}^t_\tau)^2 \quad \tau = 1, \ldots, T. \tag{3}$$

where T is the time horizon of *hereditary computation* or *re-computation* of the estimated samples. This means that the estimator *adapts* its horizon every T samples to get the time-variyng parameters that characterize the signal of interest.

Derivating (3) with respect to the model parameters and separating terms to each side of the equality, leads us to the *normal equations*, see [13], which contain the time-varying correlation and intercorrelation terms:

$$\begin{bmatrix} \sum_{\tau=t-T+1}^{t} \hat{y}^{t-i}_{\tau-i} \hat{y}^{t-i}_{\tau-i} & \sum_{\tau=t-T+1}^{t} \hat{y}^{t-i}_{\tau-i} \tilde{y}^{t-i}_{\tau-i} \\ \sum_{\tau=t-T+1}^{t} \tilde{y}^{t-i}_{\tau-i} \hat{y}^{t-i}_{\tau-i} & \sum_{\tau=t-T+1}^{t} \tilde{y}^{t-i}_{\tau-i} \tilde{y}^{t-i}_{\tau-i} \end{bmatrix} \begin{bmatrix} a^t_i \\ b^t_i \end{bmatrix} = \begin{bmatrix} \sum_{\tau=t-T+1}^{t} y_\tau \hat{y}^{t-i}_{\tau-i} \\ \sum_{\tau=t-T+1}^{t} y_\tau \tilde{y}^{t-i}_{\tau-i} \end{bmatrix} i = 1 \ldots n. \tag{4}$$

Once the aforementioned model parameters are calculated, it is possible to build the system transfer function and obtain the estimated output time series or the estimated system impulse response.

3 Nonstationary Behavior

A time series is considered nonstationary when there is a considerable part of the power spectra in the low frequencies, since the corresponding mode has very few oscillations during the observation time. Besides this, if the signal alternates between periodic and irregular behavior, it presents a nonlinear phenomena called *intermittency*. These caotic phases can be long or in the form of bursts [14]. The newborn cry presents both characteristics, a periodic behavior with and important spectral component in the low frequencies (pitch of 400-600Hz) combined with short intermittencies or bursts, see Fig. 1.

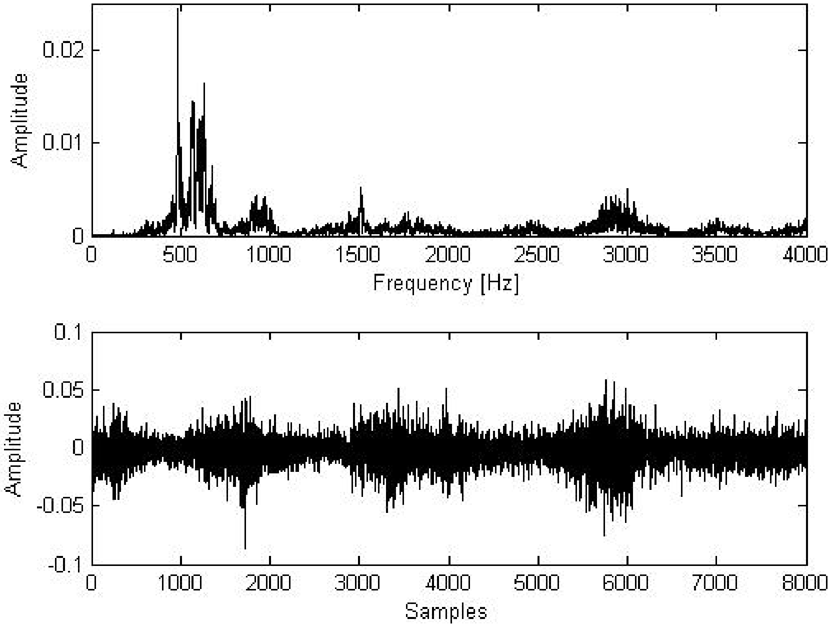


Fig. 1. Normal newborn cry spectra (up) and linear estimation error presenting *bursts* (bottom)

Figure 2 shows a different spectral pattern distribution as well as very frequent jumps on the estimation error obtained by using (2).

3.1 Cubic Volterra Estimation

The error $\tilde{y}_t^1$ resulting of substracting the ARMA estimator (2) from the measured time series y_t, can be approximated by a cubic Volterra kernel K_3 of the form:

$$\hat{y}_t = y_t - \hat{y}_\tau^t = \sum_{\tau_1=1}^{t} \sum_{\tau_2=1}^{\tau_1} \sum_{\tau_3=1}^{\tau_2} K_3(t, \tau_1, \tau_2) \tilde{y}_{\tau_1} \tilde{y}_{\tau_2} \tilde{y}_{\tau_3} \qquad (5)$$

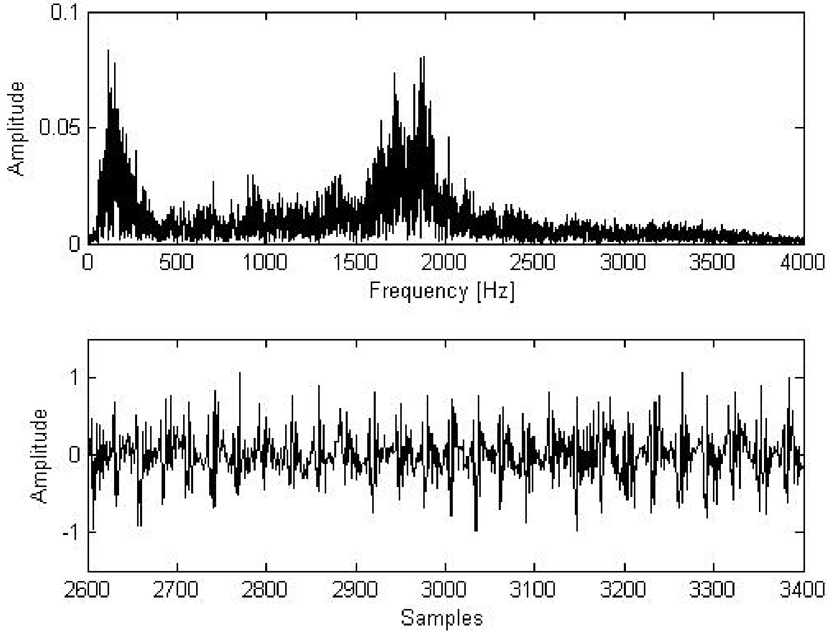


Fig. 2. Uncommon newborn cry spectra (up) and linear estimation error presenting *glitches* (bottom)

3.2 Bilinear Structure

It is a well known fact that a disadvantage of using a Volterra series for modeling nonlinear events is the curse of dimensionality, which makes the number of coefficients employed to grow fast with prediction horizon [8]. In the view of overcoming this feature, a bilinear homogeneous structure can generate *separated* Volterra series kernels recursively, yielding few parameters and allowing to choose the nonlinear degree of estimation [15].

Proposition: The homogeneous bilinear system that realizes the cubic Volterra kernel $K_3(t, \tau_1, \tau_2)$ is the following:

$$\begin{aligned} x_t^1 &= A_t^1 x_{t-1}^1 + B_t \tilde{y}_t \\ x_t^2 &= A_t^2 x_{t-1}^2 + D_t^2 x_t^1 \tilde{y}_t \\ x_t^3 &= A_t^3 x_{t-1}^3 + D_t^3 x_t^2 \tilde{y}_t \\ \hat{y}_t^3 &= C_t x_t^3 \end{aligned} \tag{6}$$

with

$$B_t^1 = \left[b_{n_1,t}^1 \cdots b_{1,t}^1 \right]^T$$

$$D_t^k = \begin{bmatrix} d^k_{n_k,n_{k-1},t} & \ldots\ldots & & d^k_{n_k,1,t} \\ \vdots & \vdots & \vdots & \vdots \\ d^k_{2,n_{k-1},t} & \ldots\ldots & & d^k_{2,1,t} \\ 0 & \ldots & 0 & 1 \end{bmatrix}; \forall k = 2 \ldots d$$

$$A_t^k = \begin{bmatrix} 0 & \ldots\ldots & 0 & a^k_{n_k,t} \\ 1 & \ddots & \vdots & \vdots \\ 0 & \ddots & \ddots \vdots & \vdots \\ \vdots & \ddots & \ddots 0 & \vdots \\ 0 & \ldots & 0 \; 1 & a^k_{1,t} \end{bmatrix}; \forall k = 1 \ldots d$$

$$C = \begin{bmatrix} 0 \ldots 0 \; 1 \end{bmatrix}$$

where $x_t^k \in \Re^{n_3}$ and $A_t^k \in \Re^{n_3 \times n_3}, \forall k = 1 \ldots 3, D_t^k \in \Re^{n_3 \times n_2}, \forall k = 2, 3, C_t \in \Re^{n_3}$ and $B_t \in \Re^{n_1}$.

Proof: Every state x_t^k, for $k = 1 \ldots 3$ can be considered as a linear system output with input $D_t^k x_t^{k-1} \tilde{y}_t$. Thus, we have:

$$\begin{array}{c} x_t^1 = \sum_{\tau=1}^{t} \phi^1(t,\tau) B_\tau \tilde{y}_\tau \\ x_t^2 = \sum_{\tau=1}^{t} \phi^2(t,\tau) D_\tau^2 x_\tau^1 \tilde{y}_\tau \\ x_t^3 = \sum_{\tau=1}^{t} \phi^3(t,\tau) D_\tau^3 x_\tau^2 \tilde{y}_\tau \\ \hat{y}_t^3 = C_t x_t^3 \end{array} \tag{7}$$

where $\phi^k(t,\tau) = A_t^k \phi^k(t-1,\tau)$ is the transition matrix and $K_3(t,\tau_1,\tau_2) = C_t \phi^3(t,1)(\phi^3(\tau_1,1)^{-1}) D_{\tau_1}^3 \phi^2(\tau_1,1)(\phi(\tau_2,1))^{-1}) D_{\tau_2}^2 \phi^1(\tau_2,1)(\phi(\tau_3,1))^{-1}) B_{\tau_1}$.

In this work this kind of structure is used, but in order to be able to adapt the hereditary computation to it, the matrix canonical form in (6) has been developped to end up with a NARMA expression of the form:

$$\begin{array}{c} y_t^1 = \sum_{i=1}^{n_1} a_{i,t}^1 y_{t-i}^1 + \sum_{i=1}^{n_1} b_{i,t}^1 \tilde{y}_{t-i+1} \\ y_t^2 = \sum_{i=1}^{n_2} a_{i,t}^2 y_{t-i}^2 + y_t^1 \tilde{y}_t + \sum_{i=2}^{n_2} \sum_{j=1}^{n_1} d_{i,j,t}^2 y_{t-i-j+2}^1 \tilde{y}_{t-i+1} \\ y_t^3 = \sum_{i=1}^{n_3} a_{i,t}^3 y_{t-i}^3 + y_t^2 \tilde{y}_t + \sum_{i=2}^{n_3} \sum_{j=1}^{n_2} d_{i,j,t}^3 y_{t-i-j+2}^2 \tilde{y}_{t-i+1} \end{array} \tag{8}$$

The expresion for any nonlinear degree of (6) and derivation of (8) can be found in [12].

3.3 Algorithm

Resuming the last two sections, the algorithm used in this work is the following:

1. Estimate the linear part of the signal by using (2), taking as optimization criterion (3).

2. Obtain and approach the resulting error on (5) by employing the estimator (8) at time t:

$$\begin{array}{c} \hat{y}_{\tau}^{1,t} = \sum_{i=1}^{n_1} a_{i,t}^{1} \hat{y}_{\tau-i}^{1,t-i} + \sum_{i=1}^{n_1} b_{i,t}^{1} \tilde{y}_{t-i+1} \\ \cdots \\ \hat{y}_{\tau}^{3,t} = \sum_{i=1}^{n_3} a_{i,t}^{3} \hat{y}_{\tau-i}^{3,t-i} + \hat{y}_{\tau}^{2,t} \tilde{y}_t + + \sum_{i=2}^{n_3} \sum_{j=1}^{n_2} d_{i,j,t}^{3} \hat{y}_{\tau-i-j+2}^{2,t-i-j+2} \tilde{y}_{t-i+1} \end{array} \tag{9}$$

   where $\hat{y}_{\tau}^{1,t-1}$ and $\hat{y}_{\tau}^{3,t-1}, \forall \tau = 1, \ldots, t-1$ were obtained by minimizing the criterion:

$$J_T^{t-1} = E_T^{t-1}[(\tilde{y}_t - \hat{y}_*^{3,t-1})^2] = \frac{1}{T+t-1} \sum_{\tau=T}^{t-1} (\tilde{y}_t - \hat{y}_*^{3,t-1})^2 \tag{10}$$

   and T is the time horizon of *hereditary computation* as in (3).
3. Obtain the kernel coefficients.

4 Results

This work has analyzed 10 recordings of 5 different kinds of newborn cries each: normal, deaf, asfixia, pain, and uncommon. These are one second recordings sampled at $F_s = 8000Hz$ that have been treated in segments of time horizon $T = 20ms$. It is important to mention that there is no overlapping between segments nor windowing to carry out the analysis.
The order or dimension used for the linear estimator was of $n = 10$, considering the experience on speech and newborn cry analysis [2].

After doing some testing, it was realized that the cubic kernel yields a good estimation from the beforehand mentioned error by using a dimension $n = 1$, leaving us with the expression (9) reduced to:

$$\hat{y}_{\tau}^{3,t} = a_{1,t}^{3} \hat{y}_{\tau-1}^{3,t-1} + a_{1,t}^{2} \hat{y}_{\tau-1}^{2,t-1} \tilde{y} + a_{1,t}^{1} \hat{y}_{\tau-1}^{1,t-1} \tilde{y}^2 + b_{1,t}^{1} \tilde{y}^3 \tag{11}$$

The algorithm quality is assessed after estimation by the criterium:

$$R = \left(1 - \frac{\sum_{t=1}^{N} |y_t - \hat{y}_t|^2}{\sum_{t=1}^{N} |y_t|^2} \right) \tag{12}$$

with N the number of samples.

Table 1 shows that in a few cases using linear estimation suffices to describe a newborn cry behavior, whereas in most of them it is not enough to describe this kind of signal. This is an even more important characteristic in cases where the linear signal estimation performance is below 80 *percent*, as it is the case mainly for deaf newborn cry analysis. Pain crying was obtained by light stimulation of the infants and uncommon crying analysis has been included to show that linear signal estimation is not enough when the crying behavior presents abrupt changes, see Fig 2.

Table 1. Linear ARMA and Linear+Cubic Volterra Series Fit

Normal		Deaf		Asfixia		Pain		Uncommon	
Lin	Lin+Cub	Lin	Lin+Cub	Lin	Lin+Cub	Lin	Lin+Cub	Lin	Lin+Cub
82.75	99.23	67.91	97.66	97.30	-	90.18	99.51	85.70	97.89
95.62	99.64	70.79	96.01	55.00	89.62	84.84	96.85	87.48	99.15
90.10	98.95	72.47	95.76	89.33	98.86	86.18	99.19	88.44	99.08
90.83	99	75.08	96.31	96.83	-	88.37	98.73	97.20	-
97.29	-	73.51	97.81	78.07	99.76	77.53	96.10	91.43	99.33
82.65	98.98	65.93	95.45	90.08	99.74	77.77	96.74	97.26	-
91.54	99.59	59.05	97.54	90.08	98.94	78.02	97.31	97.60	-
92.70	99.56	82.28	98.97	92.07	96.46	77.65	97.27	83.35	97.45
69.14	98.77	77.12	97.88	98.15	-	85.03	97.85	87.01	98.83
87.75	99.28	77.12	96.98	99.06	-	79.82	97.77	71.96	95.72

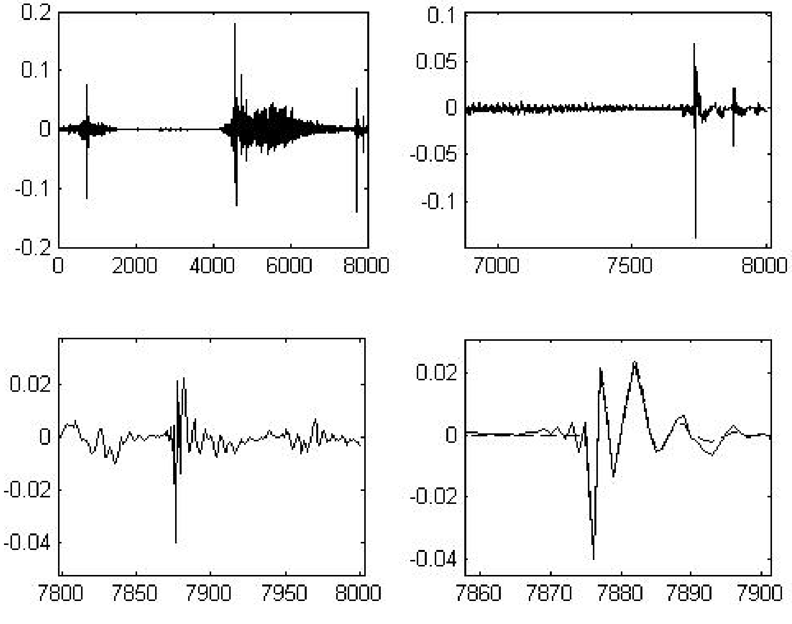


Fig. 3. Asfixia newborn cry burst close up images and estimation (*bottom right, broken line*)

5 Conclusions

This work presents as main contributions:

- The study of intermittencies present within newborn cries by using cubic Volterra analysis, which to the authors knowledge, it hasn't been implemented until now.

- The non use of overlapping nor windowing to compensate the nonstationary newborn cry behavior; the signal was just directly analized on segments of 20ms due to hereditary computation.
- The possibility of obtaining nonlinear characteristics from newborn cries, which will be useful to the community of pattern recognition and classification.

Once the Volterra kernels are *realized*, the estimator is able to *recontruct* the time series behavior based on the obtained error. There are some recent works on estimation of chaotic time series, see [16] and [17], but they employ different Volterra kernel realizations. An interesting comparison of performance can be developped with optimization methods like the one shown in [18]; even though, these models are neither autorregresive nor transient optimized, see [19] and [11] respectively.

Given the close relation between the acoustics aspects of the crying and the anatomical and physiological characteristics of the newborn, the mathematical theory of signal processing plays an important part for the fundamental research about medical applications like diagnosis based on crying. Estimation based on hereditary and Volterra techniques allows to obtain a more accurate model of the crying signal, and therefore it can make certainly these kind of applications easier.

References

1. Golub, H., Corwin, M.: Infant cry: a clue to diagnosis. Pediatrics 69, 197–201 (1982)
2. Reyes-García, C., Cano-Ortiz, S.: Fundamentos Téoricos y Prácticos del Análisis de Llanto Infantil. Inaoe-Conacyt (2009)
3. Priestley, M.: Nonlinear and Nonstationary Time Series Analysis. Academic Press, London (1988)
4. Ezekiel, S., et al.: Seismic signal analysis using correlation dimension. In: Proc. in Applied Informatics (2003)
5. Lee, F.A., Nehorai, A.: Adaptive power spectrum estimation algorithm for heart rate variability analysis. Proc. of the IEEE, 273–276 (1992)
6. Karjalainen, P.: Estimation Theoretical Background of Root Tracking Algorithms with Applications to EEG. University of Kuopio Department of Applied Physics, Report Series (1996) ISSN 0788-4672
7. Potaminos, A., Narayanan, S.: A review of the acoustic and linguistic properties of children's speech. In: IEEE 9th Workshop on Multimedia Signal Processing, MMSP, pp. 22–25 (2007)
8. Thyssen, J., Nielsen, H., Hansen, S.: Nonlinear short-term prediction in speech coding. IEEE Proceedings I, 185–188 (1994)
9. Alipoor, G., Savoji, M.: Speech coding using nonlinear prediction based on volterra series expansion. In: SPECOM, pp. 367–370 (2006)
10. Schnell, K., Lacroix, A.: Voiced excitation models for speech production based on time variable volterra systems. In: NOLISP, pp. 184–187 (2005)
11. Monin, A., Salut, G.: Arma lattice identification: A new hereditary algorithm. IEEE Trans. on Signal Processing 44(2), 360–370 (1996)

12. Etcheverry, G., Suleiman, W., Monin, A.: Quadratic system identification by hereditary approach. In: ICASSP, vol. III, pp. 129–132 (2006)
13. Ljung, L.: System identification: theory for the user. Prentice-Hall, Englewood Cliffs (1999)
14. Kantz, H., Schreiber, T.: Nonlinear Time Series Analysis. Cambridge University Press, Cambridge (1997)
15. Rugh, J.: Nonlinear System Theory: The Volterra/Wiener Approach. Wiley, Chichester (1980)
16. Li, C., Yu, J.: Volterra-tls method for chaotic time series prediction. Proc. of the IEEE, 48–51 (2008)
17. Wang, H., Gu, H.: Prediction of chaotic time series based on neural network with legendre polynomials. LNCS, vol. 5551, pp. 836–843. Springer, Heidelberg (2009)
18. Jirong, G., Xianwei, C., Jieming, Z.: An algorithm of predictions for chaotic time series based on volterra filter. In: ISECS, Proc. of the IEEE Computer Society, vol. 2, pp. 205–208 (2009)
19. Monin, A., Salut, G.: I.i.r volterra filtering with application to bilinear systems. IEEE Trans. on Signal Processing 44(9), 2209–2221 (1996)

Cascading an Emerging Pattern Based Classifier

Milton García-Borroto[1,2], José Fco. Martínez-Trinidad[2], and Jesús Ariel Carrasco-Ochoa[2]

[1] Centro de Bioplantas. Carretera a Moron km 9, Ciego de Avila, Cuba
mil@bioplantas.cu

[2] Instituto Nacional de Astrofísica, Óptica y Electrónica. Luis Enrique Erro No. 1, Sta. María Tonanzintla, Puebla, México, C.P. 72840
ariel@inaoep.mx, fmartine@inaoep.mx

Abstract. Emerging Pattern classifiers are accurate and easy to understand classifiers. However, they have two characteristics that can degrade their accuracy: global discretization of numerical attributes and high sensitivity to the support threshold value. In this paper, we introduce a novel algorithm to find emerging patterns without global discretization. Additionally, we propose a new method for building cascades of emerging pattern classifiers, which combines the higher accuracy of classifying with higher support thresholds with the lower levels of abstention of classifying with lower thresholds. Experimental results show that our cascade attains higher accuracy than other state-of-the-art classifiers, including one of the most accurate emerging pattern based classifier.

Keywords: Classifier Cascades, Understandable classifiers, Emerging pattern classifiers.

1 Introduction

The main goal of a supervised classification algorithm is to build a model based on a representative sample of the problem classes [1]. This model can be used to predict the class of new objects or to gain understanding of the problem domain.

Classification using Emerging Patterns [2] is a relatively new methodology [3]. An emerging pattern is a combination of attribute values that occurs mostly in a class, which barely appears in the remaining classes; so the presence of a pattern in a query object gives some evidence about the class the object should belong to. After Dong and Li's work, many authors propose algorithms to extract and use emerging patterns for supervised classification [2].

Emerging pattern classifiers are very valuable tools to solve real problems in many fields like Bioinformatics [4], streaming data analysis [5], intruder detection [6] and mining spatio-temporal relationships [7].

The most successful emerging pattern-based classifier family, introduced by Bailey et al. [8] in 2002, makes a global emerging pattern search in the training stage. They make a previous discretization of all numerical features, so resulting patterns use only items with the structure ($Feature = value$). The discretized objects are ranked using the information of their attribute values, and they are

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 240–249, 2010.

represented in a multi-value tree structure, which is a tree-based representation of the complete training sample. Finally, the authors make a depth first traverse of the tree to extract patterns, using a predefined threshold value for the pattern minimal support. These classifiers have two main drawbacks:

- Global discretization on numerical attributes can seriously degrade the classification accuracy.
- High sensitivity to the support threshold value, which makes very hard to find an automated method to estimate it correctly.

In this paper, we introduce a novel algorithm to find emerging patterns, which does not apply global discretization of numerical attributes. It extracts patterns from a collection of decision trees, using a specialized procedure. To find a representative collection of patterns, it uses a novel object weighting scheme. This algorithm applies local discretization, using only the attribute values appearing in the current node objects.

It is important to highlight that we are not introducing a new method for inducing decision trees or forests. Our algorithm builds decision trees to extract the emerging patterns from them, but the induced trees are completely discarded after the extraction procedure.

Additionally, we propose a new method for building cascades of emerging pattern based classifiers. The method starts estimating the minimal support used in the first classifier and the maximum abstention allowed to the last classifier. Then, it creates a cascade of classifiers using decreasing support values, from the maximum inferred value to the value that attains the maximum allowed abstention. This method obtains an ensemble that classifies unseen objects using emerging patterns with the highest possible support.

The paper is organized as follows: section 2 presents a brief revision about classification using emerging patterns, section 3 introduces the new algorithm for mining emerging patterns without global discretization, section 4 presents the method to build a cascade of emerging patterns based classifiers, section 5 shows the experimental results, and section 6 presents the conclusions.

2 Classification Using Emerging Patterns

A *pattern* is an expression, defined in a language, which describes a collection of objects; the objects described by a pattern are named the pattern *support*. In a supervised classification problem, we say that a pattern is *emerging* if its support increases significantly from one class to the others [3]. Emerging patterns are usually expressed as combinations of feature values, like $(Color = green, Sex = male, Age = 23)$ or as logical properties, like $[Color = green] \wedge [Sex = male] \wedge [Age > 23]$.

Most algorithms for emerging pattern mining have as goal to find the patterns that satisfy a desired property: being supported by a single class (JEP), minimality over subset inclusion (SJEP), or tolerance on noisy objects (NEP) [9]. These algorithms have the following steps:

1. Selection of the minimal support threshold μ
2. Global discretization of numerical attributes
3. Representation of the transformed objects using a particular structure
4. Traversing the structure to find emerging patterns
5. Pattern filtering

Using this traditional algorithm might have two important drawbacks:

1. Global discretization of numerical attributes could drastically degrade the classifier accuracy, since an emerging pattern relates a combination of feature values with a class. Therefore, discretizing a numerical attribute without considering the values of other features could hide important relations.
   In Table 1, we can see that SJEPC [9], one of the most accurate classifiers of the family, obtains very poor accuracies in databases like Iris, while all other classifiers attain accuracies above 93%. In some other databases, SJEPC is unable to extract even a pattern, because most numerical features are discretized into a single categorical value.
2. High sensitivity to the support threshold value. The accuracy of the classifier can have serious accuracy degradation on small variations of the minimal support value. For example, in *chess* and *census* databases, the accuracy drops 3% with a variation of 2 in the threshold[8].

Emerging Pattern classifiers are not frequently used in ensembles, because they are complex and stable classifiers. Nevertheless, they have been used as base classifiers in ensembles using Bagging and Boosting [10] methods. The Boosted [11] and the Bagged [12] Emerging Pattern classifiers are more accurate than the base classifier, but they do not solve the mentioned drawbacks.

3 Crisp Emerging Pattern Mining (CEPM)

In this section, we introduce CEPM, a new emerging pattern mining algorithm with local discretization of numerical features. It extracts patterns from a collection of C4.5 decision trees [13], using an special pattern mining procedure. To guarantee that CEPM finds a representative collection of patterns, it uses a novel object weighting scheme.

The tree induction procedure (InduceTree, steps 2 and 16, Algorithm 2) has the following characteristics:

- Candidate splits are binary. Nominal attributes use properties like $[Feature = a]$ and $[Feature \neq a]$ for each of its values; numerical attributes use properties like $[Feature > n]$ and $[Feature \leq n]$ for all candidate cut points.
- If a node has less than μ objects, it is not further split because it cannot generate emerging patterns.
- To select the best split, the algorithm evaluates the *weighted information gain*. The weighted information gain is a modification of the information gain to use weighted probabilities for each class and child node (Equation 1.) Note

```
Data: $T$ - training sample, $maxIter$ - maximum number of iterations, $\mu$ -
      minimum support ratio
Result: $EPS$ - Mined Patterns, $abstentionRatio$ - Abstention ratio of $EPS$
        with respect to $T$
forall $o \in T$ do $w_o \leftarrow 1$ ;
/* Simplify procedure deletes duplicated and non-minimal patterns */
$EPS \leftarrow Simplify\,(InduceTree(T, w, \mu))$ ;
$averageSupport \leftarrow \frac{\sum_{ep \in EPS} support(ep)}{|EPS|}$ ;
$i \leftarrow 0$ ;
repeat
    $AbstentionCount \leftarrow 0$ ;
    // Weight recalculation
    foreach $o \in T$ do
        $EPc \leftarrow \{$ Patterns that support $o$, belonging to its class$\}$ ;
        $EPnc \leftarrow \{$ Patterns that support $o$, belonging to a different class$\}$ ;
        $support = \sum_{ep \in EPnc} sup(ep) - \sum_{ep \in EPc} sup(ep)$ ;
        $w_o = \operatorname{arccot}\left(DesiredSupport \cdot \frac{support}{averageSupport}\right)/\pi$;
        if any pattern support $o$ then
            $AbstentionCount \leftarrow AbstentionCount + 1$

    end
    $EPS \leftarrow Simplify\,(EPS \bigcup InduceTree(T, w, \mu))$;
    $i \leftarrow i + 1$
until $i = maxIter$ OR no new pattern was added in this iteration ;
```

Algorithm 1. Pseudocode of the algorithm CEPM

that objects with weight close to 0 have low influence in the determination of the best split.

$$P_{Class} = \frac{\sum_{o \in Class} w_o}{\sum w_o}, \quad P_{child} = \frac{\sum_{o \in child} w_o}{\sum w_o} \tag{1}$$

During the tree induction, every child node having at least μ objects in a class, and at most one object in the complement of that class, generates a new emerging pattern. This pattern consists in the conjunction of the properties from the node to the root. For example, from the decision tree in Figure 1 we can extract the patterns $(Length > 30) \wedge (Color = red)$ from class **Good**, and the pattern $(Length \leq 30) \wedge (Fly = false)$ from class **Bad**.

Additionally, CEPM extracts patterns while evaluating the splits, even if a split has not the optimal gain. Any child node having at least μ objects in a class, and at most one object in the complement of that class, generates an emerging pattern. For example, Figure 2 shows two candidate splits, using different properties. Although the first one has a higher information gain, the second contains the emerging pattern $(Edad < 20)$. So, the pattern is extracted even if the split is discarded.

CEPM iteratively induces different decision trees, updating the object weights after each iteration. The algorithm updates weights using Equation 2.

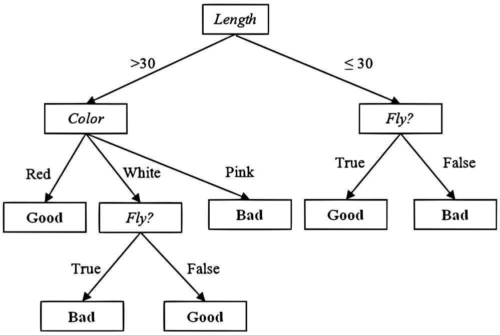


Fig. 1. Example of decision tree with three attributes and two classes

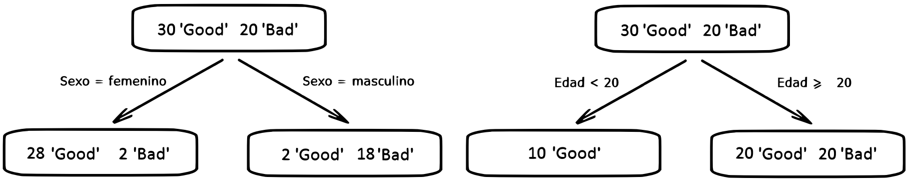


Fig. 2. Example of an emerging pattern in a non-optimal candidate split

$$w_o = \frac{\operatorname{arccot}\left(DesiredSupport \cdot \frac{Support_o}{averageSupport}\right)}{\pi} \tag{2}$$

where

- $Support_o$ is the sum of the support of the patterns contained in o. If the pattern belongs to a different class than o, its support is multiplied by -1
- $averageSupport$ is the average support of the patterns found in the first built tree
- arccot is the inverse cotangent function

Equation 2 values ranges from 0 to 1, because arccot ranges between 0 and π Figure 3. An object obtains a weight close to 1 if it has a negative total support lower than 5. In this case, it is necessary to mine more patterns to support the object to its own class. On the contrary, a weight close to 0 means the object has total support above 5. In this case, no more patterns are necessary for this object, and it is virtually ignored in gain calculations. As we can see in Figure 3, value 5 and -5 are both distinctive in the arccot function, because they are the points where the function values starts to be close to the asymptotes. That is why we use value 5 in the equation 2.

The pseudocode of CEPM appears in Algorithm 1. It is worth to mention that CEPM returns a set of minimal emerging patterns with support greater or equal to μ. Pattern minimality is considered with respect to the subset inclusion of

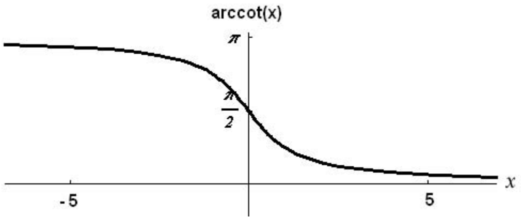


Fig. 3. The *arccot* function

the pattern component properties. Additionally, it returns the abstention ratio, which is the ratio of objects that are not covered by the resultant patterns.

4 Cascading CEPM-Based Classifiers

```
Data: $T$ - training sample
Result: $Classifiers$ - Cascade of classifiers. The first classifier is the one built
        with the highest $\mu$ value
$(MaxSupport, MaxAbstRate) \leftarrow InferParams(T)$ // in Algorithm 3
$currentSupport \leftarrow MaxSupport$ ;
$step \leftarrow \max\left\{\frac{currentSupport}{10}, 1\right\}$ ;
while $currentSupport > 1$ AND $currentAbstention > MaxAbstRate$ do
    $(patterns, currentAbstention) \leftarrow CEPM(T, maxIter = 120, \mu = currentSupport)$ ;
    if $currentAbstention < 0.5$ then
        Add a new classifier using $patterns$ to $Classifiers$
    $currentSupport \leftarrow currentSupport - step$ ;
    if $currentSupport = 0$ AND $abstentionRatio >= MaxAbstRate$ then
        $currentSupport \leftarrow step - 1$ ;
        $step \leftarrow \max\left\{\frac{currentSupport}{10}, 1\right\}$ ;
    end
end
```

Algorithm 2. Pseudocode of the algorithm CascadeCEPM

A cascade classifier is a type of ensemble where a single classifier is active at each time [10]. To classify a query object the first classifier is activated and returns the class. If a classifier is not sure enough about the correct classification, it passes the query object to the next classifier in the chain. Cascading classifiers approach is better than multi-expert methods when the topmost classifiers can handle most objects with higher accuracy, but they are unable to classify some other objects [14].

Emerging pattern classifiers with different minimal support μ are good candidates for cascading; a classifier using patterns with higher μ values, is more accurate but could reject to classify more objects. Then, to build a cascade of

```
    Data: $T$ - training sample
    Result: $MaxSup$ - Higher $\mu$ used in the classifier ensemble, $AbstentionRate$ -
            Maximum abstention rate allowed to the classifier with lowest $\mu$ in the
            ensemble
1   $(Patterns, AbstentionRate) \leftarrow CEPM(T, maxIter = 20, \mu = 2)$ ;
    /* Using the highest support value makes $MaxSupport$ to be high
       enough; checking that at least half of the objects are supported
       makes $MaxSupport$ to be not too high                            */
2   $MaxSup \leftarrow$ Highest support value such that at least half of the objects in $T$ are
    supported for at least a pattern in $Patterns$ ;
3   return $(MaxSup, AbstentionRate)$
```

Algorithm 3. Pseudocode of the algorithm InferParams

emerging pattern classifiers, the first classifier should be built with a high μ value. The remaining classifiers in the cascade should have a μ value lower than the μ value of their predecessors, for allowing them to classify uncovered objects.

Our novel cascading creation method, named CascadeCEPM, appears in Algorithm 2. It starts inferring the support of the topmost classifier ($MaxSupport$) and the maximal abstention rate allowed for the lower classifier ($MaxAbstRate$). CascadeCEPM creates classifiers starting with $\mu = MaxSupport$, decrementing μ for each new classifier until it finds an abstention rate lower than $maxAbstRate$ or $\mu = 1$. For decrementing μ, CascadeCEPM uses a calculated $Step$ (see Algorithm 2), because if $MaxSupport$ is high, decrementing μ by 1 might be too costly.

Some important remarks:

1. $MaxSupport$ is inferred based on two criteria. If it is higher than the optimum, the algorithm makes costly unnecessary iterations; otherwise, if it is lower than the optimum, better models (with higher μ) are disregarded.
2. The value 120 (Algorithm 2, Step 5) is the maximum number of iterations of the algorithm CEPM. It was introduced because, in some databases, CEPM has an slow convergence. We determine this value experimentally; using higher values does not alter the classifier accuracy.
3. $MaxAbstRate$ is inferred using $\mu = 2$, so it measures the maximum expected abstention of a pattern based classifier. A maximum iteration value equal to 20, instead of 120, speeds up the procedure (Algorithm 3, step 1).
4. We dismiss classifiers with abstention level higher than 0.5, because they were inaccurate in most of the tested databases.
5. The condition $currentAbstention < 0.5$ in Algorithm 2 (step 6) discards inaccurate classifiers, having high abstention levels.

CascadeCEPM creates a cascade of emerging pattern classifiers, each one using as the decision rule the highest sum of support. Given a query object, the first classifier returns the most supported class; if no pattern supports the object or there is a tie, the classifier refuses to classify and activates the next classifier in the cascade. If the last classifier cannot classify the query object, the whole cascade refuses to return a classification.

5 Experimental Results

To compare the performance of CascadeCEPM, we carried out some experiments over 18 well-known databases from the UCI Repository of Machine Learning [15]. We selected five state-of-the-art classifiers: 3 Nearest Neighbors [16], Bagging and Boosting [10], Random Forest [17], C4.5 [13]. For each classifier, we used the Weka 3.6.1 implementation [18] with its default parameters. We also tested the behavior of the emerging pattern based classifier SJEPC [9], with the minimal support threshold suggested by their authors.

Table 1. Accuracy results of compared classifiers. The highest accuracy per database is bolded

DBName	3NN	Boost	Bagg	C4.5	RandFor	SJEPC	CascCEPM
balance-scale	**85.44**	71.70	82.58	77.62	79.37	16.02	82.73
breast-cancer	70.31	72.40	70.96	**73.44**	65.75	44.47	72.34
cleveland	82.51	**84.15**	79.88	78.19	78.57	77.90	81.51
haberman	70.56	70.90	**72.47**	67.97	67.61	0.00	71.56
hayes-roth	71.43	53.57	75.00	**89.29**	85.71	0.00	75.00
heart-c	81.18	**83.18**	81.85	76.23	80.89	78.60	82.18
heart-statlog	79.26	**80.74**	79.26	79.26	79.26	64.81	80.00
hepatitis	81.96	81.21	**82.00**	78.75	81.33	77.46	81.33
iris	96.00	**96.67**	93.33	94.00	94.67	66.67	95.33
liver-disorders	65.47	66.08	68.68	68.70	**70.74**	0.00	69.89
lymph	**85.90**	75.67	77.67	78.48	79.86	51.48	82.52
monks-problem-1	50.00	50.00	50.00	50.00	50.00	57.87	**100.00**
monks-problem-2	51.39	50.00	55.09	59.72	58.56	34.03	**79.17**
monks-problem-3	50.00	50.00	50.00	50.00	50.00	63.66	**97.45**
spect	64.71	66.84	61.50	66.84	62.03	0.00	**83.42**
tic-tac-toe	**98.54**	73.49	91.02	83.82	91.86	91.34	94.36
vote	91.97	94.72	95.18	**96.10**	**96.10**	91.08	94.49
wine	96.05	87.48	94.31	92.65	**97.16**	55.07	94.97

We performed 10-fold cross validation, averaging the results. In both SJEPC and CascadeCEPM we reported the abstentions as errors. In these objects, the classifier is unable to assign a class; returning the majority or a random class could hide these undesirable cases. In Table 1, we can find the accuracy results, in percent.

Experimental results show that SJEPC has low accuracy values in many databases, compared with other classifiers. In those databases, most numerical attributes were transformed in a single categorical attribute, and therefore they were discarded.

In order to determine if the differences in accuracy are statistically significant, we performed a pairwise comparison between our classifier and the others. Each cell in Table 2 contains the number of databases where our classifier Win/Lose/Tie to each other classifier. We detect ties using a two-tailed T-Test

Table 2. Pairwise comparison (Win/Loss/Tie) between our classifier and the others

	3NN	AdaBoost	Bagging	C4.5	RandFor	SJEPC
cascade	8/3/7	10/1/7	6/0/12	12/1/5	10/2/6	18/0/0

[19] with significance of 0.05. The pairwise comparison shows that, in the tested databases, CascadeCEPM is more accurate than any other classifier, using the selected databases.

Like previous emerging pattern classifier, CascadeCEPM scales better to adding new objects than to adding new features.

6 Conclusions

In this paper, we introduced CEPM, a new algorithm for mining Emerging Patterns using local discretization of numerical values. Our algorithm solves the main problem of algorithms using global discretization, which makes them useless in some databases. CEPM extracts patterns from a collection of decision trees, using a specialized extraction procedure. For obtaining a collection of representative patterns, CEPM uses a novel object weighting scheme.

Additionally, this paper proposes CascadeCEPM, a new cascading method for Emerging Pattern classifiers. CascadeCEPM infers the maximal and minimal thresholds, generating a cascade of classifiers using selected threshold values in between. Experimental results show that CascadeCEPM is a more accurate method than SJEPC in most databases. A pairwise comparison reveals that it is also more accurate than other state of the art classifiers.

In the future, we will work on speeding up the algorithm to estimate the support thresholds of the classifiers in the ensemble.

Acknowledgments

This work is partly supported by the National Council of Science and Technology of México under the project CB-2008-01-106443 and grant 25275.

References

1. Berzal, F., Cubero, J.-C., Sánchez, D., Serrano, J.M.: Art: A hybrid classification model. Machine Learning 54, 67–92 (2004)
2. Ramamohanarao, K., Fan, H.: Patterns based classifiers. World Wide Web 10(1), 71–83 (2007)
3. Dong, G., Li, J.: Efficient mining of emerging patterns: Discovering trends and differences. In: Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, California, United States, pp. 43–52. ACM, New York (1999)
4. Quackenbush, J.: Computational approaches to analysis of dna microarray data. In: IMIA Yearbook of Medical Informatics, pp. 91–103 (2006)

5. Alhammady, H.: Mining streaming emerging patterns from streaming data. In: IEEE/ACS International Conference on Computer Systems and Applications (Amman), pp. 432–436 (2007)
6. Chen, L., Dong, G.: Masquerader detection using oclep: One-class classification using length statistics of emerging patterns. In: WAIMW '06: Proceedings of the Seventh International Conference on Web-Age Information Management Workshops, Washington, DC, USA, p. 5. IEEE Computer Society, Los Alamitos (2006)
7. Celik, M., Shekhar, S., Rogers, J.P., Shine, J.A.: Sustained emerging spatio-temporal co-occurrence pattern mining: A summary of results. In: ICTAI '06: Proceedings of the 18th IEEE International Conference on Tools with Artificial Intelligence, Washington, DC, USA, pp. 106–115. IEEE Computer Society, Los Alamitos (2006)
8. Bailey, J., Manoukian, T., Ramamohanarao, K.: Fast algorithms for mining emerging patterns. In: Elomaa, T., Mannila, H., Toivonen, H. (eds.) PKDD 2002. LNCS (LNAI), vol. 2431, pp. 187–208. Springer, Heidelberg (2002)
9. Fan, H., Ramamohanarao, K.: Fast discovery and the generalization of strong jumping emerging patterns for building compact and accurate classifiers. IEEE Transactions on Knowledge and Data Engineering 18(6), 721–737 (2006)
10. Kuncheva, L.I.: Combining Pattern Classifiers. Methods and Algorithms. Wiley Interscience, Hoboken (2004)
11. Sun, Y.: Boosting an associative classifier. IEEE Trans. on Knowl. and Data Eng. 18(7), 988–992 (2006); Member-Wang, Y. and Fellow-Wong, A.K.C
12. Fan, H., Fan, M., Ramamohanarao, K., Liu, M.: Further improving emerging pattern based classifiers via bagging. In: Ng, W.-K., Kitsuregawa, M., Li, J., Chang, K. (eds.) PAKDD 2006. LNCS (LNAI), vol. 3918, pp. 91–96. Springer, Heidelberg (2006)
13. Quinlan, J.R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers Inc., San Francisco (1993)
14. Alpaydin, E., Kaynak, C.: Cascading classifiers. Kybernetica 34(4), 369–374 (1998)
15. Merz, C., Murphy, P.: Uci repository of machine learning databases. Technical Report, University of California at Irvine, Department of Information and Computer Science (1998)
16. Dasarathy, B.D.: Nearest Neighbor (NN) Norms: NN Pattern Classification Techniques. IEEE Computer Society Press, Los Alamitos (1991)
17. Ho, T.K.: The random subspace method for constructing decision forests. IEEE Transactions on Pattern Analysis and Machine Intelligence 20(8), 832–844 (1998)
18. Frank, E., Hall, M.A., Holmes, G., Kirkby, R., Pfahringer, B., Witten, I.H.: 'Weka: A machine learning workbench for data mining. In: Maimon, O., Rokach, L. (eds.) Data Mining and Knowledge Discovery Handbook: A Complete Guide for Practitioners and Researchers, pp. 1305–1314. Springer, Berlin (2005)
19. Dietterich, T.G.: Approximate statistical tests for comparing supervised classification learning algorithms. Neural Computation 10(7), 1895–1923 (1998)

A New Combined Filter-Wrapper Framework for Gene Subset Selection with Specialized Genetic Operators

Edmundo Bonilla Huerta, J. Crispín Hernández Hernández,
and L. Alberto Hernández Montiel

LITI, Instituto Tecnológico de Apizaco,
Av. Instituto Tecnológico s/n, 90300 Apizaco, Mexico
{edbonn,josechh,luishm}@itapizaco.edu.mx

Abstract. This paper introduces a new combined filter-wrapper gene subset selection approach where a Genetic Algorithm (GA) is combined with Linear Discriminant Analysis (LDA). This LDA-based GA algorithm has the major characteristic that the GA uses not only a LDA classifier in its fitness function, but also LDA's discriminant coefficients in its dedicated crossover and mutation operators. This paper studies the effect of these informed operators on the evolutionary process. The proposed algorithm is assessed on a several well-known datasets from the literature and compared with recent state of art algorithms. The results obtained show that our filter-wrapper approach obtains globally high classification accuracies with very small number of genes to those obtained by other methods.

Keywords: Microarray gene expression, Feature selection, Genetic algorithms, Linear Discriminant Analysis, Filter, Wrapper.

1 Introduction

The DNA Microarray is a tool that permits to monitor and to measure gene expression levels for tens of thousands of genes simultaneously under hundreds of biological conditions. This technology enables to consider cancer diagnosis based on gene expressions [1,2,3,6]. This microarray technology enables clinicians and biologists to obtain the gene expression profile of tissue samples rapidly.

Microarray data is characterized with thousands of genes but with only a small number of samples available for analysis. Microarray data often contains many irrelevant and redundant features, which affect the speed and accuracy of most learning algorithms. The task of selecting the "best" feature subset is known as feature selection, sometimes as variable selection or subset selection [4].

Feature selection consists to select a minimal subset of m features from the original set of n features ($m \ll n$). Normally, a feature selection method consists of four components: a search mechanism, an evaluation function, a stopping criterion, and a validation procedure.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 250–259, 2010.

In this paper, we propose a new combined filter-wrapper approach for gene subset selection for microarray data classification in two stages. In the first stage is proposed a filter for individual gene ranking. In the second stage a wrapper is proposed where Fisher's Linear Discriminant Analysis (LDA) is used to provide useful information to a Genetic Algorithm (GA) for an efficient exploration of gene subsets space. LDA has been used for several classification problems and recently for microarray data [7,10,11].

This paper is organized as follows: Section 2 describes the state of the art of feature selection. In Section 3 The General Genetic algorithm procedure is discussed in detail. Experimental results are shown in Section 4 and conclusions are drawn in Section 5.

2 State of the Art

Feature selection methods are separated into three main families [12]: 1) the filter approach, 2) the wrapper approach and 3) the embedded approach.

The filter approach uses statistical information to filter out irrelevant features. All features are first ranked and then a classifier is build by selecting the highest ranking features. In most cases, the selection relies on an individual evaluation of each feature [6,13], therefore this method ignores interactions among genes.

The wrapper approach relies on a classification algorithm that is used as a black box to explore the space of features subset to evaluate each candidate subset; the quality of a candidate subset is measured by the performance of the classifier obtained on the training data, using, for example a cross validation process. Finally, the feature subset that achieves highest performance is chosen. Wrapper methods are generally computation intensive since the classifier must be trained for each candidate subset. For this reason, several strategies can be considered to explore the space of possible subsets. In the context of microarray data, several wrapper methods have been proposed using uncorrelated discriminant analysis [10], generalized discriminant analysis [31], null space [11], diagonal covariance matrices [24] and linear discriminant analysis using pseudo-inverse [5].

Finally, in embedded methods, a inductive algorithm is used as feature selector and classifier. A representative work of this approach is the method that uses support vector machines with recursive feature elimination (SVM/RFE) [18]. The selection is based on a ranking of the genes and, at each step, the gene with the smallest ranking criterion is eliminated. The ranking criterion is obtained from the weights of a SVM trained on the current set of genes. In this sense, embedded methods are similar to wrapper methods. There are other variants of these approaches, see [19,20] for two examples. In [21], the authors propose a forward search approach based in two steps. In the first a filter method is used as a pre-selection step, all features are ranked. In the second step a wrapper method is used for feature selection by building N attributes subsets. These subsets are evaluated using the so-called Rank-Search algorithm, which is very fast, but gives very large feature subsets.

3 General GA Procedure

Our approach first extracts a set of interesting genes (about $p = 150$ genes) by a filter method in order to limit the search space. Then we use a dedicated Genetic Algorithm (GA) to determine a small subset of genes that allows high classification accuracy. Contrary to most existing GAs for gene selection that rely essentially on random genetic operators, we devised a problem specific GA that takes into account useful knowledge of the gene selection and classification problem. Indeed, our GA uses a LDA classifier to assess the fitness of a given candidate gene subset and LDA's discriminate coefficients in its crossover and mutation operators.

This paper first studies the impact of these informed operators on the evolutionary process. We present experimental evidence about the performance of our dedicated crossover operator when compared with two conventional crossover operators.

To evaluate the usefulness of the proposed approach, we carry out extensive experiments on seven public datasets and compare our results with 13 best performing algorithms from the literature. We observe that our approach is able to achieve a high prediction accuracy (from 81% to 100%) with a very small number of informative genes (from 2 to 19).

3.1 Filter Method

Different feature selection approaches have been applied to gene selection for cancer classification (Fisher ratio, Wilcoxon test, Information Gain, Relief-F, T-statistics, entropy-based, BW ratio, etc.).

The **BW ratio**, introduced by Dudoit *et al.* [7], is the ratio of between-group to within-group sums of squares. For a gene j, the ratio is formally defined by:

$$BW\,(j) = \frac{\sum_i \sum_k \; I(y_i = k)(\bar{x}_{kj} - \bar{x}_j)^2}{\sum_i \sum_k \; I(y_i = k)(x_{ij} - \bar{x}_{kj})^2} \tag{1}$$

where $I(.)$ denotes the indicator function, equaling 1 if the condition in parentheses is true, and 0 otherwise. $\bar{x}_j$ and $\bar{x}_{kj}$ denote respectively the average expression level of the gene j across all samples and across samples belonging to class k only and finally y_i denotes a class label. This filter method considers the most discriminatory genes those that have the highest scores. We use this ranking-based method to identify differentially expressed genes in two tissue types.

In our filter-wrapper framework, we use three filters : between-group to within-group sums of squares (BW), t-statistic (TT), and Wilcoxon test (WT) for a comparative analysis.

3.2 Wrapper Method

We describe now the wrapper method based in a genetic algorithm and a classifier LDA (GALDA) of the general model for gene selection and classification. The

GA is designed for discovering good gene subsets by using genetic operators specialized. The LDA-based classifier is used to ensure the fitness evaluation of each candidate gene subset. The basic components of our GA are detailed in this section.

Encoding representation and initialization. In our GA, a chromosome encodes more information than a classic GA, because the chromosomes are formed by two parts of same length: 1) a binary vector and 2) a real-valued vector (as shown in Figure 1). The first part represents a candidate gene subset (150), where each allele (bit) of the chromosome represents a gene. If an allele is "1" it means that this gene is kept in the gene subset and "0" indicates that the gene is not included. Each chromosome represents thus a gene subset. The chromosome length is equal to the number of p genes pre-selected by the filter t-statistics (3.1). The second part of the chromosome is a real-valued vector that corresponds to the discriminant coefficients obtained by a LDA classifier for a gene g_i which are useful for designing powerful crossover and mutation operators (described below).

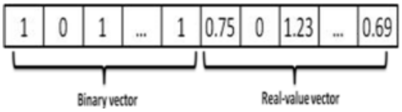


Fig. 1. Representation of a chromosome in our GA

The individual of the initial population are generated randomly according to a uniform distribution in such a way that each chromosome contains a number of genes ranging from $p = 0.60$ to $p = 0.75$. The population size is fixed at 100 in this work. The chromosomes of the current population P are sorted according to the fitness function (see subsection 3.2). The best 10% chromosomes of P are directly copied to the next population P' and removed from P. The remaining 90% chromosomes of P' are then generated by using specialized genetic operators.

Double fitness function. The fitness of a chromosome, i.e. a subset of genes, is assessed by two fitness functions: 1) the classification rate (f_1), and 2) the number of selected genes in the each chromosome (f_2). In the first part of the double fitness function, a subset of genes leading to a high classification rate is considered to be better than a subset leading to a low classification rate. In our case, the LDA classifier ensures this classification task. We apply a 10 fold cross validation to calculate the performance of the classifier LDA for a candidate gene subset. The second part of the fitness (f_2) is calculated by the formula:

$$f_2 = \left(1 - \frac{m_\omega}{p}\right) \tag{2}$$

Where m_ω is the number of bits having the value "1" in the candidate gene subset ω, p is the length of the chromosome corresponding to the number of the pre-selected genes from the filter ranking. Then the fitness function f is defined as the weighted aggregation of f_1 and f_2 as follows:

$$f(I) = \alpha f_1(I) + (1-\alpha) f_2 \; subject\, to \; 0 < \alpha < 1 \tag{3}$$

Where α is a parameter that allows us to allocate a relative importance factor to f_1 or f_2. Assigning to α a value greater than 0.5 will push the genetic search toward solutions of high classification accuracy (probably at the expense of having more selected genes). In contrast, using a small values of α helps the search toward small sized gene subsets. So varying α will change the search direction of the genetic algorithm.

Specialized genetic operator of crossover. We use the discriminant coefficients from a LDA classifier to design our specialized genetic operator of crossover. The crossover combines two parent chromosomes $I1$ and $I2$ to generate a new chromosome I_{new} in such a way that 1) top ranking genes in both parents are preserved in the child and 2) the number of selected genes in the child I_{new} is not greater than the number of selected genes in the parents. The first point ensures that "good" genes are transmitted from one generation to another while the second property is coherent with the optimization objective of small-sized gene subsets.

Specialized genetic operator of mutation. The mutation operator is used as a source of genetic variation in the population by introducing new genetic information by making local changes in a given chromosome. For binary coded GAs, this is typically realized by flipping the value of some bits ($1 \rightarrow 0$, or $0 \rightarrow 1$). In our case, mutation is used for dimension reduction; each application of mutation eliminates a single gene ($1 \rightarrow 0$). To determine which gene is removed, we use discriminant coefficients obtained from LDA classifier [5]. Given a candidate gene subset, we identify the smallest LDA discriminant coefficient and remove the corresponding gene that is the least informative gene among the current candidate gene subset.

4 Experiments on Microarray Datasets

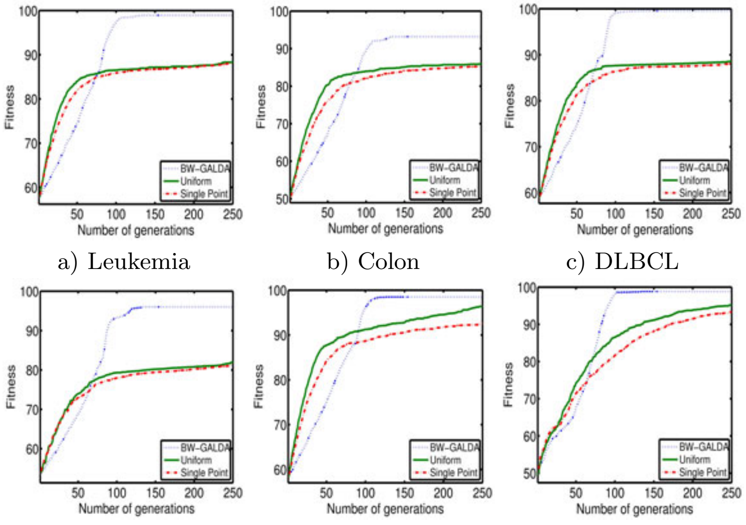
In this section, we use seven public microarray datasets for our experiments (more details in table 1). In this table is shown the number of genes, the number of samples and the first publication that has presented an analysis of this dataset.

Our filter-wrapper approach was implemented in octave for Linux O.S. A Fisher LDA classifier is used. In order to test each dataset we conduce 10 independent runs and we retain the best solution found during these 10 executions to have statistically meaningful conclusions. In the first experiment, we focus on the accuracy, so the fitness function is defined with $\alpha = 0.50$ (results shown in Figure 2).

We show in Table 2 the best accuracies (in bold) obtained by other methods and by our filter-wrapper approach on the seven datasets presented previously. An entry with the symbol (–) in this table means that the paper does not treat the corresponding dataset. All the methods reported in this table use a process of cross validation. Each cell contains the classification accuracy and the number of genes when this is available.

Table 1. Description of seven microarray datasets

Dataset	Genes	Samples	Training set	Test set	References
Leukemia	7129	72	38	34	Golub et al. [6]
Colon	2000	62	32	32	Alon et al. [2]
Lung	12533	181	102	101	Gordon et al. [22]
Prostate	12600	109	54	55	Singh et al. [23]
CNS	7129	60	30	30	Pomeroy et al. [25]
Ovarian	15154	253	91	162	Petricoin et al. [26]
DLBCL	4026	47	23	24	Alizadeh et al. [1]



a) Leukemia b) Colon c) DLBCL

d) CNS e) Lung f) Prostate

Fig. 2. Comparison of the best fitness averaged over 10 independent runs with the classical genetic algorithms: uniform crossover and single point crossover. In this experiment was used a $\alpha = 0.50$.

According to these observations, it seems clear that our filter-wrapper approach is very competitive in comparison with some top-performing methods. It is difficult to assess this statement by statistical tests since many methods only deal with two datasets.

Table 2. Comparison of our filter-wrapper approach with the most relevant works on cancer classification

Reference	Leukemia	Colon	DLBCL	Lung	Prostate	CNS
[10]	97.5	85.0	–	–	92.5	–
[27]	100(30)	91.9(30)	98(30)	100(30)	97(30)	–
[28]	91.1	95.1	–	93.2	73.5	88.3
[29]	100	93.5	–	97.2	–	–
[30]	95.9(25)	87.7(25)	93(25)	–	–	–
[31]	73.2	84.8	–	–	86.8	–
[33]	98.6(5)	87(4)	–	100(3)	–	–
[34]	95.8(20)	100(20)	95.6(20)	–	–	–
[35]	94.1(35)	83.8(23)	–	91.2(34)	–	65(46)
[36]	97.1(20)	83.5(20)	93.0(20)	–	91.7(20)	68.5(20)
[37]	100(30)	90.3(30)	92.2(30)	100(30)	95.2(30)	80(30)
[11]	83.8(100)	85.4(100)	–	–	–	–
[38]	100(4)	93.6(15)	–	–	–	–
BW-GALDA with α=0.50	99.3(±2.4)	94.1(±3.2)	99.6(±1.0)	98.6(±2.0)	99.3(±1.5)	97.8(±1.0)

We show in table 3 and table 4 the most frequent genes obtained by our model for the Leukemia and Colon by using the filters TT, BW and WT. In the first column we show the ID number for each dataset, in the second column we show their frequency and finally in third column is shown the references where is reported these gene in the literature by other well-know models.

Table 3. Summary of 10 first-genes with highest frequency for Leukemia dataset

FILTER-BW			FILTER-TT			FILTER-WT		
Gene-id	Frequency	References	Gene-id	Frequency	References	Gene-id	Frequency	References
1834	915		3847	823		3847	933	
4847	535	[6,9,8,32]	1882	582	[38,9,32]	1779	709	[9]
1239	466		3252	507	[9]	1882	693	[38,9,32]
312	383	[8]	4847	368	[6,8,9,32]	4847	584	[6,8,9,32]
1829	290		5122	367		5122	497	
2242	279		5039	312		4377	386	[9]
5501	279		1834	281	[9]	1829	318	[9]
4373	278		6041	277	[9]	4373	314	
2288	274	[8,9]	1779	223	[9]	1807	295	
6041	230	[9]	6169	182		2020	279	[9]

Table 4. Summary of 10 first-genes with highest frequency for Colon dataset

FILTER-BW			FILTER-TT			FILTER-WT		
Gene-id	Frequency	References	Gene-id	Frequency	References	Gene-id	Frequency	References
739	694		1836	532		493	879	[8]
164	670		548	526		1836	737	
576	513		792	407	[8]	1873	284	
625	464	[38]	377	392		765	277	
399	391		493	381	[8]	377	276	
1472	384		18	362	[8]	897	252	[38]
26	297		67	330		792	227	[8]
619	296		765	301		180	216	
451	294		517	288		267	205	
249	280	[38]	823	287		689	203	

In table 5 are shown the number of genes selected by our model in the seven microarray datasets. In this way we could obtain a smaller number of genes with a perfect classification accuracy in 5 of seven datasets. Those minimum set of genes can be used to differentiate the two types of diseases(p.e. ALL/AML for the Leukemia dataset). For the others datasets we obtain a very good classification for the Colon, CNS and Lung cancer.

Table 5. Best classification rates for seven public dataset using small gene subset with the filters: BW, TT and WT

Data sets	filter-BW Accuracy(Number of genes)	filter-TT Accuracy(Number of genes)	filter-WT Accuracy(Number of genes)
Leukemia	99.3(±2.3)	99.3(±3)	99.3(±2.6)
Colon	94.1(±3)	93.1(±3.1)	94.1(±2.6)
DLBCL	99.6(±1)	99.3(±3.1)	99.3(±1)
CNS	97(±1)	97.3(±1)	98.6(±1)
Lung	98.6(±2)	98.6(±2)	98.6(±2.1)
Prostate	99.3(±2)	99.3(±1.7)	99.3(±1.9)
Ovarian	97.8(±1)	99.3(±2.2)	96.3(±1.0)

5 Conclusions

In this paper we propose a new combined filter-wrapper with specialized genetic operators for the gene selection and classification of microarray gene expression. The propose approach begins with a filter that pre-selects a first set of genes (about $p = 150$ in this paper). To further explore the combinations of these genes, we rely on a hybrid Genetic Algorithm combined with Fishers Linear Discriminant Analysis. In this LDA-GA, LDA is used not only to assess the fitness of a candidate gene subset, but also to inform the crossover and mutation operators. This GA and LDA hybridization makes the genetic search highly efficient for identifying small and informative gene subsets.

We use a double function fitness that provides an interesting way for the LDA-GA to explore the gene subset space either for the minimization of the selected genes or for the maximization of the prediction accuracy.

We have extensively evaluated our filter-wrapper approach on seven public datasets using a rigorous 10-fold cross-validation process. A large comparison was carried out with 13 state-of-art algorithms that are based on a variety of methods. The results clearly show the competitiveness of our filter-wrapper approach. For all the datasets, our approach is able to select small gene subsets while ensuring the best or the second best classification rate. The proposed approach has another practically useful feature for biological analysis. In fact, instead of producing a single solution (gene subset), our approach can easily and naturally provide multiple non-dominated solutions that constitute valuable candidates for further biological investigations.

Acknowledgement

This work is supported by the PROMEP project ITAPI-EXB-000. We would like to thank the referees for their helpful questions, comments and suggestions.

References

1. Alizadeh, A., Eisen, M.B., et al.: Distinct types of diffuse large (b)-cell lymphoma identified by gene expression profiling. Nature 403, 503–511 (2000)
2. Alon, U., Barkai, N., et al.: Broad patterns of gene expression revealed by clustering analysis of tumor and normal colon tissues probed by oligonucleotide arrays. PNAS 96, 6745–6750 (1999)
3. Ben-Dor, A., Bruhn, L., et al.: Tissue classification with gene expression profiles. Journal of Computational Biology 7(3-4), 559–583 (2000)
4. Bonilla-Huerta, E., Duval, B., Hao, J.-K., et al.: A hybrid GA/SVM approach for gene selection and classification of microarray data. In: Rothlauf, F., Branke, J., Cagnoni, S., Costa, E., Cotta, C., Drechsler, R., Lutton, E., Machado, P., Moore, J.H., Romero, J., Smith, G.D., Squillero, G., Takagi, H. (eds.) EvoWorkshops 2006. LNCS, vol. 3907, pp. 34–44. Springer, Heidelberg (2006)
5. Bonilla-Huerta, E., Duval, B., Hao, J.-K., et al.: Gene selection for microarray by a LDA-based genetic algorithms. In: Chetty, M., Ngom, A., Ahmad, S. (eds.) PRIB 2008. LNCS (LNBI), vol. 5265, pp. 250–261. Springer, Heidelberg (2008)
6. Golub, T., Slonim, D., et al.: Molecular classification of cancer: Class discovery and class prediction by gene expression monitoring. Science 286, 531–537 (1999)
7. Dudoit, S., Fridlyand, J., Speed, T.: Comparison of discrimination methods for the classification of tumors using gene expression data. JASA 97, 77–87 (2002)
8. Cai, R., Hao, Z., Yang, X., Wen, W.: An efficient gene selection algorithm based on mutual information. Neurocomputing 26(3), 243–250 (2008)
9. Liao, C., Li, S., Luo, Z.: Gene selection for cancer classification using Wilcoxon Rank Sum Test and Support Vector Machine. In: International Conference on Computation Intelligence and Security, pp. 368–373 (2006)
10. Ye, J., Li, T., et al.: Using uncorrelated discriminant analysis for tissue classification with gene expression data. IEEE/ACM Trans. Comput. Biology Bioinform. 1(4), 181–190 (2004)
11. Yue, F., Wang, K., Zuo, W.: Informative gene selection and tumor classification by null space lda for Microarray data. In: Chen, B., Paterson, M., Zhang, G. (eds.) ESCAPE 2007. LNCS, vol. 4614, pp. 435–446. Springer, Heidelberg (2007)
12. Guyon, I., Elisseeff, A.: An introduction to variable and feature selection. JMLR 3, 1157–1182 (2003)
13. Furey, T.S., Cristianini, N., et al.: Support vector machine classification and validation of cancer tissue samples using microarray expression data. Bioinformatics 16(10), 906–914 (2000)
14. Li, L., Weinberg, C.R., et al.: Gene selection for sample classification based on gene expression data: study of sensitivity to choice of parameters of the GA/KNN method. Bioinformatics 17(12), 1131–1142 (2001)
15. Jourdan, L.: Metaheuristics for knowledge discovery: Application to genetic data, PhD thesis, University of Lille (2003) (in French)
16. Peng, S., Xu, Q., et al.: Molecular classification of cancer types from microarray data using the combination of genetic algorithms and support vector machines. FEBS Letter 555(2), 358–362 (2003)
17. Reddy, A.R., Deb, K.: Classification of two-class cancer data reliably using evolutionary algorithms, Technical Report. KanGAL (2003)
18. Guyon, I., Weston, J., et al.: Gene selection for cancer classification using support vector machines. Machine Learning 46(1-3), 389–422 (2002)

19. Saeys, Y., Aeyels, S., et al.: Feature selection for splice site prediction: A new method using eda-based feature ranking. BMC Bioinformatics, 5–64 (2004)
20. Goh, L., Song, Q., Kasabov, N.: A novel feature selection method to improve classification of gene expression data. In: Proc. of the 2nd Asia-Pacific Conference on Bioinformatics, ACS, Darlinghurst, Australia, pp. 161–166 (2004)
21. Hall, M., Holmes, G.: Benchmarking attribute selection techniques for discrete class data mining. IEEE Trans. Knowl. Data Eng. 15(6), 1437–1447 (2003)
22. Gordon, G.J., Jensen, R.V., et al.: Translation of microarray data into clinically relevant cancer diagnostic tests using gene expression ratios in lung cancer and mesothelioma. Cancer Research 17(62), 4963–4967 (2002)
23. Singh, D., Febbo, P., et al.: Gene expression correlates of clinical prostate cancer behavior. Cancer Cell 1, 203–209 (2002)
24. Piqué-Regí, R., Ortega, A., Asgharzadeh, S.: Sequential diagonal linear discriminant analysis (SeqDLDA) for microarray classification and gene identification. Computational Systems and Bioinformatics (2005)
25. Pomeroy, S.L., Tamayo, P., et al.: Prediction of central nervous system embryonal tumour outcome based on gene expression. Nature 415, 436–442 (2002)
26. Petricoin, E.F., Ardekani, A.M., et al.: Use of proteomic patterns in serum to identify ovarian cancer. Lancet 359, 572–577 (2002)
27. Liu, H., Li, J., Wong, L.: A comparative study on feature selection and classification methods using gene expression profiles and proteomic pattern. Genomic Informatics 13, 51–60 (2002)
28. Tan, F., Fu, X., et al.: Improving Feature Subset Selection Using a Genetic Algorithm for Microarray Gene Expression Data. In: CEC-IEEE, pp. 2529–2534 (2006)
29. Ding, C., Peng, H.: Minimum redundancy feature selection from Microarray gene expression data. Bioinformatics and Computational. Biology 3(2), 185–206 (2005)
30. Cho, S.B., Won, H.H.: Cancer classification using ensemble of neural networks with multiple significant gene subsets. Applied Intelligence 26(3), 243–250 (2007)
31. Yang, W.H., Dai, D.Q., Yan, H.: Generalized discriminant analysis for tumor classification with gene expression data. Machine Learning and Cybernetics 1, 4322–4327 (2006)
32. Yang, P., et al.: A multi-filter enhanced genetic ensemble system for gene selection and sample classification of microarray data. BMC Bioinformatics 11(suppl. 1), S6 (2010)
33. Peng, Y., Li, W., Liu, Y.: A hybrid approach for biomarker discovery from Microarray gene expression data. Cancer Informatics 2, 301–311 (2006)
34. Wang, Z., Palade, V., Xu, Y.: Neuro-fuzzy ensemble approach for Microarray cancer gene expression data analysis. In: Proc. E. Fuzzy Systems, pp. 241–246 (2006)
35. Pang, S., Havukkala, I., et al.: Classification consistency analysis for bootstrapping gene selection. Neural Computing and Applications 16, 527–539 (2007)
36. Li, G.Z., Zeng, X.Q., et al.: Partial least squares based dimension reduction with gene selection for tumor classification. In: BIBE-IEEE, pp. 1439–1444 (2007)
37. Zhang, L., Li, Z., Chen, H.: An effective gene selection method based on relevance analysis and discernibility matrix. In: Zhou, Z.-H., Li, H., Yang, Q. (eds.) PAKDD 2007. LNCS (LNAI), vol. 4426, pp. 1088–1095. Springer, Heidelberg (2007)
38. Li, S., Wu, X., Hu, X.: Gene selection using genetic algorithm and support vectors machines. Soft Computing 12(7), 693–698 (2008)

Hybrid Feature Selection Method for Supervised Classification Based on Laplacian Score Ranking

Saúl Solorio-Fernández, J. Ariel Carrasco-Ochoa, and José Fco. Martínez-Trinidad

National Institute for Astrophysics, Optics and Electronics
Luis Enrique Erro # 1, 72840, Santa María Tonantzintla, Puebla, Mexico
{sausolofer,ariel,fmartine}@ccc.inaoep.mx

Abstract. In this paper, we introduce a new hybrid filter-wrapper method for supervised feature selection, based on the Laplacian Score ranking combined with a wrapper strategy. We propose to rank features with the Laplacian Score to reduce the search space, and then we use this order to find the best feature subset. We compare our method against other based on ranking feature selection methods, namely, Information Gain Attribute Ranking, Relief, Correlation-based Feature Selection, and additionally we include in our comparison a Wrapper Subset Evaluation method. Empirical results over ten real-world datasets from the UCI repository show that our hybrid method is competitive and outperforms in most of the cases to the other feature selection methods used in our experiments.

Keywords: Supervised Feature Selection, Laplacian Score, Feature Ranking.

1 Introduction

Feature selection has been an active research area in Pattern Recognition, Data Mining and Machine Learning. The main idea of feature selection is to get a subset of features for representing the data, so that those features with low relevance are eliminated and therefore they are not taken into account for further analysis. The main goal of feature selection for supervised classification is to find a feature subset that would produce high classification accuracy [25]. Feature selection not only reduces the size of the data and run-time of learning algorithms, but also leads to more compact learning models and possibly with better generalization capability [13].

There are two main approaches for feature selection in supervised classification: filter methods and wrapper methods. The filter methods [28,30], perform the selection based on inherent properties of data such as variance, entropy, mutual information or correlation, these methods are generally fast and scalable [16]. Meanwhile, wrapper methods [12,20,26] select feature subsets based on the precision of a specific classifier. Wrapper methods are often characterized by a high quality of the selected feature subsets, but these methods have a high computational cost [16]. Additionally, hybrid filter-wrapper methods have been developed for supervised feature selection [21,24], these methods attempt to have a reasonable compromise between efficiency (computational effort) and effectiveness (accuracy of classification).

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 260–269, 2010.

In this paper, we propose a new hybrid filter-wrapper feature selection method for supervised classification based on the Laplacian Score ranking combined with a wrapper strategy. Experiments show that this combination selects relevant features that provide good accuracy.

The remainder of this paper is organized as follows. Section 2, gives a brief description of each one of the supervised feature selection methods compared in ours experiments. Section 3 describes the Laplacian Score. In section 4, we present the proposed method. Section 5 shows experimental results. Finally, section 6 provides some conclusions and future work.

2 Feature Selection Techniques in Supervised Classification

There is a considerable amount of literature about feature selection [14,15,20,24,25] for supervised classification, an excellent survey about feature selection algorithms can be found in [11]. In this section, we only present some ranking based methods for supervised feature selection, which are some of the most used in the last years of feature selection for Pattern Recognition and Machine Learning [18]. We also include the description of the WPR method, which is also used in our experiments.

2.1 Information Gain Attribute Ranking (IG)

This is one of the simplest (and fastest) feature ranking methods and it is often used in text categorization [22]. The idea behind IG is to select features that reveal most information about the classes. Ideally, such features are highly discriminative and their values occur mostly in a single class.

2.2 Relief (RLF)

Relief is an instance based feature ranking scheme introduced by Kira and Rendell [30] and later enhanced by Kononenko [28]. Relief works by randomly sampling an instance from the data and then locating its nearest neighbor from the same and opposite classes. In Relief a relevance weight is given to each feature for reflecting its ability to discern between classes. An overview of this algorithm can be found in [5].

2.3 Correlation-Based Feature Selection (CFS)

Correlation-Based Feature Selection [23] evaluates feature subsets rather than individual features. The heart of the algorithm is a subset evaluation heuristic that takes into account the usefulness of individual features for predicting the class along with the level of intercorrelation among them. The heuristic of CFS assigns high scores to subsets containing features that are highly correlated with the class and have low intercorrelation with each other.

2.4 Wrapper Subset Evaluation (WRP)

As described at the section 1, wrapper feature selection methods use a classifier algorithm to estimate the worth of feature subsets [26]. The WRP method performs a greedy forward search through the space of feature subsets, starting with the empty set and stopping when the addition of any remaining features results in an accuracy decrease. In this method, Cross-Validation is used to estimate the accuracy of the classifier for a set of features.

3 Laplacian Score

Recently, a family of feature selection algorithms based on the spectral graph theory has been developed [1,3,4,7,9]. This family of algorithms evaluates features according to their agreement with the graph Laplacian matrix of similarities of the data.

Formally, given a dataset consisting of m vectors $\{\boldsymbol{x}_i\}_{i=1}^{m}$, a matrix of similarities $W_{m\times m}$ that represents the similarity or adjacency between $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ data points (instances) can be constructed. We can interpret W as a weighted graph, whose nodes represent the instances, and the set of edges contains a connection for each pair of nodes i, j with weight w_{ij}, depending on the type of the graph one wants to use[1], such as the k-nearest neighbor or the fully connected graph [6] (Laplacian Score generally use k-nearest neighbor). The Laplacian matrix L is defined as:

$$L = D - W \tag{1}$$

Where D is a diagonal matrix such that $d_{ii} = \sum_{j=1}^{m} w_{ij}$.

Given a graph G, the Laplacian matrix L is a linear operator on a vector $\boldsymbol{f} \in \mathbb{R}^m$ [6,7], where:

$$\boldsymbol{f}^T L \boldsymbol{f} = \frac{1}{2}\sum_{i\neq j} w_{ij}\left(f_i - f_j\right)^2 \tag{2}$$

Equation (2) quantifies how much the vector $\boldsymbol{f}$ locally varies in G [1]. This fact motivates to use L to measure on a vector of values of a feature, the consistency of this feature regarding the structure of the graph G. A feature is consistent with the structure of a graph if it takes similar values for instances that are near each other in the graph, and dissimilar values for instances that are far from each other. Thus a consistent feature would be relevant to separate the classes [7,6].

The Laplacian Score [4], proposed by X. He et al. [9], assesses the significance of individual features taking into account the local preserving power and its consistency with the structure of the similarity graph.

If we denote $\boldsymbol{f_r} = (f_{r1}, f_{r2}, \dots, f_{rm})^T$ with $r = 1,2,\dots,n$, as the r-th feature and its values for the m instances. Then the Laplacian Score for $\boldsymbol{f_r}$ is calculated as:

$$L_r = \frac{\tilde{\boldsymbol{f}}_r^T L \tilde{\boldsymbol{f}}_r}{\tilde{\boldsymbol{f}}_r^T D \tilde{\boldsymbol{f}}} \tag{3}$$

Where $\tilde{\boldsymbol{f}}_r = \tilde{\boldsymbol{f}}_r - (\boldsymbol{f}_r^T D\mathbf{1}/\mathbf{1}^T D\mathbf{1})\mathbf{1}$, L is the Laplacian matrix of the graph G, D is the degree diagonal matrix, and $\tilde{\boldsymbol{f}}_r$ represents the deviation from the mean of all observations of the $\boldsymbol{f}$ vector.

[1] For the Laplacian Score, we construct the graph putting an edge between two nodes if they belong to the same class and they are among the k nearest neighbors of each other.

For the Laplacian Score, the local structure of the data space is more important than the global structure [9]. In order to model the local structure, this method constructs a *k*-nearest neighbor graph, where *k* is the degree of neighborhood for each instance in the graph (see [9] for details). This value must be specified a priori by the user and it models local neighborhood relations between instances.

According to the Laplacian Score, a "good" feature should have a small value for L_r [9]. Thus, the features are arranged in a list according to their relevance. Those features that are at the top of the list are those with smaller values for L_r; this features will be considered as the most important.

4 Proposed Method

In this section, we introduce the proposed feature selection method, which follows a hybrid filter-wrapper strategy and consists of two basic steps: I) Feature Ranking. II) A wrapper selection of a subset of relevant features that provides high accuracy for classification. In the first step (filter strategy), features are sorted according to their relevance by applying the Laplacian Score. In this step, we aim to identify the features that are consistent with the structure of the data and they are sorted according to their relevance, in order to narrow the search space of possible subsets of features

```
HIBRID LAPLACIAN SCORE FEATURE SELECTION METHOD (dataset,gnd,$k_{LS}$)

{dataset: Dataset with $m$ instances and $n$ features
gnd: label information for each data point
$k_{LS}$:  number of neighbors for the Laplacian Score Graph }

Begin
      ACCBest ← $-\infty$
      indSbest ← $\emptyset$
      S ← $\emptyset$;
      {rank of features with Laplacian Score}
      indRank ← laplacianScore(dataset,gnd,$k_{LS}$)

      for i=1 to n do
              S ← S $\cup$ indRank[i]
              dataCand ← dataset[S]
              {Run Classifier with 10-fold cross-validation for
              dataCand}
              AverageACC ← Classifier (dataCand,gnd)
              If AverageACC > ACCBest then
                      ACCBest ← AverageACC
                      indSbest ← S
              end if
      end for
      return {indBest, ACCBest}
end
```

Fig. 1. Pseudocode for the Hybrid Laplacian Score Feature Selection method (LS-FS)

(2^n subsets) and starting the second step with a good approximation. In the second step (wrapper stage), the idea is to evaluate the features considered as a subset rather than individually; in this step, the algorithm builds *n* feature subsets: the first set only contains the top ranked feature, the second set contains the two top-ranked features, the third set contains the three top-ranked features, and so on (see algorithm in Fig. 1). In our method, to evaluate the accuracy of the subsets of features, we use the target classifier. The accuracy of the classification model generated by each feature subset is estimated using 10-fold cross-validation.

The pseudocode of our method is described in the algorithm of figure 1, called Hybrid Laplacian Score Feature Selection Method (LS-FS). In the algorithm of figure 1, *Classifier* can be any supervised classifier.

5 Experimental Results

In order to show the performance of the proposed method, we compared it against Information Gain Attribute Ranking [22], Relief [30], Correlation-based Feature Selection [23], and Wrapper Subset Evaluation [26] methods. For the comparison, we used ten real datasets taken from the UCI repository [8]. The details of these datasets are shown in Table 1.

In order to compare the effectiveness for feature selection, the feature sets chosen by each selector were tested with three classifiers: a decision tree (C4.5 [29]), a probabilistic classifier (naïve Bayes [27]) and k-NN [31], an instance-based classifier. These three classifiers were chosen because they represent three quite different approaches for supervised classification.

In order to evaluate the accuracy with and without feature selection the percentage of correct classification (Accuracy), averaged over ten-fold cross validation, was calculated for each selector-dataset combination before and after feature selection.

For IG and RLF selectors, which are based on ranking, dimensionality reduction was accomplished using the wrapper step of our proposed method.

To perform the experiments we used Weka[2] [2] (Waikato Environment for Knowledge Analysis) an open-source Java-based machine learning workbench.

Table 1. Details of the used datasets

Datasets	No. of Instances	No. of Features	No. of Classes
Iris	150	4	3
Wine	178	13	3
Ionhospere	351	34	2
Sonar	208	60	2
Pima	768	8	2
Wdbc	568	30	2
Glass	213	9	7
Monks-3	432	6	2
Parkinsons	194	22	2
Vehicle	845	18	4

In the Laplacian Score, the only parameter that can vary is *k* (the number of neighbors considered for building the graph), which could affect the selected subsets

[2] http://www.cs.waikato.ac.nz/~ml/weka/

of features, usually *k* must be equal to or greater than 5 [9]. In our experiment we have taken k = 33% of the size of the dataset, this value according to Liu Rongyan [3] is close to the "optimum".

Table 2 shows the results for feature selection with naive Bayes. This table shows how often each feature selection method performs significantly better (denoted as *) or worse (denoted as ●) than performing no feature selection (column 2). We consider that the results are significantly different if the difference is statistically significant at a 5% level according to a paired two-sided t test [19, 10]. From Table 2 it can be seen that all the tested selectors are on average better than no selection, although WRP is the best selector in this experiment, which improves significant the performance on four datasets. LS-FS (proposed method) got the second best average with two significant improvements and no degradation. IG got the third best average, which improves the performance on one datasets. RLF and CFS were the fourth and fifth best feature selection methods respectively for the Naïve Bayes Classifier.

Table 2. Results of feature selection for naive Bayes

Datasets	NB (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	82.62	83.19	87.76	89.76	88.62*	90.90*
Iris	96.00	96.66	95.33	95.33	96.00	94.00
Monks-3	92.35	95.60	94.69	93.99	95.85	94.67
Pima	76.31	76.30	75.14	75.65	77.86	75.13
Sonar	67.88	78.36*	74.52	70.69	68.36	75.07
Wdbc	93.15	94.19	93.32	94.55	94.73	96.66*
Wine	96.63	97.19	97.19	97.19	97.22	96.60
Glass	46.00	49.76	48.35	45.06	46.90	57.27
Parkinsons	69.07	79.38*	83.95*	82.34*	77.71	80.32*
Vehicle	46.03	46.03	45.68	49.46*	45.55	53.96*
Average	**76.60**	**79.66**	**79.59**	**79.40**	**78.88**	**81.45**

*, ● statistically significant improvement or degradation

Table 3 shows the results for feature selection with k-NN (using k=1). We can see that the best results were obtained by our proposed method (LS-FS), which significant improves the performance on two datasets, and it was the best on average. Similar results were obtained with k=3 and k=5 (see table 4 and table 5) respectively, in which the proposed method was the best on average compared with the other features selectors.

Table 3. Results of feature selection for k-NN (using k=1)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	86.33	89.74	88.90	89.17	89.17	87.75
Iris	95.33	96.0	94.00	94.00	96.00	94.0
Monks-3	76.18	100.0*	97.21*	100.00*	96.07*	100.00*
Pima	70.17	70.44	68.88	68.74	70.97	69.79
Sonar	86.57	87.50	86.57	87.48	84.57	82.6
Wdbc	95.25	96.12	95.07	95.07	95.08	95.07
Wine	94.97	97.19	97.19	94.93	96.08	94.9
Glass	69.52	77.46*	75.63	74.22	78.46*	69.96
Parkinsons	95.29	96.39	94.24	96.37	88.18	90.24
Vehicle	70.28	71.24	71.35	69.46	61.18●	67.68
Average	**83.99**	**88.20**	**86.90**	**86.94**	**85.58**	**85.2**

*, ● statistically significant improvement or degradation

Table 4. Results of feature selection for k-NN (using k=3)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	86.60	88.31	89.76	90.33	88.90	90.04
Iris	95.33	96.66	95.33	95.33	94.67	94.67
Monks-3	97.21	100.0*	97.21	97.21	96.07	100.0*
Pima	72.65	74.08	70.05	73.17	73.70	72.65
Sonar	86.02	85.57	85.55	83.14	83.17	77.38●
Wdbc	96.48	97.18	96.66	96.13	95.60	95.43
Wine	94.97	97.19*	97.16	96.60	97.22*	93.79
Glass	69.48	75.11*	73.77	74.24	69.09	73.77
Parkinsons	95.36	94.32	95.84	88.68	94.82	95.84
Vehicle	70.29	74.55	73.84	57.15●	68.16	73.84
Average	**86.44**	**88.29**	**87.52**	**85.20**	**86.14**	**86.74**

*, ● statistically significant improvement or degradation

Table 5. Results of feature selection for k-NN (using k=5)

Datasets	k-NN (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	84.90	88.88	90.04	90.03	87.48	90.90*
Iris	95.33	96.66	95.33	95.33	96.67	93.33
Monks-3	92.12	100.0*	97.21*	97.21 *	96.07*	100.00*
Pima	73.18	74.68	73.17	73.69	72.92	72.40
Sonar	84.62	86.05	80.71	79.29	83.12	81.69
Wdbc	96.13	97.18	96.66	96.48	96.66	95.78
Wine	95.52	96.62	97.16	95.46	96.11	96.63
Glass	69.09	72.30	71.39	71.39	71.39	71.41
Parkinsons	92.74	94.32	90.74	94.29	88.66	92.21
Vehicle	71.12	71.83z	71.48	72.67	61.77●	68.04
Average	**85.48**	**87.85**	**86.39**	**86.58**	**85.08**	**86.24**

*, ● statistically significant improvement or degradation

In table 6, we can observe that the proposed method (LS-FS) improves significantly on one dataset and also was the best on average for C4.5 classifier. RLF was the second best on average although there was no significant improvement or degradation in any dataset. IG was the third best selector with one degradation. WPR was the fourth best selector with no significant improvement or degradation. On the contrary, CFS degraded the performance in two datasets; this selector was the worst for C4.5 classifier.

Table 6. Results of feature selection for C4.5

Datasets	C4.5 (No sel.)	LS-FS	IG	RLF	CFS	WRP
Ionosphere	91.46	91.45	92.89	92.6	89.75	91.46
Iris	96.00	96.00	93.33●	94.0	96.00	92.67
Monks-3	100.0	100.0	100.00	100.0	96.07●	100.0
Pima	73.83	74.47	73.18	73.71	75.01	73.44
Sonar	71.17	79.80*	77.86	75.93	73.50	74.05
Wdbc	94.02	94.71	94.38	95.78	94.38	94.02
Wine	93.86	94.94	96.08	94.41	93.86	94.41
Glass	67.60	71.83	69.44	68.48	68.94	65.24
Parkinsons	85.56	87.72	87.11	90.18	89.16	87.13
Vehicle	71.95	72.66	71.46	71.82	65.78●	71.11
Average	**84.55**	**86.36**	**85.57**	**85.69**	**84.25**	**84.35**

*, ● statistically significant improvement or degradation

Finally, in table 7 we report the run-time[3] required by each one of the selectors. In this table we can see that the best results are achieved by the proposed method for the three classifiers. It is important to highlight the superiority of our method over the other methods.

Table 7. Run-time of feature selection methods (in seconds)

Datasets	LS-FS			IG			RLF			CFS			WRP		
	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5	Bayes	KNN	C4.5
Ionosphere	3.21	1.94	8.88	4.02	14.51	36.33	4.19	14.23	37.48	4.10	14.62	37.51	6.39	29.08	43.4
Iris	0.14	0.10	0.13	0.05	0.14	0.14	0.05	0.17	0.13	0.06	0.15	0.14	0.09	0.35	0.22
Monks-3	0.27	0.31	0.28	0.24	0.91	0.19	0.27	1.20	0.26	0.21	1.00	0.18	0.35	2.43	0.47
Pima	0.63	0.94	1.15	0.47	3.99	3.51	0.76	4.44	3.51	0.46	3.80	3.62	1.06	7.86	4.15
Sonar	5.61	2.98	15.23	6.49	17.87	52.55	6.95	17.89	57.02	5.98	18.83	57.13	4.65	59.76	66.6
Wdbc	3.88	3.03	9.08	5.61	20.30	31.39	6.06	22.24	34.43	5.67	21.91	37.97	6.91	81.15	40.0
Wine	0.56	0.29	0.78	0.40	0.83	1.61	0.44	0.92	1.65	0.40	0.86	1.86	1.63	4.06	3.19
Glass	0.43	0.22	0.95	0.30	0.61	2.19	0.33	0.78	2.26	0.31	0.66	2.64	0.48	2.99	6.08
Parkinsons	1.11	0.59	2.25	0.81	2.10	5.77	0.95	2.30	6.51	0.90	2.28	6.89	1.03	9.24	8.38
Vehicle	2.41	2.55	8.26	2.05	12.33	24.90	3.04	14.44	28.91	2.55	16.75	28.58	3.36	63.50	82.5
Average	**1.83**	**1.30**	**4.70**	**2.04**	**7.36**	**15.86**	**2.30**	**7.86**	**17.22**	**2.07**	**8.08**	**17.65**	**2.59**	**26.04**	**25.53**

6 Conclusion and Future Work

In this paper, we have presented a new hybrid filter-wrapper method for supervised feature selection based on the Laplacian Score ranking.

From the experiments we can conclude that the proposed method performs better using the k-NN and C.45 classifiers, where it is better than the other evaluated feature selectors, reaching on average a precision around 87%. For the Naïve Bayes classifier, the proposed method was the second best feature selector after WRP, reaching on average precision of 79.6%. Also, it is found that the proposed method requires less run-time compared to other feature selection methods.

As future work, we will explore the use of other classifiers and other feature subset search strategies in the wrapper step of our method.

References

1. García, D.G., Rodríguez, R.S.: Spectral clustering and feature selection for microarray data. In: Fourth International Conference on Machine Learning and Applications, pp. 425–428 (2009)
2. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)
3. Liu, R., Yang, N., Ding, X., Ma, L.: An unsupervised feature selection algorithm: Laplacian Score combined with distance-based entropy measure. In: Workshop on Intelligent Information Technology Applications, vol. 3, pp. 65–68 (2009)

[3] The run-times reported in this paper were obtained using a computer with an Intel Core i5 2.27GHz, with 4MB RAM.

4. Niijima, S., Okuno, Y.: Laplacian linear discriminant analysis approach to unsupervised feature selection. IEEE/ACM Transactions on Computational Biology and Bioinformatics 6(4), 605–614 (2009)
5. Jensen, R., Shen, Q.: Computational intelligence and feature selection: rough and fuzzy approaches, pp. 61–84. Wiley, Chichester (2008)
6. von Luxburg, U.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)
7. Zhao, Z., Liu, H.: Spectral feature selection for supervised and unsupervised learning. In: ICML '07: Proceedings of the 24th International Conference on Machine learning, pp. 1151–1157. ACM, New York (2007)
8. Asuncion, A., Newman, D.J.: UCI Machine Learning Repository. School of Information and Computer Science. University of California, Irvine (2007), `http://www.ics.uci.edu/~mlearn/MLRepository.html`
9. He, X., Cai, D., Niyogi, P.: Laplacian Score for feature selection. In: Weiss, Y., Schölkopf, B., Platt, J. (eds.) Advances in Neural Information Processing Systems, vol. 18, pp. 507–514. MIT Press, Cambridge (2006)
10. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
11. Liu, H., Yu, L.: Toward integrating feature selection algorithms for classification and clustering. IEEE Transactions on Knowledge and Data Engineering 17(4), 491–502 (2005)
12. Loughrey, J., Cunningham, P.: Using Early-Stopping to Avoid Overfitting in Wrapper-Based Feature Selection Employing Stochastic Search. Technical Report (TCD-CS-2005-37). Department of Computer Science, Trinity College Dublin, Dublin, Ireland (2005)
13. Pal, S.K., Mitra, P.: Pattern Recognition Algorithms for Data Mining, pp. 59–82. Chapman & Hall/CRC (2004)
14. Yu, L., Liu, H.: Efficient feature selection via analysis of relevance and redundancy. J. Mach. Learn. Res. 5, 1205–1224 (2004)
15. Zhang, L., Sun, G., Guo, J.: Feature selection for pattern classification problems. In: International Conference on Computer and Information Technology, pp. 233–237 (2004)
16. Guyon, I.: An introduction to variable and feature selection. Journal of Machine Learning Research 3, 1157–1182 (2003)
17. Kim, Y.S., Nick Street, W., Menczer, F.: Feature selection in data mining, pp. 80–105 (2003)
18. Hall, M.A., Holmes, G.: Benchmarking attribute selection techniques for discrete class data mining. IEEE Transactions on Knowledge and Data Engineering 15, 1437–1447 (2003)
19. Nadeau, C., Bengio, Y.: Inference for the generalization error. Mach. Learn. 52(3), 239–281 (2003)
20. Xing, E.P., Jordan, M.I., Karp, R.M.: Feature selection for high-dimensional genomic microarray data. In: Proceedings of the Eighteenth International Conference on Machine Learning, pp. 601–608 (2001)
21. Das, S.: Filters, wrappers and a boosting-based hybrid for feature selection. In: ICML '01: Proceedings of the Eighteenth International Conference on Machine Learning, pp. 74–81. Morgan Kaufmann Publishers Inc., San Francisco (2001)
22. Dumais, S., Platt, J., Heckerman, D., Sahami, M.: Inductive learning algorithms and representations for text categorization. In: Proceedings of the International Conference on Information and Knowledge Management, pp. 148–155 (1998)
23. Hall, M.A.: Correlation-based feature selection for machine learning. PhD thesis, Department of Computer Science, University ofWaikato, Hamilton, New Zealand (1998)

24. Dash, M., Liu, H.: Hybrid search of feature subsets. In: PRICAI'98: Topics in Artificial Intelligence, pp. 238–249 (1998)
25. Dash, M., Liu, H.: Feature selection for classification. Intelligent Data Analysis 1, 131–156 (1997)
26. Kohavi, R., John, G.H.: Wrappers for feature subset selection. Artificial Intelligence 97, 273–324 (1997)
27. John, G.H., Langley, P.: Estimating Continuous Distributions in Bayesian Classifiers. In: Eleventh Conference on Uncertainty in Artificial Intelligence, San Mateo, pp. 338–345 (1995)
28. Kononenko, I.: Estimating attributes: Analysis and extensions of relief. In: Proceedings of the Seventh European Conference on Machine Learning, pp. 171–182. Springer, Heidelberg (1994)
29. Quinlan, R.: C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo (1993)
30. Kira, K., Rendell, L.: A practical approach to feature selection. In: Proceedings of the Ninth International Conference on Machine Learning, pp. 249–256. Morgan Kaufmann, San Francisco (1992)
31. Cover, T.M., Hart, P.E.: Nearest neighbor pattern classification. IEEE Transactions on Information Theory 13(1), 21–27 (1967)

Navigating K-Nearest Neighbor Graphs to Solve Nearest Neighbor Searches

Edgar Chávez[1,2] and Eric Sadit Tellez[1]

[1] Universidad Michoacana. México
[2] CICESE. México

Abstract. Nearest neighbor queries can be satisfied, in principle, with a greedy algorithm under a proximity graph. Each object in the database is represented by a node, and proximal nodes in this graph will share an edge. To find the nearest neighbor the idea is quite simple, we start in a random node and get iteratively closer to the nearest neighbor following only adjacent edges in the proximity graph. Every reachable node from current vertex is reviewed, and only the closer-to-the-query node is expanded in the next round. The algorithm stops when none of the neighbors of the current node is closer to the query. The number of revised objects will be proportional to the diameter of the graph times the average degree of the nodes. Unfortunately the degree of a proximity graph is unbounded for a general metric space [1], and hence the number of inspected objects can be linear on the size of the database, which is the same as no indexing at all.

In this paper we introduce a *quasi*-proximity graph induced by the all-k-nearest neighbor graph. The degree of the above graph is bounded but we will face local minima when running the above greedy algorithm, which boils down to have false positives in the queries.

We show experimental results for high dimensional spaces. We report a recall greater than 90% for most configurations, which is very good for many proximity searching applications, reviewing just a tiny portion of the database.

The space requirement for the index is linear on the database size, and the construction time is quadratic in worst case. Relaxations of our method are sketched to obtain practical subquadratic implementations.

1 Introduction and Related Work

Nearest neighbor search is a fundamental problem with a very large number of applications, ranging from pattern recognition, knowledge discovery and probability density estimation to multimedia information retrieval.

A nearest neighbor search (NNS) consist in finding the closest object to a given query among a dataset equipped with a distance function. The problem can be easily solved by a sequential scan of the dataset, but in a number of situations this solution is not acceptable; for example when comparing objects is expensive and/or when the dataset is very large. To avoid the sequential scan the dataset must be preprocessed. A data structure obtained from this

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 270–280, 2010.

preprocessing is usually called an index. The simpler (and older) applications are related to spatial queries, in this particular case it is possible to obtain logarithmic complexity guarantees. Multidimensional spatial queries still have logarithmic guarantees, but the complexity is exponential on the dimension of the space, hence above certain dimension a sequential scan is faster than using an index. This is called the *curse of dimensionality* and one of the motivations of stating the problem in a coordinate-free model is to avoid such exponential dependence on the dimension. Unfortunately, this problem also appear in the coordinate-free model, being the only resource to maintain scalable indexes the use of approximate and probabilistic methods.

The most basic model of complexity for this problem is just to count the number of distance computations performed to answer a query. This model assume all other operations have a time complexity that could be neglected. To have this in perspective think in the cost of computing the distance between two fingerprints, two faces or two complex objects in general. The time to compute the distance in this examples is in the order of seconds, while the time to traverse a data structure could be orders of magnitude smaller. We will stick to this model for its simplicity and because we are aiming at answering a basic, theoretical question instead of an application.

1.1 The Metric Proximity Searching Model

Formally the database of objects is a finite sample $\mathbb{S}$ of a (possibly) infinite set $\mathbb{X}$ and the objects are comparable only with a distance $d(\cdot,\cdot) : \mathbb{X} \times \mathbb{X} \rightarrow \mathbb{R}^+$. The pair $(\mathbb{X}, d)$ is called a *metric space.* The distance is usually assumed to obey the triangle inequality, be symmetric, and reflexive. A nearest neighbor query is defined as $NN(q) = \arg\min_{x\in\mathbb{S}} d(x, q)$, and can be solved by successive applications of *range query* $(q, r)_d = \{x \in \mathbb{S} | d(q, x) \leq r\}$ in a hierarchical index with the algorithm described in [2].

Searching for the nearest neighbor in either a multidimensional space (a.k.a. *vector space*) or a metric space with any indexing method may turn into a sequential scan of the database. This happens when the *intrinsic dimensionality* of the space is very high (about 20 dimensions for the unitary cube with uniformly distributed points) and for the coordinate-free or metric space model, when the histogram of distances is very sharp and is concentrated far from the origin. The NNS problem, the curse of dimensionality, a large number of indexes, and the problems found for *exact* nearest neighbor searching are reported in several places and surveyed in [3,4,5].

The practical impossibility of building a sublinear algorithm for NNS, and in general for exact proximity searching, has led to develop *inexact proximity searching algorithms* as surveyed in [6]. Here the algorithm returns for example ε-nearest neighbors with some closeness guarantees or just return a giving percentage of the correct nearest neighbors. Inexact indexes are faster at the expense of loosing some answers. In one form or another exact proximity searching is limited to low dimensional metric spaces. Being the focus of this paper we are interested in approximate and probabilistic algorithms.

1.2 Proximity Graphs

A *proximity graph* is a graph with the *local-imply-global* property for proximity searching. Formally $G = (V, E)$ with $V = \mathbb{S}$ the set of vertices and E the edges of the graph, G_κ will be a proximity graph if for any $x \in \mathbb{X}$ and any $v \in V$ it holds $v = NN(x)$ or there is an edge $(v, w) \in E$ such that $d(x, w) < d(x, v)$.[1] Informally, the property is: at any vertex of the graph, and for any query point $x \in \mathbb{X}$ the current node v is the nearest neighbor of x or one of the edges adjacent to v is closer to x than v.

A proximity graph defines a greedy algorithm for NNS. To find the nearest neighbor the idea is quite simple, we start in a random node and get iteratively closer to the nearest neighbor following only adjacent edges in the proximity graph, all nodes reachable from the current object are inspected and only the closer-to-the-query node is expanded in the next round. The algorithm stops when none of the neighbors of the current node is closer to the query. The number of revised objects will be proportional to the diameter of the graph times the average degree of the nodes. Please notice that extra edges in the proximity graph affect only the number of revised objects, but deliver a correct algorithm.

An example of such a proximity graph in multidimensional (vector) spaces is the Delaunay graph, the dual of the Voronoi graph surveyed in [7]. The quest for a proximity graph in a general metric space was deterred when it was proved in [1] that the degree of the graph is unbounded if only the distance information is used. The proof is essentially the following observation, for any given pair of nodes $u, v \in \mathbb{S}$ one can build a query $q \in \mathbb{X}$, not violating the triangle inequality, which needs the edge $(u, v) \notin E$ in the greedy search for the nearest neighbor; hence the graph degree is unbounded. If the degree of the nodes is unbounded then each step in the greedy algorithm will take $O(n)$ distance computations, rendering the technique useless: the curse of dimensionality again in another facet.

1.3 Approximate Greedy Searching

The inability to build a proximity graph with bounded degree in a general metric space is due to the lack of knowledge about the probable position of the query with respect to the database elements. This restrictions can be fixed for vector spaces, the Delaunay graph is a constructive proof of existence; but we are not aware of the existence of algorithmic extensions for general metric spaces.

We need a definition of the *All Nearest Neighbor Graph.* First, we need to define the KNN set.

Definition 1. *The K Nearest Neighbors search of q, $KNN(q)$ for short, comes as a natural extension to $NN(q)$ search: Sort the entire database by increasing distance to q, the first K objects in the rank are the $KNN(q)$ set.*

[1] The case of ties is special, if two objects are at the same minimal distance to the query, either one of them is a nearest neighbor.

Definition 2. *The AKNN graph is defined as* $AKNN(V, E)$ *where* $V = \mathbb{S}$ *and* $E = \{(u, v) \mid v \in KNN(u)\}$*, there will be an edge from every object towards each one of its* k*-nearest neighbors.*

This graph has been used for approximate proximity searching in at least two contexts described next. The AKNN has been used for proximity searching in [8], where authors propose the use of the graph as a distance bounding device. Each edge in the AKNN is weighted by the distance between the objects. The graph distance between two objects u and v in $\mathbb{S}$ is defined as the sum of the weights of the path joining u and v, if there is no such path the graph distance is infinite. In [8] the graph distance is used to estimate both an upper and a lower bound for the actual distance to the query using the triangle inequality. The AKNN graph is connected with high probability if k is *large enough* and the objects are distributed with a bounded away from zero distribution as discussed in [9]. The algorithm is for exact proximity searching for both nearest neighbor and range queries. The performance of the algorithm is comparable to the baseline, pivot based algorithm AESA [10], using only a fraction of the memory requirements. Nevertheless the index is useful only for low dimensional data.

Other approach, closer to ours, was presented in [11] where the authors propose an interesting alternative which consists essentially in applying the greedy algorithm in a *non* proximity graph. They propose the use of an extended version of the AKNN graph (or just AKNN were the context allow us) as an approximation to the proximity graph. The extension consists in dynamically adding edges to the AKNN for nodes τ-closer to the query q, where a node p_i is τ-closer to q with respect to p if $d(q, p) \leq \tau d(q, p_i)$ and p_i is in the connected component of p. The degree of node p depends on the value of τ, then for large values produces larger degrees. They report perfect recall for shape queries examining around 25% of the database. Even if this percentage is better than the baseline AESA, it is still not scalable. Perfect recall cannot always be obtained, for example when the nearest neighbor is in a different connected component of the graph; and they use different starting points, named *seeds* in the paper, to increase the probability of not getting stuck in a local minimum.

2 Our Contribution

From the previous work described above in [8,11] we learned that AKNN graph is useful for proximity searching, both as a distance bounding device and as an approximation to a proximity graph, since it acceps greedy nearest neighbor searching. In this paper we investigate the *proximity power* of a graph induced by the AKNN. Instead of extending a candidate graph adding nodes as in [8,11] (which increase the degree of the nodes), we propose to find a subgraph of the AKNN with the proximity property, even if the property is valid only in probability.

We begin by observing that the AKNN graph is not necessarily symmetric. If u is the k nearest neighbor of a node v, v is not necessarily the k nearest neighbor of u. There are two ways to make the AKNN graph symmetric. If a node is at the same time a nearest neighbor and a reverse nearest neighbor the two nodes realizing the relation are said to be *mutual* (k) nearest neighbors, $u \in MKNN(v) \leftrightarrow u \in KNN(v)$ and $v \in KNN(u)$, this procedure delete an edge if the symmetric does not exist. The other alternative is to symmetrize adding edges. We can also define the *symmetric* (k) nearest neighbors as $u \in SKNN(v) \leftrightarrow u \in KNN(v)$ or $v \in KNN(u)$.

2.1 Approximate Proximity Hypothesis

In the greedy setup we use adjacent nodes as routing devices for queries. Proximal nodes are certainly useful for routing and should be adjacent in the graph. We additionally postulate that two nodes sharing a certain amount (κ) of their k-nearest neighbors should be also adjacent in the proximity graph.

Definition 3. *Define $G_\kappa(\mathbb{S}, E)$ the proximal transitive graph of strength κ. With $(u, v) \in E \leftrightarrow |KNN(u) \cap KNN(v)| \geq \kappa$.*

We can also define a similar graph using $SKNN$ as the building block. We call this the *symmetric* version in the experimental results section, below.

Definition 4. *Define $\hat{G}_\kappa(\mathbb{S}, E)$ the* symmetric *proximal transitive graph of strength κ. With $(u, v) \in E \leftrightarrow |SKNN(u) \cap SKNN(v)| \geq \kappa$.*

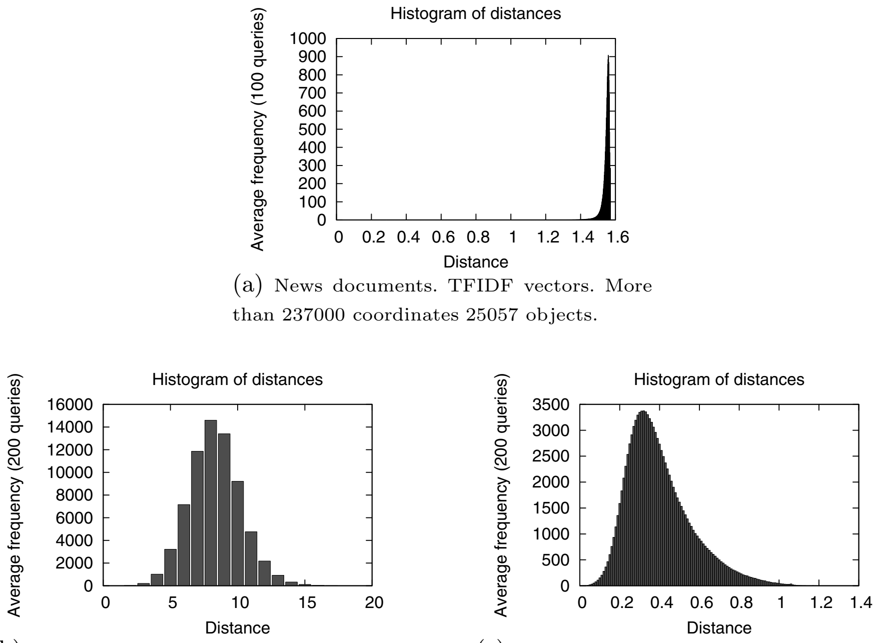
We will work on the basis of the following hypothesis: *The connected components of G_κ and $\hat{G}_\kappa$ are quasi-proximity graphs.*

There is a potentially large number of connected components in a giving graph. Since we do not know in advance in what component we can find the query we need to search in each connected component. In the next section we report on the time complexity of searching in all the connected components.

3 Experimental Results

We used a collection of 25057 TFIDF (term frequency × inverse document frequency) vectors using the angle between vectors as the metric. We used the data sets of the SISAP metric space library[2], which is a standard set of databases and objects used to compare proximity searching algorithms. The TFIDF vectors are commonly described using several thousands of coordinates. Every vector is sparse and is represented using only with a few hundred of coordinates, e.g. the average number of effective coordinates in our query set is of 360 while needs more than 237000 explicit dimensions, which is really large for standard proximity searching algorithms. The intrinsic dimensionality is also very large, as depicted in Figure 1. We can see in Figure 1(a) that the set of distances is concentrated around a large value with a small variance, according to [3] this characteristic is an indication of a large intrinsic dimension. Similarly Figures 1(c)

[2] The homepage of the SISAP project is http://www.sisap.org



(a) News documents. TFIDF vectors. More than 237000 coordinates 25057 objects.

(b) English dictionary. Edit distance. 200 queries. 69069 entries.

(c) Histogram of colors as vectors of dimension 112. Euclidean distance. 200 queries. 112682 objects.

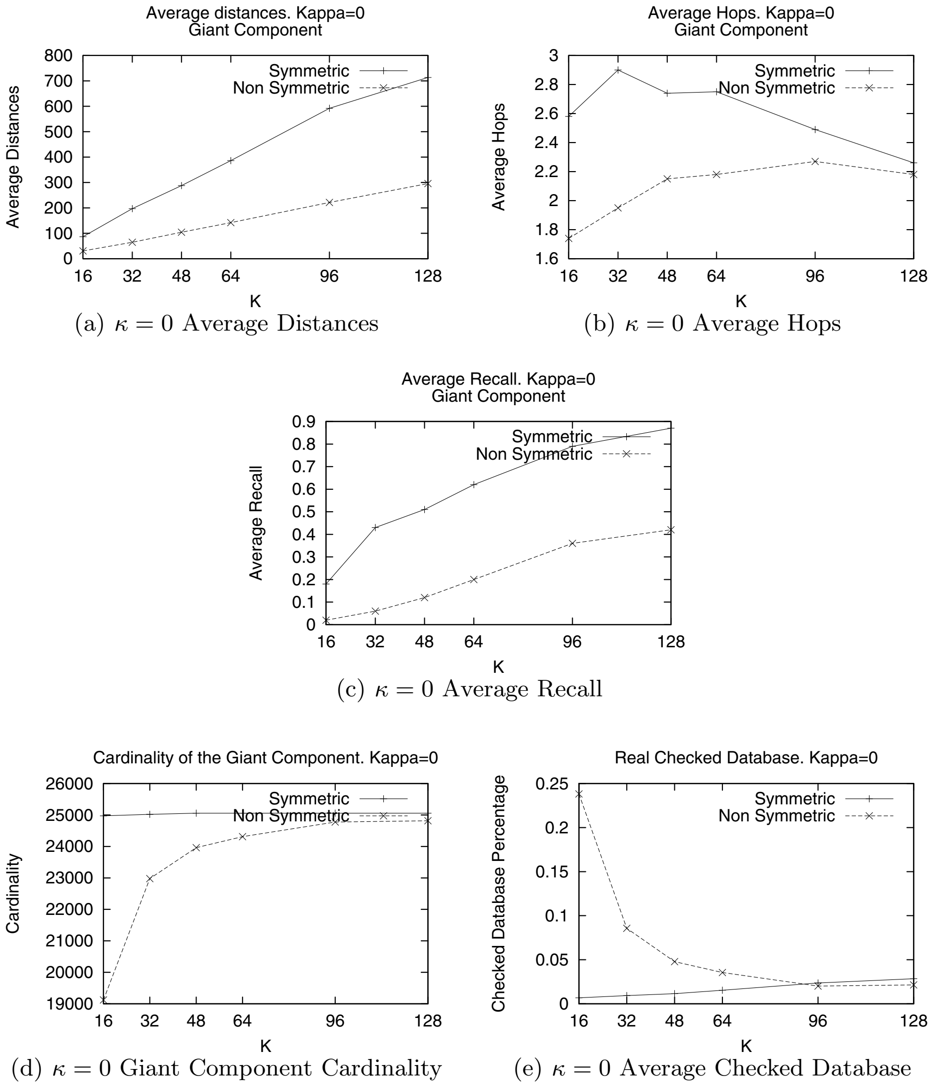
Fig. 1. Histograms of different databases. The news articles holds clearly the higher intrinsic dimensionality (large mean, small variance).

and 1(b) show the histogram of distances for vector images and an english dictionary, respectively.

Please notice that the news articles have the higher intrinsic dimension, as they have the larger mean and the smaller variance. We will focus our attention in this dataset. We have omitted the experiments for the other datasets for the space constraints of the conference format. We believe solving the problem for the hardest problem is enough evidence of the strong points of the technique. We also omitted investigating the use of $MKNN$ as the building block of the proximal transitive graph. As a quick comment, an exact index such as the Burkhard-Keller Tree (BKT) [12,3] searching the nearest neighbor needs to check more than 22% for the english dictionary 1(b) for words not in the dictionary and 9.8% for the image database with random queries selected from the same database 1(c), and 98.8% for the short news articles 1(a) (with random queries selected from the database).

A set of six indexes for $K = 16, 32, 48, 64, 96$ and 128 were built for G_κ and $\hat{G}_\kappa$. We repeated the nearest neighbor queries 300 times per index reporting the average values of the results as follows:

Recall. How many times we found the real NN, as a ratio of the 300 searches.



(a) $\kappa = 0$ Average Distances

(b) $\kappa = 0$ Average Hops

(c) $\kappa = 0$ Average Recall

(d) $\kappa = 0$ Giant Component Cardinality

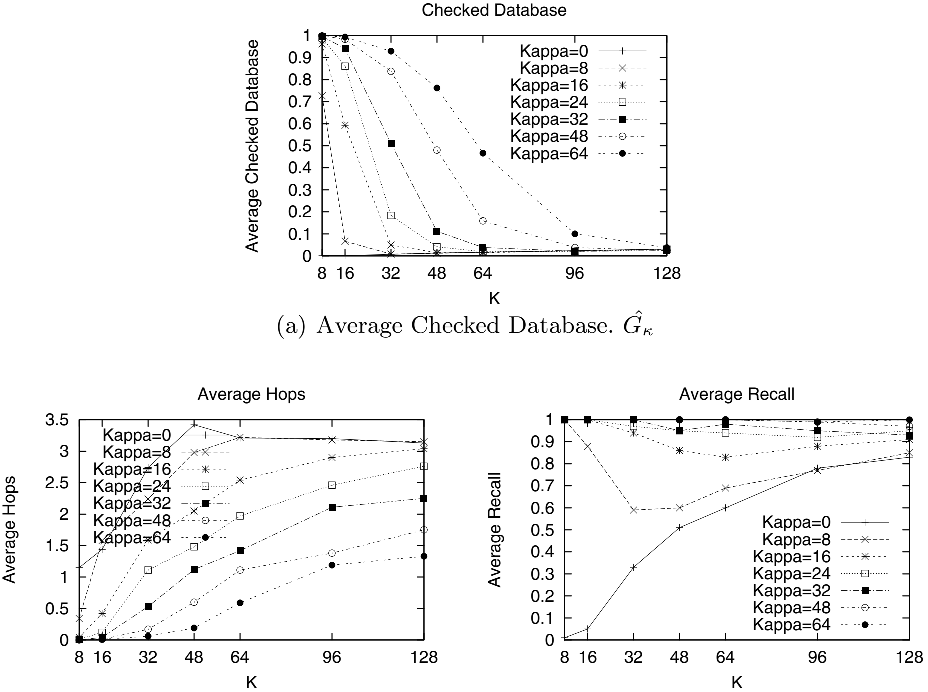
(e) $\kappa = 0$ Average Checked Database

Fig. 2. General behavior for G_0 and $\hat{G}_0$ graphs

Hops. The path inside a single connected component, measured in distances, to reach the NN.

Checked ratio. The average number of distance computations as a ratio over the total size of the database.

The special case of $\kappa = 0$ was investigated first. For the curves of $\hat{G}_\kappa$ the first observation was that a giant connected component was obtained for $\kappa = 0$,



(a) Average Checked Database. $\hat{G}_\kappa$

(b) Average Hops. $\hat{G}_\kappa$

(c) Average Recall. $\hat{G}_\kappa$

Fig. 3. Searching in queries in the connected components of $\hat{G}_\kappa$ with $\kappa \geq 0$

covering most of the graph. This makes sense because this is the least restrictive condition. Figure 2 shows the general behavior of the giant component and the index with $\kappa = 0$. The number of hops is very small, and experimentally is independent of the database size (see Figure 2(b)). The recall is very good for the giant component in the symmetric graph as depicted in 2(c). The cardinality is really close to the entire database and the leftovers can be checked sequentially or by using other indexes, this can be observed in Figure 2(d).

A notable property of the method is that for $K \geq 48$, in G_κ, we check less than 5% of the entire database even if we add the cost of the linear scan outside of the giant component. The $\hat{G}_\kappa$ needs to check an even smaller ratio of the database as shown by Figure 2(e). Due to the above fact, in the rest of the section we restrict the experiments to $\hat{G}_\kappa$ since it is superior to G_κ experimentally.

3.1 Larger κ

Figure 3 shows the result for $\kappa > 0$. Large κ values creates a large number of small CC, these behavior can be seen in 3(a) where the entire database should be checked for small K and κ. In fact, an interesting relation can be found in $\frac{\kappa}{K} \approx 1$ because it defines an inflection point affecting the performance and recall. Since the graph nodes have higher degree, this inflection point has a smaller impact than in G_κ. In general $\frac{\kappa}{K} < 1$ gives better performance, see Figure 3(a).

The recall, Figure 3(c), it is affected inversely proportional to the performance because we are avoiding the sequential scan of the database. In the other hand, the number of hops is still small and can be considered $O(1)$ as in the giant component of the previous experiment. Surprisingly, we have setups needing less than 1 hop in average to find the result. This value puts in evidence the closeness in every connected component, i.e. smaller components with strongest notions of proximity. The cost is delegated to find the correct CC which is a minimum cost for the symmetric graph, see Figure 3(b).

We have found some interesting values for some configurations, for example for $K = 62$ (i.e. 31 if we represent symmetric graphs with undirected edges) we found that for $\kappa = 24$ it fails 10 times (10 in 300 queries). The reported nearest neighbors are at a distance of 1.28. In Figure 1(a) we can see that the mean is located 1.55 and the mass is concentrated around 1.5 and 1.6. This means that 1.28 is a very (relatively) close object, unfortunately remains inexact. Another important value is the degree, which is 31. This means that we can really control the necessary space for the index using K and κ.

4 Lightweighting the Preprocessing Step

As we see the indexing process takes $O(n^2)$ comparisons to compute the $AKNN$ and hence computing either G_κ or $\hat{G_\kappa}$. Using an index to obtain AKNN is not really an option, since exact indexes will degrade to sequential search in high dimensions. The space needed to store AKNN, G_κ or $\hat{G_\kappa}$ is $O(Kn)$ and it can be handled easily even for large datasets.

Due to the high cost of the construction, the index can be used in medium or large databases, this is specially a concern for practical implementations.

We can speed up the preprocessing stage using a small modification to the algorithm. Instead of using $\hat{G}$ we can compute a graph with similar properties as follows:

- Randomly select $Y \subset S$.
- Let $m = |Y|$, $m \ll n$ in order to consider it $o(1)$ with respect to n, but large enough to accomplish the previous requirement.
- We construct in at most mn comparisons a graph of all K nearest neighbors of $u \in S$ in Y, defining $KNN_Y(u)$, and $SKNN_Y(u)$ to represent the same operations working over the restricted Y.
- In the same way, we define $\hat{G}_Y(\mathbb{S}, \kappa)$, as in definition 4, but using $SKNN_Y(\cdot)$ instead of $SKNN(\cdot)$.
- Create an inverted list I [13], from $Y \rightarrow S$, as follows: Let $u \in S$, $v \in Y$ if $v = NN(u, Y)$ then we add u to the v entry in I. This can be created in n steps with no additional distance comparisons, since we already compute the graph.
- Finally, every NN query is solved using $\hat{G}_Y$ and the greedy search algorithm. At the end, we will be placed in the neighborhood of NN at some node

$w \in S$. So, we must find in I the place for w and filter the list to get the real result.

Each posting list w inside I can be sorted in increasing distance to w, allowing to prune using the triangle inequality. Another enhancement to this linear algorithm is to increase the number of verification lists on I of the $KNN_Y(w)$ instead of just w, this should increase the possibility of find the right result.

The K parameter should be smaller than the complete algorithm because our graph holds an smaller diameter. Clearly, the above algorithm only works for symmetric graphs.

5 Conclusions and Future Work

In this paper we introduced a new approximation to a proximity graph using the notion of shared (symmetric, mutual)k-nearest neighbors. The defined graph $\hat{G}$ is divided in connected components, and the query is searched greedily in each connected component. Our algorithm solves effectively and efficiently the nearest neighbor problem using an approximate approach with high recall and checking a very small fraction of the database.

Additionally, we sketch a linear preprocessing time algorithm allowing to create a practical implementation of the method. We are currently investigating another possible enhancements using a hierarchical structure defined applying recursively different κ values.

One of the main problems is the avoidance of paths leading to local minimums, or be capable to know when a search process is stalled. In other words, converge to an exact algorithm. Many standard techniques can be used to search outside local minima.

We should notice that searching in many connected components leads to a natural parallelization technique, one thread for each connected component, this an interesting optimization exploiting capabilities of new hardware as powerful GPU's, and special networking schemes like clouds or grids.

An interesting alternative for effectively indexing large databases is to consider a mixture of the distance bounds obtained in [8] with the current work. It is not hard to see a more clever way to navigate the collection of connected components by estimating the distances between them and obtaining distance bounds to prune some candidate fractions of the database. We see many potential applications to the decomposition technique into small clusters. Our current attention is the practical implementation of the method, linear time construction keeping the recall and time characteristics of the approach, presented in this paper.

References

1. Navarro, G.: Searching in metric spaces by spatial approximation. The VLDB Journal 11(1), 28–46 (2002)
2. Samet, H.: Foundations of Multidimensional and Metric Data Structures. Morgan Kaufmann Publishers, San Francisco (2006)

3. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in metric spaces. ACM Comput. Surv. 33(3), 273–321 (2001)
4. Hjaltason, G.R., Samet, H.: Index-driven similarity search in metric spaces (survey article). ACM Trans. Database Syst. 28(4), 517–580 (2003)
5. Zezula, P., Amato, G., Dohnal, V., Batko, M.: Similarity search: The metric space approach. Springer, New York (2006)
6. Patella, M., Ciaccia, P.: Approximate similarity search: A multi-faceted problem. Journal of Discrete Algorithms 7(1), 36–48 (2009)
7. Aurenhammer, F.: Voronoi diagrams—a survey of a fundamental geometric data structure. ACM Computing Surveys (CSUR) 23(3), 405 (1991)
8. Paredes, R., Chávez, E.: Using the k-nearest neighbor graph for proximity searching in metric spaces. In: Consens, M.P., Navarro, G. (eds.) SPIRE 2005. LNCS, vol. 3772, pp. 127–138. Springer, Heidelberg (2005)
9. Brito, M., Chavez, E., Quiroz, A., Yukich, J.: Connectivity of the mutual k-nearest-neighbor graph in clustering and outlier detection. Statistics & Probability Letters 35(1), 33–42 (1997)
10. Vidal, E.: New formulation and improvements of the Nearest-Neighbour approximating and eliminating search algorithm(AESA). Pattern Recognition Letters 15(1), 1–7 (1994)
11. Sebastian, T., Kimia, B.: Metric-based shape retrieval in large databases. In: Proceedings of 16th International Conference on Pattern Recognition, vol. 3 (2002)
12. Burkhard, W.A., Keller, R.M.: Some approaches to best-match file searching. Communications of the ACM 16(4), 230–237 (1973)
13. Baeza-Yates, R.A., Ribeiro-Neto, B.A.: Modern Information Retrieval. ACM Press/Addison-Wesley (1999)

On the Design of a Hardware-Software Architecture for Acceleration of SVM's Training Phase

Lázaro Bustio-Martínez[1,2], René Cumplido[2], José Hernández-Palancar[1], and Claudia Feregrino-Uribe[2]

[1] Advanced Technologies Application Center,
7^a ♯ 21812 e/ 218 y 222, Rpto. Siboney, Playa, C.P. 12200, Havana, Cuba
{lbustio,jpalancar}@cenatav.co.cu

[2] National Institute for Astrophysics, Optics and Electronic,
Luis Enrique Erro No 1, Sta. Ma. Tonantzintla, 72840, Puebla, México
{rcumplido,cferegrino}@inaoep.mx

Abstract. Support Vector Machines (SVM) is a new family of Machine Learning techniques that have been used in many areas showing remarkable results. Since training SVM scales quadratically (or worse) according of data size, it is worth to explore novel implementation approaches to speed up the execution of this type of algorithms. In this paper, a hardware-software architecture to accelerate the SVM training phase is proposed. The algorithm selected to implement the architecture is the Sequential Minimal Optimization (SMO) algorithm, which was partitioned so a General Purpose Processor (GPP) executes operations and control flow while the coprocessor executes tasks than can be performed in parallel. Experiments demonstrate that the proposed architecture can speed up SVM training phase 178.7 times compared against a software-only implementation of this algorithm.

Keywords: SVM, SMO, FPGA, Parallel, hardware-software architectures.

1 Introduction

Support Vector Machines (SVM) is a recent technique that has been widely used in many areas showing remarkable results, specially in data classification [5]. It was developed by Vladimir Vapnik in the early 90's and created an explosion of applications and theoretical analysis that has established SVM as a powerful tool in Automatic Machine Learning and Pattern Recognition [10].

Due to SVM's training time scales quadratically (or worse) according to training database size [2], the problems that can be solved are limited. Many algorithms have been proposed to avoid this restriction, although at present there are three basic algorithms for training SVM [11]: *Chunking* [9], *Sequential Minimal Optimization* (SMO) [8] and SVM^{Light} [6] (this algorithm is an improvement

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 281–290, 2010.

to [7]). SMO has proved to be the best of them because it reduces the training time, it does not need expensive computational resources as the others, it is easily programmable and it does not require complex math libraries to solve Quadratic Programming (QP) problems that SVM involves.

SVM is inadequate for large scale data classification due to the high training times and computational resources that it requires. Because of this, is very important to explore techniques that can help to improve SVM's performance. This is the case of hardware-software architectures, especially, GPPs that can enhance their instruction set by using an attached coprocessor.

To prove the feasibility using hardware-software architectures to accelerate algorithms, a *Field Programmable Gates Arrays* (FPGA) is used as a prototyping platform. A FPGA is an integrated circuit that can be configured by the user making possible to build circuits. FPGAs are formed by logic blocks wired by reprogrammable connections, who can be configured to perform complex combinational functions (even to implement a GPP). FPGAs are used in many areas obtaining significant speed ups, such as automatic target recognition, string pattern matching, transitive closure of dynamic graphs, Boolean satisfiability, data compression and genetic algorithms [3], among others.

In this paper, SMO's performance was analyzed to identify those sections that are responsible of the processing bottleneck during its execution. To accelerate SMO, a hardware-software architecture was designed and implemented. In this architecture, hardware executes the most time-consuming functions while the software executes control flow and iterative operations.

This paper is organized as follows: in Section 2 describes the different approaches to implement processing algorithms, including a brief description of the FPGAs. In Section 3, the SVM and their theoretical foundation are revised as well as the most cited algorithms that train SVM are described, explaining their characteristics and particularities, specially for the SMO algorithm. In Section 4 the architecture proposed is described, detailing software and hardware implementations while in Section 5 the results are shown. The work is concluded in Section 6.

2 Platforms for Algorithms Implementation

There are two main approaches to implement algorithms. The first one consists in building *Application Specific Integrated Circuits* (ASICs)[3]. They are designed and built specifically to perform a given task, and thus they are very fast and efficient. ASICs can not been modified after fabrication process and this is their main disadvantage. If an improvement is needed, the circuit must be re-designed and re-builded, incurring in the costs that this entails.

The second one consists in using a GPP which is programmed by software; it executes the set of instructions that are needed by an algorithm. Changing the software instructions implies a change in the application's behavior. This results in a high flexibility but the performance will be degraded. To accomplish certain function, the GPP, first must read from memory the instructions to be executed and then decode their meaning into native GPP instructions to determine which actions must be done. Translating the original instructions of an algorithm introduces a certain delay.

The hardware-software architectures combines the advantages of those two approaches. It aims to fills the gap between hardware and software, achieving potentially much higher performance than software, while maintaining a higher level of flexibility than hardware.

In classification tasks, many algorithms are expensive in terms of processing time when they are implemented in GPP and they classifies large scale data. When a classification algorithm is implemented, it is necessary to perform a high amount of mathematical operations that can not be done without the flexibility that software provides. So, a hardware-software architectures offer an appropriate alternative to implement this type of algorithms.

FPGAs appeared in 1984 as successors of the *Complex Programmable Logic Devices* (CPLDs). The architecture of a FPGAs is based on a large number of logic blocks which performs basic logic functions. Because of this, an FPGA can implement from a simple logical gate, to a complex mathematical function. FPGAs can be reprogrammed, that is, the circuits can be "erased" and then, a new algorithm can be implemented. This capability of the FPGAs allow us to create fully customized architectures, reducing cost and technological risks that are present in traditional circuits design.

3 SVM for Data Classification

SVM is a set of techniques based on convex quadratic programming for data classification and regression. The main goal of SVM is to separate training data into two different groups using a decision function (separating hyperplane) which is obtained from training data. The separating hiperplane can be seen, in its simplest way, as a line in the plane whose form is $y = \boldsymbol{w} \cdot \boldsymbol{x} + b$ or $\boldsymbol{w} \cdot \boldsymbol{x} - b = 0$ for the canonical hyperplane. SVM classification (in a simple two-class problem) simply looks at the sign of a decision function for an unknown data sample.

Training a SVM, in the most general case, is about to find those λ's that maximizes the Lagrangian formulation for the dual problem L_D according to the following equation:

$$L_D = \sum_{i=1}^{l} \lambda_i - \frac{1}{2} \sum_{i,j=1}^{l} y_i y_j K\left(\mathbf{x_i} \cdot \mathbf{x_j}\right) \lambda_i \lambda_j \tag{1}$$

subject to:

$$\sum_{i=1}^{l} y_i \lambda_i = 0; 0 \leqslant \lambda_i \leqslant C, i = 1, 2, ..., l \tag{2}$$

where $K(\mathbf{x_i} \cdot \mathbf{x})$ is a positive definite kernel that maps input data into a high dimension feature space where linear separation becomes more feasible [12]. $\mathbf{x_i}, \mathbf{x_j} \in R^d$ are the input vectors of the i^{th} and j^{th} training data respectively, l is the number of training samples; $y \in \{-1; 1\}$ is the class label; $\lambda = \lambda_1, \lambda_2 ... \lambda_n$ are the Lagrange multipliers for the training dataset in the Lagrangian formulation. So,the unknown data can be classified using $y = sign \left(\sum_{i=1}^{l} y_i \lambda_i K \left(\mathbf{x_i} \cdot \mathbf{x} \right) - b \right)$ where b is the SVM's threshold and is obtained using $\lambda_i \left(y_i \left(\mathbf{w} \cdot \mathbf{x_i} - b \right) - 1 \right) = 0, i = 1, 2, ..., l$ for those data samples with $\lambda_i > 0$ (those data samples are called *Support Vectors*).

The kernel function depends on the user's choice, and the resultant feature space determines the functional form of the support vectors; thus, different kernels behave differently. Some common kernels can be found on [7]. Many of the kernel functions are formed by Linear Kernel, except RBF one. Mathematically, to accelerate the Linear Kernel implies to accelerate the others. Because of this, the Linear Kernel is focused in this paper.

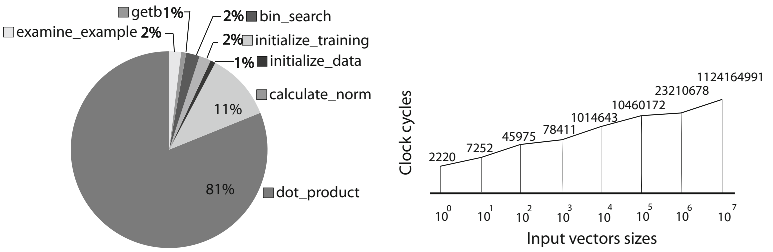
4 Architectural Design

SMO is basically a sequential algorithm: heuristic hierarchy is formed by a set of conditional evaluations which decides the algorithm behavior, with every evaluation depending on the result of the previous evaluation. Because of this sequentiality, SMO can not be implemented as it is in hardware. In addition, the highly time-consuming functions are fully parallelizable, as it is the case of kernel function computation. Thus, a hardware-software architecture that implements in hardware the most time-consuming functions and heuristic hierarchy in software could be the right approach for reducing execution time in SVM training.

4.1 SMO's Performance Profiling

There are a few SMO's performance analyses in the literature. Only Dey et al. in [4] analyze SMO's performance and identify the most time-consuming functions. In their paper, Dey et al. demonstrate the convenience of using hardware-software architectures to speed up algorithms and use SVM as an example to prove this approach. In order to identify task level hot spots in SMO's execution and to validate Dey's results, a performance profiling was made. The results are shown in Fig. 1(a).

It was observed that 77% of the total calls in SMO corresponds to the *dot_product* function. The time profile analysis shows that 81% of the total execution time was spent by the *dot_product* function. As a result of performance analysis it is evident that the *dot_product* function is responsible of bottleneck in SMO's execution. Fig. 1(b) supports this conclusion. From the performance analysis we concluded that using a hardware-software architecture to implement SMO algorithm, where software implements heuristic hierarchy, and hardware implements the *dot_product* function could obtain an speed up of at least one order of magnitude when is compared to software implementations.



(a) Performance profile for SMO.

(b) Performance profile for dot product calculation in software.

Fig. 1. Performance profile analysis

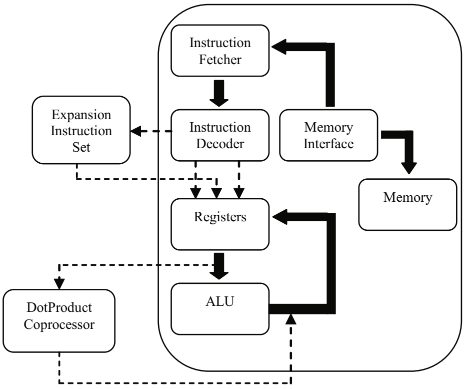


Fig. 2. Diagram of the proposed architecture

4.2 Architecture Description

Fig. 2 shows a diagram of the proposed architecture. The architecture is formed by a GPP that can enhance its performance by using a coprocessor, where the control structures are executed in software on the GPP and the dot product computations are executed on the coprocessor. The software reads the training file, initializes the data structures and receives the parameters for the SMO. Thus, when training starts, the software executes control mechanisms and the coprocessor executes high time-consuming functions.

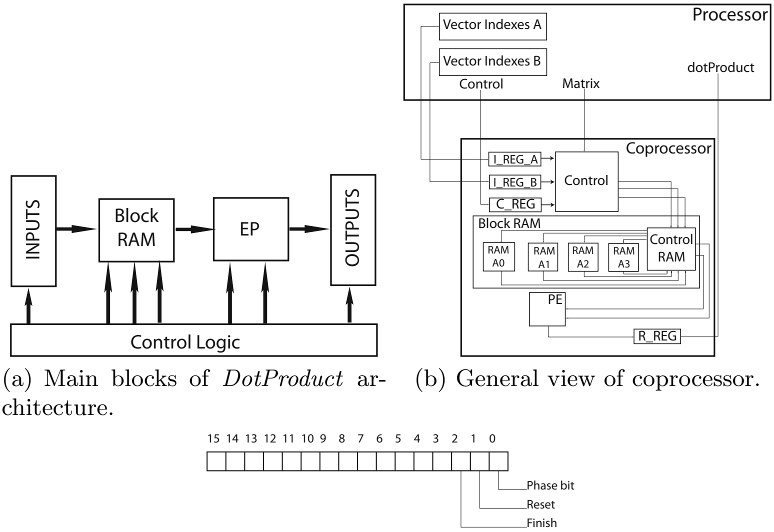
4.3 Software Implementation

To accomplish the proposed architecture, the software implementation must first load a training file and algorithm parameters. After that, the application executes the SMO algorithm and selects the correct branch from the heuristic hierarchy that SMO implements. When a dot product is needed, the application indicates the vectors that will be sent to the coprocessor. When the computation is finished, the application obtains the resulting dot product from the coprocessor, generates the output file with the training results and finishes the training process.

4.4 Hardware Implementation

The hardware architecture for the *dot_product* calculation will be named *DotProduct*, while SMO with the *dot_product* function implemented in hardware will be named *FSMO*. For this architecture, the training dataset will be represented as a matrix without using any compression method and requires that values of the matrix to be 1 or 0. Since the dot product is calculated many times and the values for this calculation remains constant, the right strategy to avoid unwanted delays is to map the training dataset inside the coprocessor. The dot product is $dotProduct = \sum_{i,j=1}^{l} \mathbf{x_i} \cdot \mathbf{x_j}$ where $\mathbf{x_i}$ and $\mathbf{x_j}$ are training vectors and l is the number of elements on vectors. The digital architecture that implements this mathematical expression consists of 5 main blocks as shown in Fig. 3(a).

INPUTS represents control signals, registers and data necessary for the architecture to work. BLOCK RAM is a memory block that contains the training dataset. Each row corresponds to one training data sample. The Processor Element (PE) is the basic computation unit which calculates the dot products of two input vectors. OUTPUT is the element that addresses the dot product computation results, and CONTROL LOGIC are those elements that permit to control and data flow inside the architecture.



(a) Main blocks of *DotProduct* architecture.

(b) General view of coprocessor.

(c) Structure of C-REG.

Fig. 3. Description of the proposed architecture

Through INPUTS, the *DotProduct* architecture obtains the indexes that will be used on the dot product calculation. INPUTS is used for mapping training data into BLOCK RAM. At this point, all data necessary to calculate a dot product of input vectors are inside the *DotProduct* architecture. Those two vectors whose indexes were given through INPUTS are delivered to the PE where the dot product is calculated and then, the result is stored in OUTPUTS. A general view of architecture is shown in Fig. 3(b).

There are two registers, *I_REG_A* and *I_REG_B*, which hold indexes of training data samples that will calculate the dot product. Register *C_REG* controls when to load data, when to read from BLOCK RAM or when to start a dot product calculation. *C_REG* is shown in Fig. 3(c). The *Phase* bit states whether the architecture is in the Initialization and Load Data Phase (set to 0) or in the Processing Phase (set to 1). When *Reset* is active (set to 1), all registers are initialized to 0, Initialization and Load Data Phase are enabled and the PE is ready to process new data. The *Finish* bit indicates when processing is finished and it is active at 1.

When *FSMO* starts, the Initialization and Load Data Phases are activated (the *Phase* bit of *C_REG* is set to 0). After this, register *I_REG_B* is disabled and the address bus of BLOCK RAM is connected to *I_REG_A* indicating the address where the value stored in matrix will be written (see Fig. 3(b) for more details) ensuring data transfer from training dataset into BLOCK RAM. When BLOCK RAM is filled, the architecture stays at this state while the *Phase* bit of *C_REG* is 0. When the *Phase* bit is changed to 1, matrix input is disabled and *I_REG_B* is enabled and connected to BLOCK RAM. At this moment, the training data samples whose indexes are stored in *I_REG_A* and *I_REG_B* are delivered to the PE where the dot product is calculated. The result of the dot product computation is stored in *R_REG*, *Finish* bit is activated and the architecture is ready to calculate a new dot product.

The PE calculates the dot product of two given training data samples. For this training dataset representation, the dot product computation is reduced to apply a logical AND operation between input vectors and counts the number of 1's in resulting vector. In this way, the architecture that implements the PE is shown in Fig. 4. Notice that the PE can calculate a dot product using three clock cycles; so, the processing time for the dot product calculation is: $t = 3 \cdot v$ where v is the number of dot products. To prove the validity of the architecture proposed, the *DotProduct* architecture was implemented using VHDL language over ISE 9.2 Xilinx suite, and was simulated using ModelSIM SE 6.5. Hardware architecture was implemented on an XtremeDSP Virtex IV Development Kit card. The software application was written using Visual C++ 6.0 and ANSI C.

Based on the fact that the Linear Kernel are used by many others, the *Dot_Product* architecture is suitable to perform others kernel functions. Using the *Dot_Product* architecture as starting point, any of most used kernel are obtained just adding some blocks that implement the rest of their mathematical formulation.

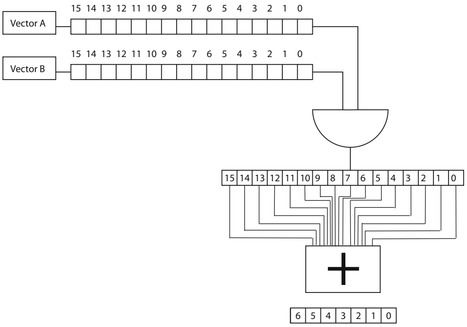


Fig. 4. Hardware implementation of *DotProduct* architecture

5 Experiments and Results

Since the dot product is the responsible of the bottleneck in SMO execution, a performance profile for this function was made. Eight experiments were carried out using a Pentium IV processor running at 3GHz and the results are shown in Fig. 1(b). The number of clock cycles required grows with the size of the input vectors.

In hardware, the dot product calculation is independent of input vector size. The *DotProduct* architecture can handle input vectors of 128-bits wide in 3 clock cycles: 1) receives data samples indexes, 2) fetches data sample vectors and 3) calculates the dot product. If the dot product calculation in software of two input vectors of 128-bits wide is compared with hardware implementation, the second one will be completed at 3 clock cycles while the first one will be completed between 45957 and 78411 clock cycles.

5.1 Experiments on Adult Dataset

Adult dataset [1] was used by Platt in [8] to prove the feasibility of SMO, and the same dataset was used here to prove the feasibility of proposed architecture. *Adult* dataset consists of 9 corpuses which contain between 1605 and 32562 data samples of 123 characteristics each one. *DotProduct* can manage training datasets of 4096 training data samples of 128 characteristics because of area limitations of the chosen FPGA. Only *Adult-1*, *Adult-2* and *Adult-3* have sizes that can be handled by the *DotProduct* architecture and the results of training those datasets are shown in table 1. In those tables, *C.C.* means *Clock Cycles.*

Table 2 shows the results for Platt's SMO. There is a deviation in threshold b for this implementations when is compared to *FSMO*. Platt in [8] does not present any implementation detail so it is not possible explain exactly the reason of this deviation: the *epsilon* value of the PC could be responsible for that behavior. Table 3 shows that in the worst case, the deviation incurred is less than 0.5% when is compared to Platt's SMO. So, the proposed architecture trains correctly the SVM.

Table 1. Experimental results of training *Adult* with *FSMO*

Corpus	Objs.	Iter.	Training Time		b	Non Bound	Bound
Adult			sec.	C.C.(10^{12})		Support Vectors	Support Vectors
1	1605	3474	364	1.089	0.887	48	631
2	2265	4968	746	2.232	1.129	50	929
3	3185	5850	1218	3.628	1.178	58	1212

Table 2. Experimental results of *Adult's* training with Platt's SMO

Corpus	Objs.	Iter.	Time	b	Non Bound	Bound
Adult			sec		Support Vectors	Support Vectors
1	1605	3474	0.4	0.884	42	633
2	2265	4968	0.9	1.127	47	930
3	3185	5850	1.8	1.173	57	1210

Table 3. Deviation in *Adult* training for *FSMO* and Platt's SMO

Corpus	Threshold b		Dif.	%
Adult	*FSMO*	SMO(Platt)		
1	0.887279	0.88449	0.0027	0.257
2	1.129381	1.12781	0.0015	0.139
3	1.178716	1.17302	0.0056	0.483

5.2 Analysis of Results

In this paper the hardware architecture to speed up the dot product computation was implemented taking advantage of parallel capabilities of hardware. Also, the heuristic hierarchy of SMO was implemented in software and it uses the hardware architecture for the dot product calculations. *FSMO* trains correctly a SVM, and it accuracy is over 99% compared to Platt's implementation [8].

After the synthesis of the *DotProduct* architecture, it was determined that this architecture can run at 35 MHz of maximum frequency. Since the dot product in hardware takes three clock cycles is then the *DotProduct* architecture could calculate 11666666 dot products of 128-bits wide input vectors in a second. Meanwhile, the same operation for input vectors of 128-bits wide using a Pentium IV processor running at 3GHz of frequency requires 45957 clock cycles, so in this processor, we can calculate 65278 dot products in a second. This demonstrates that the *DotProduct* architecture can run up to 178.7 times faster than its implementation in a modern GPP. The *DotProduct* architecture requires 33% of the available reprogrammable area, thus we can extend it to handle training datasets three times bigger. Larger training datasets can be handled if external memories are used, in this case the architecture can be extended 10 more times.

6 Conclusions

In this paper we proposed a hardware-software architecture to speed up SVM training. SMO algorithm was selected to be implemented in our architecture. SMO uses a heuristic hierarchy to select two candidates to be optimized. The dot product calculation in SMO spent 81% of the total execution time so this function was implemented in hardware while heuristic hierarchy was implemented in software, on the GPP. To validate the proposed architecture we used an XtremeDSP Virtex IV Development Kit card as coprocessor obtaining a speed up of 178.7x for the dot product computations when compared against a software-only implementation running on a GPP.

References

1. Newman, D.J., Asuncion, A.: UCI machine learning repository (2007)
2. Burges, Christopher, J.C.: A tutorial on support vector machines for pattern recognition. Data Min. Knowl. Discov. 2(2), 121–167 (1998)
3. Compton, K., Hauck, S.: Reconfigurable computing: a survey of systems and software. ACM Comput. Surv. 34(2), 171–210 (2002)
4. Dey, S., Kedia, M., Agarwal, N., Basu, A.: Embedded support vector machine: Architectural enhancements and evaluation. In: VLSID '07: Proceedings of the 20th International Conference on VLSI Design Held Jointly with 6th International Conference, Washington, DC, USA, pp. 685–690. IEEE Computer Society, Los Alamitos (2007)
5. Guyon, I.: Svm application list (2006)
6. Joachims, T.: Making large-scale support vector machine learning practical. pp. 169–184 (1999)
7. Osuna, E., Freund, R., Girosi, F.: An improved training algorithm for support vector machines. In: Proceedings of the 1997 IEEE Workshop on Neural Networks for Signal Processing, vol. VII, pp. 276–285 (1997)
8. Platt, J.C.: Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report, Microsoft Research, MST-TR-98-14 (1998)
9. Vapnik, V., Kotz, S.: Estimation of Dependences Based on Empirical Data: Empirical Inference Science (Information Science and Statistics). Springer, New York (2006)
10. Vapnik, V.N.: The nature of statistical learning theory. Springer, New York (1995)
11. Wang, G.: A survey on training algorithms for support vector machine classifiers. In: NCM '08: Proceedings of the 2008 Fourth International Conference on Networked Computing and Advanced Information Management, Washington, DC, USA, pp. 123–128. IEEE Computer Society, Los Alamitos (2008)
12. Weisstein, E.W.: Riemann-lebesgue lemma (online)

A Highly Parallel Algorithm for Frequent Itemset Mining

Alejandro Mesa[1,2], Claudia Feregrino-Uribe[2], René Cumplido[2],
and José Hernández-Palancar[1]

[1] Advanced Technologies Application Center, CENATAV. La Habana, Cuba
{amesa,jpalancar}@cenatav.co.cu
[2] National Institute for Astrophysics, Optics and Electronics,
INAOE. Puebla, México
{amesa,cferegrino,rcumplido}@inaoep.mx

Abstract. Mining frequent itemsets in large databases is a widely used technique in Data Mining. Several sequential and parallel algorithms have been developed, although, when dealing with high data volumes, the execution of those algorithms takes more time and resources than expected. Because of this, finding alternatives to speed up the execution time of those algorithms is an active topic of research. Previous attempts of acceleration using custom architectures have been limited because of the nature of the algorithms that have been conceived sequentially and do not exploit the intrinsic parallelism that the hardware provides. The innovation in this paper is a highly parallel algorithm that utilizes a vertical bit vector (VBV) data layout and its feasibility for making support counting. Our results show that for dense databases a custom architecture for this algorithm can perform faster than the fastest architecture reported in previous works by one order of magnitude.

1 Introduction

Nowadays, many data mining techniques have emerged to extract useful knowledge from large amounts of data. Finding correlations between items, specifically frequent itemsets, is a widely used technic in data mining. The algorithms that have been developed in this area require powerful computational resources and a lot of time to solve the combinatorial explosion of itemsets that can be found in a dataset. The high computational resources required to process large databases can render the implementation of this kind of algorithms impractical. This is mainly due to the presence of thousands of different items or the use of a very low threshold of support (minsup[1]).

Attempts to accelerate the execution of algorithms for mining frequent itemsets have been reported. The most common practice in this area is the use of parallel algorithms such as CD [1], DD [1], CDD [1], IDD [5], HD [5], Eclat [12]

[1] Is the minimum number of times in a database an itemset must occur to be considered as frequent.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 291–300, 2010.

and ParCBMine [7]. However, all these efforts have not reported good execution times given a reasonable amount of resources for some practical applications. Recently, hardware architectures have been used in order to speed up the execution time of those algorithms. These architectures were presented in [2,3,11,10,9], and improved the execution time of software implementations by some orders of magnitude.

Previously proposed parallel algorithms used a coarse granularity parallelism. Generally a partition of the database is made in order to process in a parallel fashion each block of data. In this paper an algorithm that exploits the inherent task parallelism of hardware implementation and the feasibility to perform bitwise operations is proposed. A hardware architecture for mining frequent itemsets is developed to test the efficiency of the proposed algorithm.

The remainder of this paper is organized as follows. Section 2 describes related work. Section 3 discusses the proposed algorithm. Section 4 describes the systolic tree architecture that supports the algorithm. The results are discussed in Section 5 and Section 6 presents the conclusions.

2 Related Work

Algorithms that use data parallelism deal with load balancing, costs of communication and synchronization. These are problems not commonly present in algorithms that exploit task parallelism. This is because in this type of algorithms the efficiency is based on the high speed that can be achieved by a simple task executed in a multiprocess environment. Therefore, normally the data are accessed sequentially. Currently, hardware architectures for three frequent itemsets mining algorithms (Apriori, DHP and FP-Growth) have been proposed in the literature [2,3,11,10], where sequential algorithms are used.

A systolic array architecture for the Apriori algorithm was proposed in [2] and [3]. A systolic array is an array of processing units that processes data in a pipelined fashion. This algorithm generates potentially frequent itemset (candidates itemsets) following a heuristic approach. Then, it prunes the candidates known to be infrequent, and finally it counts the support of the remaining itemsets. This is done for each size of itemset until there is no frequent itemset left. In the proposed architectures, each item of the candidate set and the database are injected to the systolic array. In every unit of the array, the subset operation and the candidate generation is performed in a pipelined fashion. The entire database has to be injected several times through the array. In [2], the hardware approach provides a minimum of a 4× time performance advantage over the fastest software implementation running on a single computer. In [3], an enhancement to the architecture is explored, introducing a bitmaped CAM for parallel counting the support of several itemsets at once, achieving a 24× time performance speedup.

In [11] the authors use as a starting point the architecture proposed in [3] to implement the DHP algorithm [8]. This introduces two blocks at the end of the systolic array to collect useful information about transactions and candidate itemsets to perform a trimming operation over them and to reduce the amount of data that the architecture has to process.

In [10] a systolic tree architecture has been proposed to implement the FP-Growth algorithm [6]. This algorithm uses a compact representation of the data in an FP-Tree data structure. To construct the FP-Tree, two passes through the database are required and the remaining processing is made through this data structure. This tree architecture consists of an array of processing elements, arranged in a multidimensional tree to implement the FP-Tree. With this architecture a reduction of data workload is achieved.

All presented works in this section use a horizontal items-list data layout. This representation requires a number of bits per item to identify them (usually 32). Also, all architectures implement algorithms that have been conceived for software environments and do not take full advantage of the hardware intrinsic parallelism.

3 The Proposed Algorithm

Conceptually, a dataset can be defined as a two-dimensional matrix, where the columns represent the elements of the dataset and the rows represent the transactions. A transaction is a set of one or more items obtained from the domain in which the data is collected. Considering a lexicographical order over the items they can be grouped by equivalence classes. The equivalence class ($E(X)$) of an itemset X is given by:

$$E(X) = \{Y | \mathit{Prefix}_{k-1}(Y) = X \wedge |Y| = k\} \tag{1}$$

The proposed algorithm is based on a search over the solution space through the equivalence class, considering a lexicographical order over the items. This is a two-dimensional search, both breadth and depth is performed concurrently. Using the search through the equivalence class allows us to exploit a characteristic of VBV data layout. With this type of representation, the support of an itemset can be defined as the number of active bits of the vector that represents it ($\overline{X}$). This vector $\overline{X}$ represents the co-occurrence of items in the database and can be obtained as a consecutive bitwise *and* operation between all the vectors that represent each item of the itemset(see equation 2).

$$\begin{aligned} X &= \{a, b, c, \ldots, n\} \\ \overline{X}_n &= \overline{a} \text{ and } \overline{b} \text{ and } \overline{c} \text{ and } \ldots \text{ and } \overline{n} \end{aligned} \tag{2}$$

The process to obtain $\overline{X}$ can be done by steps. First, an *and* operation can be performed between the first two vectors ($\overline{a}$ and $\overline{b}$) and then the results are accumulated. This accumulated value can be used to perform another *and* operation with the third vector to obtain $\overline{X}$ for $X = \{a, b, c\}$. To obtain $\overline{X}_n$ this must be done for all the items in X. This procedure is shown in Algorithm 1.

Defining the search space as a tree (Figure 1 shows a 5 items search space) in which each node represents an item of the database, an itemset is determined by

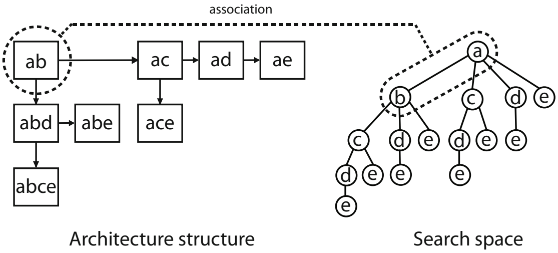
Algorithm 1 $\overline{X}_n$ calculation

```
Initialize vector $\overline{X}_{accum}$ with 1's
for all $items\ i\ in\ X$ do
    $\overline{X}_{accum} = \overline{X}_{accum}$ and $\overline{i}$
end for
$\overline{X}_n = \overline{X}_{accum}$
```

Algorithm 2 Data interchange

```
for all $sections\ S\ in\ DB$ do
    for all $data\_vector\ in\ S$ do
        $Process\_Data(data\_vector);$
    end for
end for
$Process\_Minsup(min\_sup\_value);$
$Result := Get\_Data();$
```

the shortest path between two nodes of the tree. Using the procedure previously described, the vector $\overline{X}_n$ of an itemset can be obtained recursively as $\overline{X}_n = \overline{X}_{n-1}$ and $\overline{n}$, being $\overline{X}_{n-1}$ the accumulated value in the parent node. Once $\overline{X}_{n-1}$ is calculated, it can be used to obtain all the accumulated vectors on each child node. With this process, each node provides partial results for each one of the itemsets that includes it in the path of the tree.



Architecture structure

Search space

Fig. 1. Structure for processing 5 item solution tree

In this algorithm there is no candidate generation, but the search space is explored until reaching a node for which the support is zero. This process is sustained by the downward closure property, which establishes that the support of any itemset is greater than or equal to the support of any of its supersets. Because of this, if this node does not have active ones in the accumulated vector, the nodes in the lower subtree will not receive any contribution in the support value from that vector.

The lexicographical order of the items is established according to their frequency, ordering first the ones with the smallest values. This order causes the value of the support of itemsets to decrease rapidly when descending through the tree architecture.

Managing large databases with VBV data layout can be expensive because the size of $\overline{X}_n$ is determined by the number of transactions of the database. To solve this, a horizontal partition of the database is made. Each section is processed independently, as shown in Algorithm 2, and each node accumulates the itemsets section support. Every time a node calculates a partial support, it is added to the accumulated support of the itemset. When the database processing finishes, each node has the global support of all the itemsets that it calculated.

A structure of processing elements is needed to implement the algorithm. This structure interchanges data as described in Algorithm 2. Two modes of data injection and one for data extraction are needed to achieve the task of frequent itemsets mining: "Data In", "Support Threshold" and "Data Flush". In the first mode ($Process\_Data()$) all data vectors are fed into the structure by the root element to process the itemsets. The second mode ($Process\_Minsup()$) injects the support threshold so the processing elements can determine which itemset is frequent and which is not. For the data extraction $Get\_Data()$, all the elements of the structure flush the frequent itemsets through their parent and the data exits the structure through the root element.

To implement the algorithm we use a binary tree structure of processing elements (PE). Figure 1 shows a five-item solution tree processing structure. The structure is a systolic tree and it was chosen because this type of construction allows to exponentially increase the concurrent operations at each processing step. For this, the number of concurrent operations can be calculated as $1 + \sum_{i=0}^{cc} 2^i$, being cc the number of processing steps that have elapsed since the process started.

Each node of the systolic tree (n_{arq}) is associated to a node of the solution tree (n_{sol}). This n_{arq} determines an itemset (IS_n) and it is formed with the path from the tree root to n_{sol}. The n_{arq} calculates the support of the supersets of IS_n. Those supersets are the ones that are determined by all the nodes in the path from the root to a leaf, following the node that is most to the left of the solution tree.

A PE has a connection from its parent and it is connected to a lower PE (PEu, with an upper entry) and to a right PE (PEl, with a left entry). In general, the amount of PEs a structure has, can be calculated as follows:

$$ST_n = 1 + PEu_{n-1} + PEl_{n-1}, \text{ for } n \geq 2,$$

with PEu_n and PEl_n:

$$\begin{array}{l|l} PEu_2 = 0, & PEl_1 = 0, \\ PEu_3 = 1, & PEl_2 = 1, \\ PEu_n = 1 + PEl_{n-2} + PEu_{n-1} & PEl_n = 1 + PEl_{n-1} + PEu_{n-1} \end{array}$$

3.1 Processing Elements

The itemsets that a PE calculates are determined by the data vectors that the parent feeds to it. This must be in such a way that the first vector to reach each PE is the $\overline{X}_n$ of the prefix of the equivalence class that the IS_n belongs to. Each PE calculates the support of the vector that receives from the parent, it accumulates the bitwise *and* operation in a local memory and propagates the data vector that it receives from the parent or the accumulated value correspondingly. To achieve this, the behavior of the two types of PEs is defined. In "Data In" mode, the main difference between PEl (described in Algorithm 3) and PEu

```
Algorithm 3 PEl in “Data In” mode
  if ($is\_first(data\_vector)$) then
    $and\_tmp = data\_vector$
    $Right\_vector = data\_vector$
  else if ($is\_second(data\_vector)$) then
    $and\_tmp = and\_tmp$ and $data\_vector$
    $Down\_vector = and\_tmp$
    $mem[i] = calc\_sup(and\_tmp) + mem[i]$
  else
    $and\_tmp = and\_tmp$ and $data\_vector$
    $Down\_vector = data\_vector$
    $Right\_vector = data\_vector$
    $mem[i] = calc\_sup(and\_tmp) + mem[i]$
  end if
```

```
Algorithm 4 PEu in ”Data In” mode
  if ($is\_first(data\_vector)$) then
    $and\_tmp = data\_vector$
    $Right\_vector = data\_vector$
  else if ($is\_second(data\_vector)$) then
    $Down\_vector = and\_tmp$ and $data\_vector$
  else if ($is\_third(data\_vector)$) then
    $and\_tmp = and\_tmp$ and $data\_vector$
    $Down\_vector = data\_vector$
    $mem[i] = calc\_sup(and\_tmp) + mem[i]$
  else
    $and\_tmp = and\_tmp$ and $data\_vector$
    $Down\_vector = data\_vector$
    $Right\_vector = data\_vector$
    $mem[i] = calc\_sup(and\_tmp) + mem[i]$
  end if
```

(described in Algorithm 4) is that *PEu* does not accumulate the result of the bitwise *and* operation of the second data vector that it receives. This operation provides the down *PE* the prefix with the equivalence class of its IS_n.

3.2 Feedback Strategy

As the number of *PEs* of the tree is directly dependent on the number of frequent items that exist in the database, it is impractical to create this structure for databases with many frequent items. To solve this problem a structure for n items can be designed and taking into account the recursive definition of trees we could calculate the itemsets stepwise, see Figure 2. In the first step all the itemsets that can be calculated with this structure are obtained. Since it is known which level of the solution tree is processed for a given structure (Figure 2 shows it as the architecture processing border), data are injected back into this structure except that this time the first vector will not be the first item, but the prefix of the equivalence class of the itemset that is determined by the border tree node that was not processed.

In Figure 2, an example of a six-items search space is shown. The dotted line subtrees are examples of subtrees with different parents that have to be processed with the feedback strategy. In the example, to process the first subtree to the left, the first vector that has to be injected to the architecture is the vector that represents the itemset $X = \{a, b, c\}$.

This feedback is repeated for each solution tree node that is in the border of the nodes that were not processed. As each of these nodes defines a solution subtree (all the subtrees that are below the processing border in Figure 2), this structure is compatible with the entire tree and consequently can process a solution tree of any size. This process is defined recursively for the entire solution tree and as a result, we obtain a partition of the tree and each subtree of the partition represents a solution tree of n items.

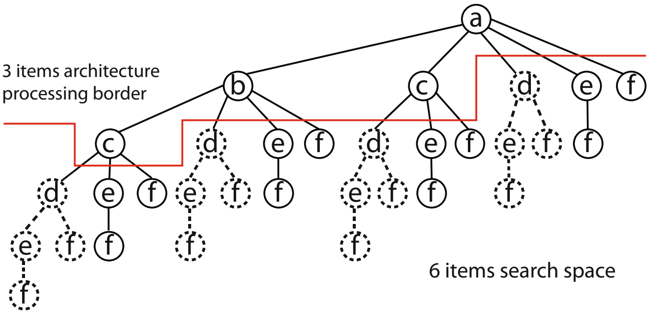


Fig. 2. Feedback strategy

4 Architecture Structure

The proposed architecture has three modes of operation: “Data In”, “Support Threshold” and “Data Flush”. In the “Data In” mode, all $\overline{X}_n$ of each section of the database are injected. When the entire database is injected, the architecture enters in the “Support Threshold” mode and the support threshold value is provided and propagated through the systolic tree. Once a node receives the support threshold value, all the *PEs* change to “Data Flush” mode and the results are extracted from the architecture.

In this architecture, the amount of clock cycles for the entire process can be divided into two general stages: Data In (CC_{data}) and Data Flush (CC_{flush}). The CC_{data} is mainly defined by the number of frequent items in the database to specific support threshold (nf), the number of transactions of the database (T) and the size of the data vector that is chosen (bw). The data vector defines the number of sections (S) in the partition of the database. CC_{data} can be calculated as follows:

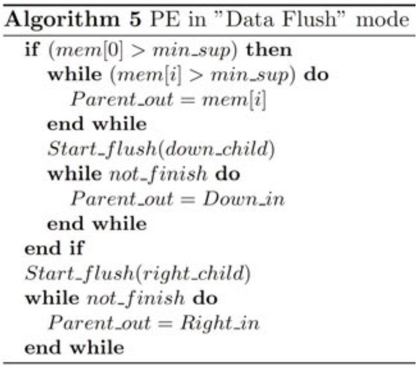
$$CC_{data} = S * (nf + 1) + 1, \text{ where } S = \lceil T/bw \rceil$$

For the data flush stage, CC_{flush} is mainly defined by the number of *PEs* that have frequent itemsets and the number of empty *PEs* in the path up to the next *PE* with useful data. In this strategy, once the data are extracted from a *PE*, data must be extracted from the *PEl*, since there is no information to determine if they will have frequent itemsets or not. In the case of the *PEu* in the lower subtree, if there are no frequent itemsets in the current *PE*, then no frequent itemsets will be found in the lower subtree. This is because all the itemsets calculated in the subtree will have as a prefix the first itemset of the current *PE*, and if this is not frequent, then no superset of it will be frequent.

The number of *PE* with and without useful data, and the position in the systolic tree that they occupy, only depends on the nature of the data and the support threshold. In the worst case, all the *PEs* will have frequent itemsets and CC_{flush} could be calculated as $ST_n + |\text{FI set}|$.

4.1 Flush Strategy

To obtain the data from the architecture, each *PE* enters in "Data Flush" operation mode. All *PE* have a connection to the parent so all the frequent itemsets can be flushed up (*Parent_out*). In this strategy each node of the systolic tree flushes the frequent itemsets stored in local memory, then it serves as a gateway to data from the lower *PE* (*Down_in*) and when it ends, it serves as a gateway to the data of the right *PE* (*Right_in*).



Algorithm 5 PE in "Data Flush" mode

```
if ($mem[0] > min\_sup$) then
    while ($mem[i] > min\_sup$) do
        $Parent\_out = mem[i]$
    end while
    $Start\_flush(down\_child)$
    while $not\_finish$ do
        $Parent\_out = Down\_in$
    end while
end if
$Start\_flush(right\_child)$
while $not\_finish$ do
    $Parent\_out = Right\_in$
end while
```

Extracting the data through the root node of the systolic tree allows to take advantage of the characteristics of the itemsets. These characteristics permit to define heuristics to prune subtrees in the flush strategy and therefore shortens the amount of data that is necessary to flush from the architecture. Because of this, less time is needed to complete the task.

5 Implementation Results

The proposed architecture was modeled using VHDL and it was verified in simulation with ModelSim SE 6.5 simulation tool. Once the architecture was validated, it was synthesized using ISE 9.2i. The target device was set to a Xilinx Virtex-4 XC4VFX140 with package FF1517 and −10 speed grade.

For the experiments, three datasets from [4] were used: Chess, Accidents and Retail. As explained in Section 4, the size of the architecture increases ruled by the number of frequent items that it is capable of processing, so the size of the device being used will highly affect the time it will take to finish the task. For the experiments we used a 32 bits data vector and the biggest architecture that fits the device was a structure to process 11 items with 264 *PE*s. This architecture consumes 74.6% of the LUTs (94,248 out of 126,336) and 18, 6% of the flip-flops (23,496 out of 126,336) available in the device.

Since the architecture is completely decentralized, there are no global connections and thus the maximum operating frequency is not affected by the number

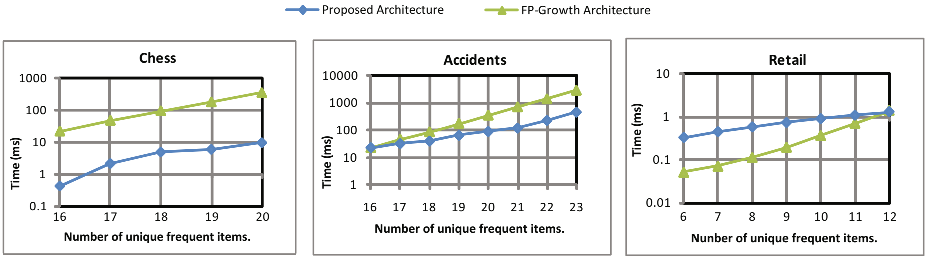


Fig. 3. Mining time comparison

of PEs of the architecture. The maximum frequency obtained for this architecture was 137 MHz. The proposed algorithm is sensitive to the number of frequent items more than the support threshold, so to show the behavior more precisely the experiments were carried out based on this variable.

In Figure 3 we compare the mining time of the architecture against the best time of the FP-Growth architecture presented in [9]. This Figure shows that the greatest improvement in execution time was for the Chess database, where more than one order of magnitude was obtained. In the other two databases it is shown how the execution time of the proposed architecture grows slower than the FP-Growth architecture when increasing the number of frequent items. Moreover, when increasing the ratio between the number of obtained frequent itemsets and the size of the database, the percentage of CC_{data} decreases. In the case of Accidents and Retail databases CC_{data} is 99% and 97% of total time correspondingly and for Chess database the percentage is between 62% and 68%. This is because the execution time of the “Data in” mode depends only on the amount of frequent items and the number of transactions that the databases have and the remaining time will depend on the number of frequent itemsets obtained.

This behavior shows a feature of the algorithm and its scalability. The architecture performs better when the density of the processed database increases and generally performs better when the number of frequent itemsets obtained also increases. This feature (better performance at higher density) has a great importance if it is considered that the denser the databases the more difficult to obtain frequent itemsets and the higher the number of frequent itemsets obtained.

For sparse databases and high supports (low number of frequent items), this task can be solved without the necessity of appealing to hardware acceleration in most cases. This need is more evident when dealing with large databases with low threshold of support or high density, or both. Because of this, this feature is desirable in the algorithms for obtaining frequent itemsets.

6 Conclusion

In this paper we proposed a parallel algorithm that is specially designed for environments which allow a high number of concurrent processes. This characteristic best suits a hardware environment and allows to use a task parallelism with fine granularity. Furthermore, a hardware architecture to validate the algorithm was developed. The experiments show that our approach outperforms the FP-Growth architecture presented in [10] by one order of magnitude when processing dense databases, and the mining time grows slower than the FP-Growth architecture mining time when processing sparse matrices.

References

1. Agrawal, R., Shafer, J.C.: Parallel mining of association rules design, implementation and experience. Technical Report RJ10004, IBM Research Report (February 1996)
2. Baker, Z.K., Prasanna, V.K.: Efficient Hardware Data Mining with the Apriori Algorithm on FPGAs. In: Proc. of the 13th Annual IEEE Symposium on Field Programmable Custom Computing Machines 2005 (FCCM '05), pp. 3–12 (2005)
3. Baker, Z.K., Prasanna, V.K.: An Architecture for Efficient Hardware Data Mining using Reconfigurable Computing System. In: Proc. of the 14th Annual IEEE Symposium on Field Programmable Custom Computing Machines 2006 (FCCM '06), pp. 67–75 (2006)
4. Goethals, B.: Frequent itemset mining dataset repository, http://fimi.cs.helsinki.fi/data/
5. Han, E.H., Karypis, G., Kumar, V.: Scalable parallel data mining for association rules. In: Proc. of the ACM SIGMOD Conference, pp. 277–288 (1997)
6. Han, J., Pei, J., Yin, Y.: Mining frequent patterns without candidate generation. In: 2000 ACM SIGMOD Intl. Conf. on Management of Data, pp. 1–12. ACM Press, New York (2000)
7. Palancar, J.H., Tormo, O.F., Cárdenas, J.F., León, R.H.: Distributed and shared memory algorithm for parallel mining of association rules. In: Perner, P. (ed.) MLDM 2007. LNCS (LNAI), vol. 4571, pp. 349–363. Springer, Heidelberg (2007)
8. Park, J., Chen, M., Yu, P.: An effective hash based algorithm for mining association rules. In: Carey, M.J., Schneider, D.A. (eds.) SIGMOD Conference, pp. 175–186. ACM Press, New York (1995)
9. Sun, S., Steffen, M., Zambreno, J.: A reconfigurable platform for frequent pattern mining. In: RECONFIG '08: Proc. of the 2008 Intl. Conf. on Reconfigurable Computing and FPGAs, pp. 55–60. IEEE Computer Society, Los Alamitos (2008)
10. Sun, S., Zambreno, J.: Mining association rules with systolic trees. In: Proc. of the Intl. Conf. on Field-Programmable Logic and its Applications (FPL), pp. 143–148. IEEE, Los Alamitos (2008)
11. Wen, Y., Huang, J., Chen, M.: Hardware-enhanced association rule mining with hashing and pipelining. IEEE Trans. on Knowl. and Data Eng. 20(6), 784–795 (2008)
12. Zaki, M.J., Parthasarathy, S., Ogihara, M., Li, W.: New algorithms for fast discovery of association rules. In: Proc. of the 3rd Intl. Conf. on KDD and Data Mining (KDD'97), pp. 283–286 (1997)

A Hybrid Methodology for Pattern Recognition in Signaling Cervical Cancer Pathways

David Escarcega[1], Fernando Ramos[1], Ana Espinosa[2], and Jaime Berumen[2]

[1] ITESM, Computer Science Department, Morelos, México
daescarcega@gmail.com, fernando.ramos@itesm.mx
[2] Hospital General de México, Unidad de Medicina Genómica,
Ciudad de México, México
anaesga@hotmail.com, jaimeberumen@hotmail.com

Abstract. Cervical Cancer (CC) is the result of the infection of high risk Human Papilloma Viruses. mRNA microarray expression data provides biologists with evidences of cellular compensatory gene expression mechanisms in the CC progression. Pattern recognition of signalling pathways through expression data can reveal interesting insights for the understanding of CC. Consequently, gene expression data should be submitted to different pre-processing tasks. In this paper we propose a methodology based on the integration of expression data and signalling pathways as a needed phase for the pattern recognition within signaling CC pathways. Our results provide a top-down interpretation approach where biologists interact with the recognized patterns inside signalling pathways.

1 Introduction

Cervical Cancer (CC) is one of the most widespread cancers in women worldwide [1]. Cervical carcinogenesis is caused by an infection of high-risk Human Papilloma Viruses (hrHPV) [2]. After hrHPV infection and CC progression other transformation events occur within the cell, for instance, deregulation of genes expression levels and alteration of cellular processes either metabolic or signaling cascades [3].

Based on the integration of signaling pathways and high-throughput gene expression data, biologists seek to find modified or unchanged cellular processes related with cervical carcinogenesis or CC progression. Signaling pathways regulate the reception of external biochemical information, that will affect processes inside the cell, or intracellular interchange information; assemble of cascade events within the cell and finally, activate of cellular response to internal or external stimuli. Meanwhile, thousands of genes transcription levels can be measured using a single microarray [4], either to prove or propose novel hypothesis of complex diseases, as CC, by providing gene expression profiles. Gene expression profiles allow individual comparison of genes expression between populations or extrapolation of genes state [5]. Expression profiles integrated with signaling pathways eases the process for inferring the inner state of the cellular mechanisms by providing biologists with a big picture of expression compensation of

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 301–310, 2010.

genes, probably related with CC progression. Microarray data should be normalized before subsequent analysis could be accomplished. Normalization is the removal of technical noise, generated by experimental protocol, leaving expression profiles intact [6]. Once expression data is normalized, different workflows to infer internal cellular behavior could be followed.

Clustering of gene expression data is a common workflow to infer unknown genes function, find new disease subclasses, and primarily, data reduction and visualization. Clustering approaches group genes with similar expression level by measuring closeness in a quantitative way [7]. Clustering methods focus on quantitative data are considered 'unsupervised' methods [8], meaning that no gene functionality or previous phenotypic is considered for gene classification. Clustering approaches provide an overall picture of data variation. Classified expression profiles can be enriched with ontology data or cellular context, i.e. Gene Ontology [9]. An alternative method that has acquired an increased attention from genomic and computational scientists is to use pathway contexts to infer cellular processes alterations [10]. The pathway context provides biologists with a functional perspective, visualization of cellular processes and the impact of genes expression variations in such processes [11]. Furthermore, pathway analysis goes beyond the genes list interpretation of expression levels by considering cellular interactions associated with a phenotype [12]. Pathways could also be stored and enriched by biologists' expertise interactions or inserting new data provided by metabolomics or proteomics experimentation [13]. Based on the implementation of the methodology of data integration proposed in this work, the results will contribute to facilitate the interpretation of CC gene expression data and the inference of hypothesis formulation made by biologist interactions. The integration of recognized patterns into signaling pathways, represented by Petri nets, simplifies as well the interaction with biologists for the enrichment process of signaling pathways.

In this work an introduction is presented in section1. The remainder of the paper is organized as follow; section 2 exposes relevant works related with data bases of signaling pathways and gene expression data; section 3 provides an introduction of computational models and signaling pathways; section 4 describes our methodology and in section 5 we expose our conclusion and future work.

2 Databases of Signaling Pathways and Gene Expression Data

Nowadays, available pathway databases contain organized gene regulation relationships mapped into metabolic or signaling pathways, for instance, KEGG [16], BioCarta [14] and MetaCyc [15]. Signaling pathway databases are mostly used as inert diagrams of signaling pathways, as KEGG or Biocarta. Nevertheless, some databases also provide XML or SQL interfaces of pathways data, as KGML which provides an XML abstraction of the KEGG pathway database [16]. Other efforts to collect gene regulation data, on a large scale, are based on using text mining approaches, iHOP [17].

KEGG provides a curated reference to study and analyze metabolic and signaling pathways, including different cellular processes [16]. KEGG offers metabolic and non-metabolic pathways. KGML data lacks of the details provided by pathways diagrams, for instance, some relations between proteins are not included in the KGML data. The KEGG pathway database is widely used and different approaches have been proposed to integrate the KEGG knowledge base into pathway modeling. Heiner and Koch [18], modeled apoptosis, from KEGG apoptosis diagrams, and provided a qualitative Petri Net model, enabling the confirmation of known properties as well as new insights of intrinsic and extrinsic apoptotic pathways. Other tools, as KEGGraph [19] converts KGML data into graphs, capturing the topology of KEGG diagrams; KEGGanim [20] is a visualization tool that integrates pathways and microarray expression data but lacks of interaction with biologists to enrich the signaling pathways. Cell Illustrator [21] has a connection to the KGML repository; however it is limited to the metabolism pathway acquisition. Alternatively, KEGG converter [22] is an online tool that emphasizes the conversion of KGML data into executable SBML models. In this work, we work with EIP and CP non-metabolic pathways from the KEGG database; we complement and integrate KGML data and gene expression data for pattern recognition within signaling transduction cascades. We emphasize the interaction and enrichment of results through biologists' expertise.

3 Computational Models and Developments

Different computational models could be frameworks for experimental interpretation, as expression microarrays. Notice that, acquired models could be validated, improved and enriched with accurate interpretation made by biologists. To achieve this goal several computational and formal models have been proposed, for instance, Boolean networks [23], Bayesian networks [24], graph interaction networks [25] and Petri nets [26].

Petri nets (PNs), proposed by Carl Petri, are bipartite graph representation of processes useful both for visualization and computational analysis of dynamic systems. PNs are a directed-bipartite graph with two types of nodes: places and transitions. Reddy et al. apply PNs to represent biochemical reactions networks [27]. PNs graph-structure enables biologist to track processes and the interactions among their elements. Places represent static elements of the system and transitions correspond to interactions between elements of the system. Transitions are a powerful tool representing interactions that could result in relevant semantic significances of the processes involved in the system. A formal overview of PNs related with biological systems is exposed in [28].

Different extensions of standard PNs have been proposed to model signaling pathways: coloured petri nets have been applied for modelling EGF signaling pathway [29]; stochastic petri nets captures uncertainty related within pathways [30]; finally, Matsuno et al proposed an extension of PNs to model continuous and discrete behaviours in a system [31]. In this work, we use PNs as a tool that facilitates the interaction of biologists with the modeled system.

4 Methodology

In this section, we describe the proposed methodology to integrate cervical cancer expression data and KGML data; recognition of patterns in signaling pathways, and lastly, recognized patterns to be enriched with biologists' interactions by modifying Petri net models. Fig. 1 synthesizes the tasks we propose.

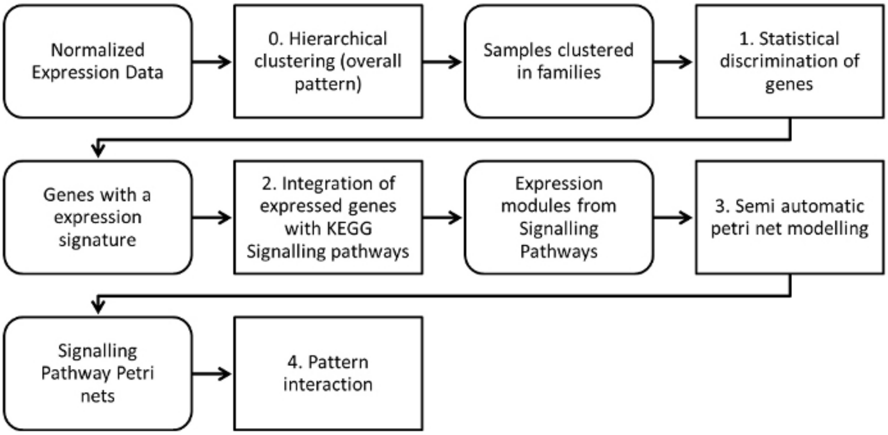

Normalized Expression Data
0. Hierarchical clustering (overall pattern)
Samples clustered in families
1. Statistical discrimination of genes
Genes with a expression signature
2. Integration of expressed genes with KEGG Signalling pathways
Expression modules from Signalling Pathways
3. Semi automatic petri net modelling
Signalling Pathway Petri nets
4. Pattern interaction

Fig. 1. This figure shows the methodology we propose. Rounded rectangles stand for input or output data, initial data is a list of normalized gene expression data and through methodology is transformed into a Petri net model. Normal rectangles represent a task or process that transforms input data.

4.1 Hierarchical Clustering

In this paper, we applied our methodology to a dataset of thirty nine cases and twelve controls. A case represents a sample of CC tissue; a control represents normal tissue. All samples were analyzed with the Affymetrix HG-Focus gene expression microarray. Each microarray represents over 8,500 genes from the NCBI RefSeq [32]. The dataset was obtained by the Unidad de Medicina Genómica team of the Hospital General de México.

Several algorithms to normalize expression data have been developed. In this work, initial input data was normalized with FlexaArray, which is a statistical program for expression microarray processing [33]. We applied a robust multi-array average (RMA) algorithm [34]. A matrix of expression data is the output of the RMA accomplishment.

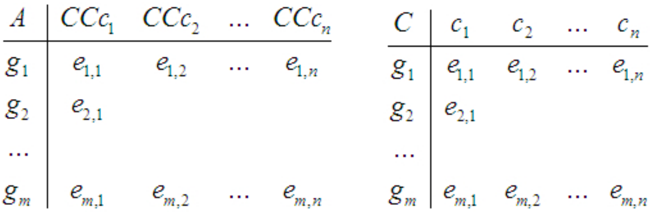
With initial input data, our first question to answer is weather controls and cases have different expression profiles. Therefore, we performed an unsupervised clustering; using the R hierarchical clustering tools [35]. First, a Pearson test to measure the correlation and dependence between samples, and secondly, we use the Spearman correlation to group genes, dendogram with genes clustering not shown.The first clustering aims to express the certainty that gene expression profiles are well-differentiated between cases and controls. In fact, four clusters of CC cases expression profiles are close in distance. Nevertheless, our first clustering is focus on quantitative data and provides biologists with a global reference of data and no evidence of cellular processes if provided. In the following section, we try to answer our next question, which genes with an expression level are important to each CC case in comparison with controls.

4.2 Statistical Discrimination of Genes

So far, hierarchical clustering delivers a differentiation between cases and controls; and four clusters of CC cases. In order to assign a significant over or sub expression level to each gene we use a z-score. Z-scores are assigned to each gene by grouping a CC cluster with the control group, using the matrix shown in figure 2. Then, z-score is calculated for each gene to assign them over, normal or sub expression values [36]. Z-scores are calculated by subtracting the total average gene intensity, within a cervical cancer group and control group, from the raw intensity data for each gene, and dividing that result by the standard deviation (SD) of all of the measured intensities, according to the formula:

$$z - score = (g_x - mean_{g_1} \dots g_n)/SD_{g_1} \dots g_n \tag{1}$$

Then, z-score is calculated for each gene to assign them over, normal or sub expression values. Genes with a z-score value under -1.96 are considered to be under expressed with respect to the media and genes with a z-score over 1.96 are considered to be over expressed with respect to the media. Z-score provides a discriminant by assigning an expression level to each gene per CC case within a cluster obtained in hierarchical clustering. A z-score with a value of 1.96, either negative or positive, represents a significant value of 0.05 for a gene to be up or down regulated. The statistical discrimination outputs a list of over and under expressed genes, which now will be integrated with KGML data to identify a signaling pathway context for each gene involved in a signal transduction process.



$$\begin{array}{c|cccc} A & CCc_1 & CCc_2 & \dots & CCc_n \\ \hline g_1 & e_{1,1} & e_{1,2} & \dots & e_{1,n} \\ g_2 & e_{2,1} & & & \\ \dots & & & & \\ g_m & e_{m,1} & e_{m,2} & \dots & e_{m,n} \end{array} \qquad \begin{array}{c|cccc} C & c_1 & c_2 & \dots & c_n \\ \hline g_1 & e_{1,1} & e_{1,2} & \dots & e_{1,n} \\ g_2 & e_{2,1} & & & \\ \dots & & & & \\ g_m & e_{m,1} & e_{m,2} & \dots & e_{m,n} \end{array}$$

Fig. 2. In matrix A will be represented the first cluster obtained in hierarchical clustering, where, each row, g_{m} , represent a gene; each column represent a cervical cancer case, CCc_{n}; and each cell is gene expression value, $e_{\mathrm{m,n}}$. In matrix C, each row g_{m} represents a gene; each column, C_{n}, represents a control sample; and each cell is gene expression value, $e_{\mathrm{m,n}}$.

4.3 Integration of Relevant Genes and KEGG Signaling Pathways

In this section, we describe the integration of over and sub expression genes and KGML data to find each gene context within signaling pathways. As mentioned before, we work with environmental information processing (EIP) and cellular processes (CP) signaling pathways from the KEGG database.

At this point of the methodology, two subtasks must be achieved to integrate expression data and signaling pathways. First, the KGML data files are downloaded, directly from the KEGG ftp, subsequently; each KGML file is parsed to extract information. A local database, named KGMLD, is created to store information of pathways, genes of each pathway and relations between genes.

And secondly, each gene, from the microarray, with an over or sub express z-score is associated with a gene expression level by gene name matching from the KGMLD. Context for genes, with an expression level, is accomplished by searching genes that interact directly to the gene with a significant expression level.

Integration of signaling pathways and expression data is presented to biologists as shown in figure 3. Figure 3 depicts the process of integration of significant genes and a signaling pathway context, it is exemplified using a segment from the MAPK signaling pathway: 3A) first, a set of genes G, with a significant expression level, is presented to biologists; 3B) then, a set of adjacent genes N, where each gene gi from G is adjacent to one or more genes from N; 3C) finally, a set of relations, R. Relations associates each gene gi with adjacent genes belonging to the set N. Each gene, gi, could be connected with one or more genes of N.

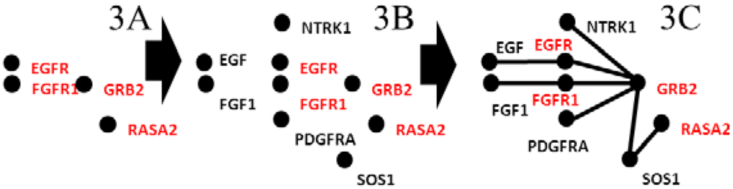


Fig. 3. This figure shows the steps to be accomplished in the integration of significant gene expression data into signaling pathway context. Vertices or genes with a name in red represent genes with a significant expression level, 3A. Genes with a name in black represent adjacent genes, 3B. Finally, relationships between both sets of genes take place by linking them, 3C.

In the subsequent task, the set of recognized graphs, GG, is integrated with the complete signaling pathway. In this example, only three graphs are displayed nonetheless the Petri net model will incorporate the complete set of graphs. Probably, not all these genes are biologically interesting. Nevertheless, we recognized a substructure inside the signaling pathway and are presented to be interpreted.

4.4 Semi-automatic Petri Net Modeling

The full context of the integration of expression data and signaling pathways is achieved in this step. As previous steps, the following subtasks are essential to achieve the final model. Firstly, a petri net model is created from KGML data, stored in the KGMLD; secondly, as mentioned, KGML data contains broken relations or missing elements, we manually incorporated missing elements based on KEGG diagram of the signaling pathway and saved in the KGMLD; finally, the set of gene graphs obtained are displayed in the proper signaling pathway petri net model. Figure 4 shows a segment of the MAPK signaling pathway with recognized expression graphs, for lack of space we present a representative segment of the MAPK signalling pathway.

As shown in Figure 4, blue places represent adjacent genes of those denoting a significant expression level and whose variation in expression could impact part of the process and genes that interact directly with them, in this particular case, a sub module of the MAPK signaling pathway. The Petri net model provides a framework for the interaction with biologists who will be able to validate or enrich the recognized patterns; in the following section we describe in detail such interaction.

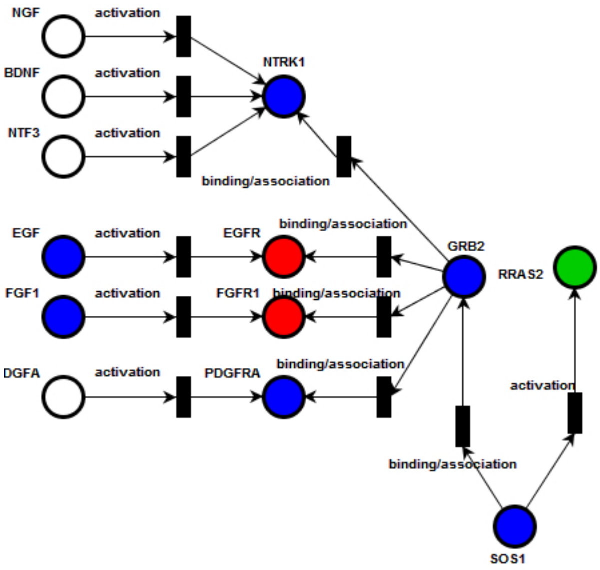


Fig. 4. This figure represents the integration of a segment of the MAPK signaling pathway and recognized genes graphs. Places in blue are genes or compounds that interact with genes with a significant expression level. Places in red denoted genes with a sub expression value, while places in green have an over expression level.

4.5 Interaction of Biologist with the Framework

The signaling pathway model represented by visual Petri nets provides biologists with an intuitive abstraction to interact. The Petri net model could be refined by incorporating personal knowledge, new data or by modifying the structure of the pathway. Operations of addition, deletion or modification of places and transitions are provided by the Petri net tool. Thus, recognized patterns require proper operations to be manipulated.

Sub-graphs or building blocks provide biologists with the capacity to manipulate patterns recognized within a signaling pathway for a better interpretation of expression microarray data. Interactions with the final output could be endless according with the interpretation or biological pursue. As demonstrated, an integrative perspective of data requires the coordination of different algorithms and computational models.

5 Conclusion

In this work, we have proposed a methodology to facilitate the interpretation of CC gene expression data and the inference of hypothesis by providing a signaling pathway context. Clustering methods, statistical discrimination, data preprocessing and systems modeling are integrated tasks to aid biologist to clarify the inner compensatory gene expression mechanisms of cervical cancer cells. The steps proposed by the methodology achieve the following: data reduction of expression profiles, selection of significantly altered genes and a visual representation of signaling pathways probably involved in the CC progression.

The facility to interrogate expression levels of thousands of genes in one experiment gives biologists a fresh look of cellular machinery compensatory events. The integration of cellular context and high-throughput expression microarray data increases the understanding of cellular systems, by providing a more interpretative model for cancer biology. At the same time the hybrid approach provides a framework to validate hypothesis.

Finally, a pattern within a signalling pathway might be represented by repetitive mutated substructure within the cascade, for instance a motif. A possible limitation of this methodology is the constant validation by biologists. In this methodology, we proposed a Petri net representation to visualize and, more significantly, to interact with identified patterns within a signalling pathway for interpretation of CC progression rather than automatization of pattern analysis.

References

1. Ferlay, J., Bray, F., Pisani, P., Parkin, D.M.: GLOBOCAN 2002; cancer incidence, mortality and prevalence worldwide. Iarc. Cancer Base No. series 5 Version 2.0. IARC Press, Lyon (2004)
2. zur Hausen, H.: Papilloma viruses in the causation of human cancers - a brief historical account. Virology 384, 260–265 (2009)

3. Jayshree, R.S., Sreenivas, A., Tessy, M., Krishna, S.: Cell intrinsic and extrinsic factors in cervical carcinogenesis. Indian J. Med. Res. 103, 286–295 (2009)
4. Ramaswamy, S., Golub, T.R.: DNA microarrays in clinical oncology. J Clin. Oncol. 20, 1932–1941 (2002)
5. Segal, E., Friedman, N., Kaminski, N., Regev, A., Koller, D.: From signatures to models: Understanding cancer using microarrays. Nat. Genet. 37, 38–45 (2005)
6. Irizarry, R.A., Hobbs, B., et al.: Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics 4(2), 249–264 (2003)
7. Spirin, V., Mirny, L.A.: Protein complexes and functional modules in molecular networks. Proc. Natl. Acad. Sci. USA 100, 12123–12128 (2003)
8. Jain, A.K., Dubes, R.C.: Algorithms for Clustering Data. Prentice-Hall, Englewood Cliffs (1988)
9. The Gene Ontology Consortium. The Gene Ontology (GO) database and informatics resource. Nucleic Acids. Res. 32, 258–261 (2004)
10. Goh, K.-I., Cusick, M.E., Valle, D., Childs, B., Vidal, M., Barabasi, A.L.: The human disease network. Proc. Natl. Acad. Sci. 104, 8685–8690 (2007)
11. Barabsi, A.L., Oltvai, Z.: Network biology: understanding the cells functional organization. Nat. Rev. Genet. 5, 101–113 (2004)
12. Nam, D., Kim, S.Y.: Gene-set approach for expression pattern analysis. Brief. Bioinform. 79, 189–197 (2008)
13. Delongchamp, R., Lee, T., Velasco, C.A.: Method for computing the overall statistical significance of a treatment effect among a group of genes. BMC Bioinformatics 7, S11 (2006)
14. BioCarta pathways, http://www.biocarta.com/
15. Caspi, R., Foerster, H., Fulcher, C.A., et al.: The MetaCyc database of metabolic pathways and enzymes and the BioCyc collection of pathway/genome databases. Nucleic Acids. Res. 36, D623–D631 (2008)
16. Kanehisa, M., et al.: From genomics to chemical genomics: new developments in KEGG. Nucleic Acids. Res. 34, 354–357 (2006)
17. Hoffmann, R., Valencia, A.: Implementing the iHOP concept for navigation of biomedical literature. Bioinformatics 21(ii), 252–258 (2005)
18. Heiner, M., Koch, I.: Petri net based model validation in systems biology. In: Cortadella, J., Reisig, W. (eds.) ICATPN 2004. LNCS, vol. 3099, pp. 216–237. Springer, Heidelberg (2004)
19. Zhang, J., Wiemann, S.: KEGGgraph: a graph approach to KEGG Pathway in R and Bioconductor (2008)
20. Adler, P., Reimand, J., Janes, J., Kolde, R., Peterson, H., Vilo, J.: KEGGanim: pathway animations for high-throughput data. Bioinformatics 24(4), 588–590 (2008)
21. Cell Illustrator, http://www.cellillustrator.org/
22. KEGG Converter, http://www.grissom.gr/keggconverter/
23. Shmulevich, I.: Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks. Bioinformatics 18, 261–274 (2002)
24. Kim, S.Y.: Inferring gene networks from time series microarray data using Bayesian networks, Brief. Bioinform. 34, 228–235 (2003)
25. Aittokallio, T., Schwikowski, B.: Graph-based methods for analysing networks in cell biology Brief. Bioinform. 7(3), 243–255 (2006)
26. Nagasaki, M., et al.: Petri Net Based Description and Modeling of Biological Pathways. Algebraic Biology, 19–31 (2005)
27. Reddy, V.N., Mavrovouniotis, M.L., Liebman, M.N.: Petri net representations in metabolic pathways. In: Proceedings of the ISMB, pp. 328–336 (1993)

28. Pinney, J.W., Westhead, D.R., McConkey, G.A.: Petri Net representations in systems biology. Biochem. Soc. Trans. 31(Pt 6), 1513–1515 (2003)
29. Zielinski, R., et al.: The crosstalk between EGF, IGF, and Insulin cell signaling pathways-computational and experimental analysis. BMC Systems Biology 3, 88 (2009)
30. Gilbert, D., Heiner, M., Lehrack, S.: A unifying framework for modelling and analysing biochemical pathways using Petri nets. In: Calder, M., Gilmore, S. (eds.) CMSB 2007. LNCS (LNBI), vol. 4695, pp. 200–216. Springer, Heidelberg (2007)
31. Matsuno, H., Tanaka, Y., Aoshima, H., Doi, A., Matsui, M., Miyano, S.: Bio pathways representation and simulation on hybrid functional Petri net. In: Silico Biology (2003)
32. Affymetrix, http://www.affymetrix.com
33. FlexArray: statistical data analysis software for gene expression microarrays, http://genomequebec.mcgill.ca/FlexArray
34. Irizarry, R.A., Hobbs, B., Collin, F., Beazer-Barclay, Y.D., Antonellis, K.J., Scherf, U., Speed, T.P.: Exploration, normalization, and summaries of high density oligonucleotide array probe level data. Biostatistics, Oxford, England 4(2), 249–264 (2003)
35. The R package, http://cran.r-project.org/
36. Cheadle, C., Vawter, M.P., Freed, W.J., Becker, K.G.: Analysis of micro array data using Z score transformation. J Mol. Diagn. 5, 73–81 (2003)

Graph Indexing and Retrieval Based on Median Graphs

Francesc Serratosa, Albert Solé-Ribalta, and Enric Vidiella

Universitat Rovira i Virgili, Computer Science Department, Spain
{francesc.serratosa,albert.sole}@urv.cat,
{enric.vidiella}@estudiants.urv.cat

Abstract. M-trees are used to organize and define fast queries on large databases of Attributed Graphs. In classical schemes based on metric trees, the routing information stored in a routing tree node is a selected Attributed Graph from the sub-cluster the node represents. Depending on the sub-cluster and the application, it is difficult to select a good representative of the sub-cluster. To that aim, we propose to use Generalized Median Graphs as the main information kept in the routing nodes of the m-tree. Experimental validation shows that in database queries, the decrease of the nodes explored in the m-tree while using a Generalized Median Graph is about 20% respect using a selected Attributed Graph.

Keywords: Graph database, m-tree, graph organization, graph prototype, graph indexing.

1 Introduction

Indexing structures are fundamental tools in database technology; they are used to obtain efficient access to large collections of images. Traditional database systems manage global properties of images, such as histograms, and many techniques for indexing one-dimensional data sets have been defined. Since a total order function over a particular attribute domain always exists, this ordering can be used to partition the data and moreover it can be exploited to efficiently support queries. Several multi-dimensional indexes have appeared, such as, color, texture, shape, with the aim of increasing the efficiency in executing queries on sets of objects characterized by multi-dimensional features. Once again, ordering systems of individual orthogonal dimensions are used for partitioning the search space, so these methods can, in fact, be considered as direct extensions of the one-dimensional case.

Effective access to image databases requires queries addressing the expected appearance of searched images [1]. To this end, it is needed to represent the image as a set of entities and relations between them. The effectiveness of retrieval may be improved by registering images as structural elements rather than global features [2]. In the most practiced approach to content-based image retrieval, the visual appearance of each spatial entity is represented independently by a vector of features. Mutual relationships between entities can be taken into account in this retrieval process through a cascade filter, which evaluates the similarity in the arrangement of entities

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 311–321, 2010.

after these have been retrieved on the basis of their individual features [3]. To overcome these systems, local entities and mutual relationships have to be considered to have the same relevance and to be defined as parts of a global structure that captures mutual dependencies. In this case, the model of content takes the shape of an Attributed Graph (AG). The attributes of the vertices of the AGs represent the features of the local entities and the attributes of the arcs of the AGs represent the features of the relationships.

While the distance between two sets of independent features can be computed in polynomial time, the exact distance between two AGs is computed in exponential time respect the number of nodes of the AGs. For this reason, few contributions, of practical interest, have been proposed supporting the application of AGs to content-based retrieval from image databases [4] and [5].

Out of the specific context of content-based image retrieval, the problem of comparing an input graph against a large number of model graphs has been addressed in several approaches. In some applications, the classes of objects are represented explicitly by a set of prototypes, which means that a huge amount of model AGs must be matched with the input AG and so the conventional error-tolerant graph matching algorithms must be applied to each model-input pair sequentially. As a consequence, the total computational cost is linearly dependent on the number of model graphs and exponential (or polynomial in suboptimal methods) with the size of the AGs. For applications dealing with large databases, this may be prohibitive. To alleviate these problems, some attempts have been designed with the aim of reducing the computational time of matching the unknown input patterns to the whole set of models from the database. Those approaches assume that the AGs that represent a cluster or class are not completely dissimilar in the database and in this way only one structural model is defined from the AGs that represent the cluster; as a consequence only one comparison is needed for each cluster [6], [7] and [8].

In this paper, we show an indexing scheme implemented by an m-tree in which the cluster knowledge embedded in each node of the m-tree is represented by a Median Graph. In the experimental section, we have compared our scheme with a similar one in which the cluster information is represented by one of the AGs of the cluster [4]. We show that the use of Median Graphs instead of AGs in the m-tree scheme makes the queries more efficient. In the next section, we comment the related work and introduce our method. In section 3, we give some definitions related to AGs and Median Graphs. In sections 4 and 5, we first present the metric trees and then this technique is applied to AGs. In section 6, we experimentally evaluate our model. We finish the paper drawing some conclusions and presenting the future work.

2 Related Work and Our Proposal

Some indexing techniques have been developed for graph queries. We divide these techniques in two categories. In the first ones, the index is based on several tables and filters [9], [10]. In the second ones, the index structure is based on metric trees [4], [11], [12].

In the first group of techniques, the ones that are not based on trees, we emphasize the method developed by Shasha *et. al.* [9] called GraphGrep. GraphGrep is based on a table in which each row stands for a path inside the graph (up to a threshold length) and each column stands for a graph. Each entry in the table is the number of occurrences of the path in the graph. Queries are processed in two phases. The filtering phase generates a set of candidate graphs for which the count of each path is at least that of the query. The verification phase verifies each candidate graph by subgraph isomorphism and returns the answer set. More recently, Yan *et. al.* [10] proposed GIndex that uses frequent patterns as index features. These frequent patterns reduce the index space as well as improve the filtering rate. The main drawback of these models is that the construction of the indices requires an exhaustive enumeration of the paths or fragments which increases the memory and time requirements. Moreover, since paths or fragments carry little information about a graph, the lost of information at the filtering step seems to be unavoidable.

Considering the second group, the first time that metric trees were applied to graph databases was done by Berretti *et. al.* [4]. Attributed Graphs were clustered hierarchically according to their mutual distances and indexed by m-trees [13]. Queries are processed in a top-down manner by routing the query along the index tree. Each node of the index tree represents a cluster and it has one of the graphs of the cluster as a representative. The graph matching problem, in the tree construction and at query time, was solved by an extension of the A* algorithm that uses a look-ahead strategy plus a stopping threshold. Latter, Lee *et. al.* [11] used this technique to model graphical representations of foreground and background scenes in videos. The resulting graphs were clustered using the edit-distance metric, and similarity queries were answered using a multi-level index structure.

More recently, He and Singh [12] proposed what they called a Closure-tree. It uses a similar structure than the one presented by Berretti [4] but, the representative of the cluster was not one of the graphs but a graph prototype (called closure graph) that could be seen as the union of the AG that compose the cluster. Figure 1 shows the closure of 3 graphs. The structurally similar nodes that have different attributes in the graphs are represented in the closure graph with only one node but with more than one attribute. Closure trees have two main drawbacks. First, they can only represent discrete attributes at nodes of the AGs. Second, they tend to generalize to much the set that represent, allowing AGs that have not been used to synthesize the closure graph.

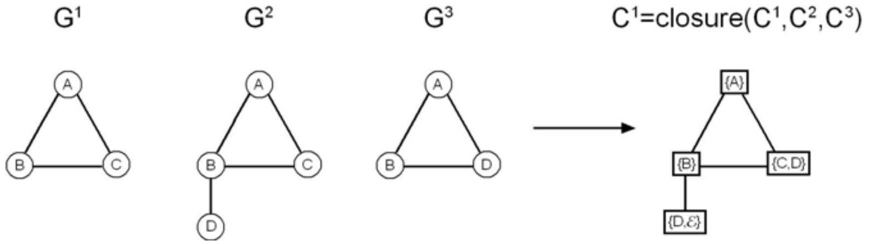


Fig. 1. Example of a Closure obtained by 3 AGs

Our proposal is to use Median Graphs as a representative of the sub-clusters in the routing nodes of the metric trees instead of an AG representative [4] or a closure graph [12]. On one hand, we aim to find a better representative of the sub-set and on the other hand, we aim to use continuous attribute values.

3 Graph Preliminaries

Given an alphabet of labels for the nodes and arcs of the AGs, *L*, we define *U* as the set of all AGs that can be constructed using labels from *L*. Moreover, we assume there is a distance function *d* between AGs.

Given $S=\{g_1,g_2,...,g_n\}\subseteq U$, the *Generalized Median Graph* $\overline{g}$ of *S* is defined as,

$$\overline{g}=\arg\min_{g\in U}\sum_{g_i\in S} d(g,g_i) \tag{1}$$

That is, the generalized median graph $\overline{g}$ of *S* is a graph $g\in U$ that minimizes the sum of distances to all the graphs in S. Notice that $\overline{g}$ is usually not a member of *S*, and in general, more than one generalized median graph may exist for a given set *S*. The computation of a generalized median graph is a NP-complete problem. Nevertheless, several suboptimal methods to obtain approximate solutions for the generalized median graph, in reasonable time, have been presented [14], [15] and [16]. These methods apply some heuristic functions in order to reduce the complexity of the graph distance computation and the size of the search space.

An alternative to the generalized median graph but less computationally demanding is the *Set Median Graph.*

$$\overline{g}=\arg\min_{g\in S}\sum_{g_i\in S} d(g,g_i) \tag{2}$$

The difference between the two models consists in the search space where the median is looked for. As it is shown in (1), the search space for the generalized median graph is *U*, that is, the whole universe of graphs. In contrast, the search space for the set median graph is simply *S*, that is, the set of given graphs. It makes the computation of set median graph exponential in the size of the graphs, due to the complexity of graph edit distance, but quadratic with respect to the number of graphs in *S*.

4 Database Indexing Based on m-trees

A metric tree [13], m-tree, is a tree of nodes, each containing a fixed maximum number of *m* entries, *< node > := {< entry >}^{m}*. In turn, each entry is constituted by a routing element *H*; a reference to the root r^H of a sub-index containing the element in the so-called covering region of *H*; and a radius d^H providing an upper bound for the distance between *H* and any element in its covering region, *< entry > := {H, r^H, d^H}.* During retrieval, triangular inequality is used to support efficient processing of range queries. That is, queries seeking for all the elements in the database which are within a given range of distance from a query element *G*. To this end, the distance between *G* and any element in the covering region of a routing element *H* can be lower-bounded using the radius r^H and the distance between *G* and *H*.

To perform range queries in Metric Trees, the tree is analyzed in a top down fashion. Specifically, if d_{max} is the range of the query and G is the query graph, the following conditions are employed, at each node of the tree, to check whether all the elements in the covering region of *H*, sub^H, can be discarded or accepted. The conditions are based on the evaluation of the distance between the routing element and the graph query *d(G,H)*.

If condition (3) holds, we will reject all elements deeper from the routing element.

$$d(G,H) \geq d_{\max} + r^{H} \quad \Rightarrow \textit{No element in } sub^{H} \textit{ is acceptabl}\text{e} \tag{3}$$

In a similar manner, the following condition checks whether all the elements in the covering region of *H*, sub^{H}, fall within the range of the query. In this case, all the elements in the region can be accepted:

$$d(G,H) \leq d_{\max} - r^{H} \quad \Rightarrow \textit{Every element in } sub^{H} \textit{ is acceptabl}\text{e} \tag{4}$$

In the critical case that neither of the two inequalities holds, the covering region of H, sub^{H}, may contain both acceptable and no acceptable elements, and the search must be repeated on the sub index sub^{H}.

5 Graph Indexing Based on Median Graphs

In this section, we first present the qualities of the Median Graphs as routing elements and second, the method used to obtain a metric tree based on Median Graphs.

Accordingly to the definition of the Median Graphs, they are supposed to be the best representatives of a set of graphs, due to they represent a graph which minimizes the sum of distances to all other graphs of the set. The advantages of using Median Graphs as routing elements in an m-tree are manifold. The main effect of using them is the reduction of the overlap between sub-clusters, due to the radius of the covering region can be more tightly adjusted. In fact, if we use the Generalized Median Graphs as a routing element, the radius of the covering region has to be equal or lower than the radius of the covering region represented by a Set Median Graph.

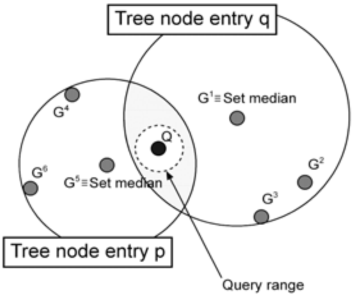


Fig. 2.1. Clusters represented by a Set Median

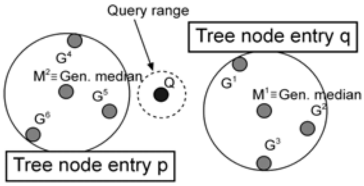


Fig. 2.2. Clusters represented by a Generalized Median

Figures 2.1 and 2.2 show the same 6 elements in two sub-clusters and the radius of their covering regions. The representative of sub-clusters in figure 2.1 is the Set Median Graph and in figure 2.2 is the Generalized Median Graph. Suppose a hypothetical query graph Q with a query range represented by the outer doted circle. The execution of the search will behave very different on both representations. In the

Set Median approach, neither entry p nor q holds for equations (3) and (4), so the sub^q and sub^p must be explored. However, due to the better representation that the Generalized Median provides, (3) holds for both tree node entries p and q. Consequently, it can be assumed that none of the entries contain any desired graph. Thus, they can be discarded and not explored.

We provide a general construction methodology from which we are able to construct a metric tree independently of the type of the routing element; a Generalized Median Graph or a Set Median Graph. Given an AG set, it is crucial to obtain the same structure of the m-tree for both types of routing elements, since we want to compare its representational power in similar conditions. We use a non-balanced tree constructed through a hierarchical clustering algorithm and complete linkage clustering. In this way, given a set of graphs, we first compute the distance matrix over the whole set and then we construct a dendogram. We obtain a set of partitions that clusters the AGs with the dendogram using some horizontal cuts. With these partitions we generate the m-tree and we synthesize a Generalized Median or a Set Median. Figure 3.1 shows an example of a dendogram. The AGs G^i are placed on the leaves of the dendogram and the Generalized Medians or Set Medians M^j are placed on the junctions between the cuts and the horizontal lines of the dendograms. Figure 3.2 shows the obtained m-tree.

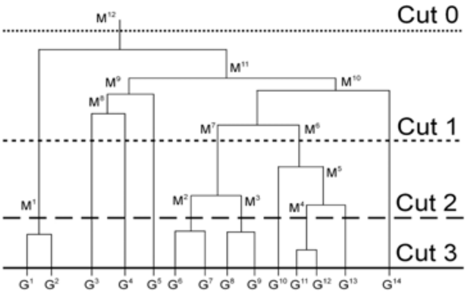


Fig. 3.1. Example of a dendogram

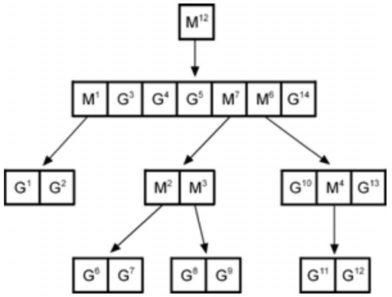


Fig. 3.2. The obtained m-tree

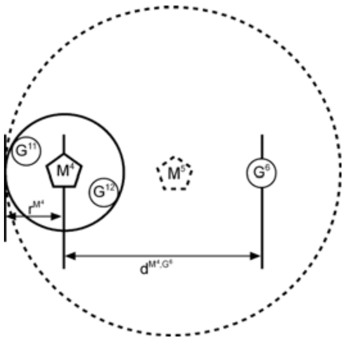


Fig. 4.1. Second radius computation

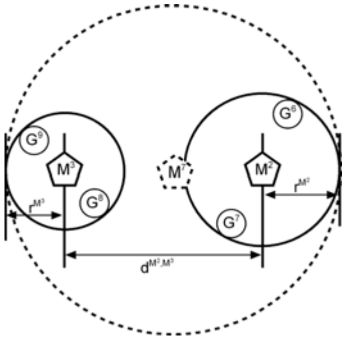


Fig. 4.2. Third radius computation rule

Computing the m-tree based on the Generalized Median Graph
At each node of the m-tree, we have to compute a Generalized Median, we use the method presented in [17]. With the aim of reducing the computational cost of computing these Medians, we compute them as pairwise consecutive computations of the Medians obtained in lower levels of the tree. For instance, to compute M^7, which appears at Figure 3.1, we only use M^2 and M^3 Medians. That is, we assume that:

$$M^7 \cong \overline{(M^2, M^3)} \cong \overline{(\overline{(G^6, G^7)}, \overline{(G^8, G^9)})} \tag{5}$$

The covering region radius r^p of the Generalized Median M^p is computed applying three rules, depending whether the type of the descendant of M^p in the dendogram is another Median (that is, a routing node of the m-tree) or an AG (that is, a leaf of the m-tree):

- When both descendants are AGs (G^a and G^b):

$$r^p = Max(Dist(M^p, G^a), Dist(M^p, G^b)) \tag{6}$$

- When a descendant is a Median (M^a) and the other is an AG(G^b):

$$r^p = Max(Dist(M^p, M^a) + r^a, Dist(M^p, G^b)) \tag{7}$$

- When both descendants are Medians (M^a and M^b):

$$r^p = Max(Dist(M^p, M^a) + r^a, Dist(M^p, M^b) + r^b) \tag{8}$$

Fig. 4.1 and 4.2 illustrate the second and third rule, respectively. In the first case, $Dist(M^5,M^4) + r^{M4}$ is greater than $Dist(M^5,G^6)$, and in the second case $Dist(M^7,M^3) + r^{M3}$ is greater than $Dist(M^7,M^2)+r^{M2}$.

Computing the m-tree based on the Set Median Graph
At each node of the m-tree, it is desired to compute the Set Median. Given the distance matrix of the whole set of AGs, the computation of the Set Median given a sub-set is simply performed by adding the pre-computed distances between the involved AGs. For instance, to compute M^7 that appears at Figure 3.1, we use the distances between the AGs G^6, G^7, G^8 and G^9.

The covering region radius r^p of the Set Median M^p is computed as the maximum distance between M^p and any of the AGs in the sub-set.

6 Evaluation

To evaluate the performance of both model, we used two indices. The first index is addressed to evaluate the quality of the tree. The lower is the overlap between the covering regions of sibling nodes, the higher is the quality of the m-tree since they are more discriminative and therefore the time to compute the query reduces.

Given two sibling nodes, we define the overlap of their covering regions as follows,

$$S(i,j)=\begin{cases}\dfrac{(R_i+R_j)}{d(i,j)} & if\ \dfrac{(R_i+R_j)}{d(i,j)}>1\\ 0 & Otherwise\end{cases} \tag{9}$$

Given a node of the m-tree, their own overlap is computed as the normalized overlap between their children. The radius of the sub-clusters that the children represent is obtained from the parameter d^H in their m-tree nodes.

$$S_g=\sum_{i=1}^{E}\sum_{j=i+1}^{E}S(N_i,N_j)\Big/\binom{E}{2} \tag{10}$$

where E is the number of entries of the m-tree node. Finally the general overlap of an m-tree is computed as,

$$S=\sum S_g\Big/numberOfNodes \tag{11}$$

The second index, called *access ratio,* is addressed to evaluate the capacity of the m-tree to properly route the queries. Given a query element, this index is the number of accessed nodes and leaves of the m-tree. That is, the number of comparisons required between the queried AG and the median graphs (in the case of nodes of the m-tree) plus the number of comparisons between the queried AG and the AGs (in the case of leaves of the m-tree). This value is normalized by the number of AGs used to generate the m-tree.

$$access\ ratio=number\ of\ comparisons\Big/number\ of\ elements \tag{12}$$

In the evaluation phase, we used the Letter database created at the University of Bern [18]. It is composed by 15 classes and 150 AGs per class representing the Roman alphabet. Nodes are defined over a two-dimensional domain that represents its plane position (x, y). Edges have a binary attribute that represents the existence of a line between two terminal points.

We constructed 12 different m-trees per each letter (or class) varying the number of dendogram partitions *{4, 7, 10 , 12}* and the number of AGs that represent each class, that is, the AGs that are used to generate the m-tree *{30, 50, 100}*. Therefore, we analyzed 15x12=180 m-trees with the Generalized Median Graph as routing elements and other 180 m-trees with the Set Median Graph as routing elements. Figures 5.1 and 5.2 show the general overlap (11) of the m-trees depending on the number of partitions and the number of AGs per class. Figure 5.3 shows the difference between the Set Median and the Generalize Median.

The overlap index is slightly lower in the Generalize Median model in comparison with the Set Median model. The difference increases when the number of AGs per partition decreases since it is statistically more difficult to find a good representative using the Set Median.

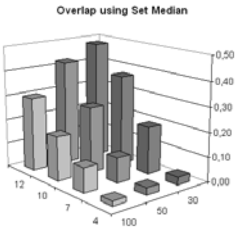


Fig. 5.1. Overlap using Set Median

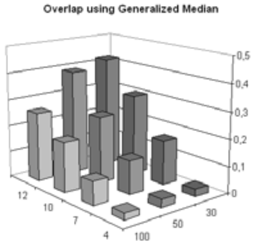
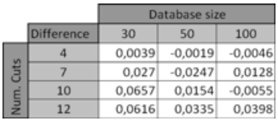


Fig. 5.2. Overlap using Generalized Median



		Database size		
	Difference	30	50	100
Num. Cuts	4	0,0039	-0,0019	-0,0046
	7	0,027	-0,0247	0,0128
	10	0,0657	0,0154	-0,0055
	12	0,0616	0,0335	0,0398

Fig. 5.3. Overlap difference of both methods

To analyze our model through the access ratio (12) we generated several queries on the above m-trees. Each test was carried out by 9 queries in which we used 9 different AGs. 3 of these AGs were used to create the m-tree, 3 AGs where not used to create the m-tree but belong to the same letter and 3 AGs belong to other letters. Figures 6 to 8 show the access ratio of these queries on m-trees with Generalized Median, Set Median and the difference between them. In these figures, we applied the following query ranges (section 4) of $d_{max} = \{D_{max}/8, D_{max} / 4, D_{max}/2\}$, respectively, where D_{max} is the maximum distance of any two AGs of the m-tree.

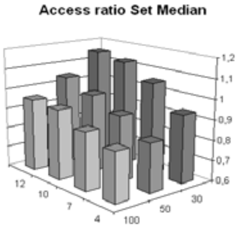


Fig. 6.1. Access ratio using $d_{max} = D_{max}/2$

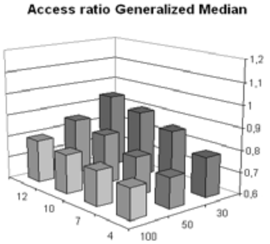
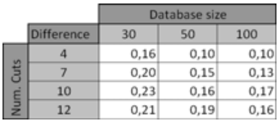


Fig. 6.2. Access ratio using $d_{max} = D_{max}/2$



		Database size		
	Difference	30	50	100
Num. Cuts	4	0,16	0,10	0,10
	7	0,20	0,15	0,13
	10	0,23	0,16	0,17
	12	0,21	0,19	0,16

Fig. 6.3. Difference of access ratio

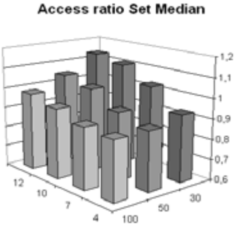


Fig. 7.1. Access ratio using $d_{max} = D_{max}/4$

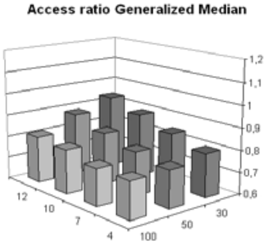
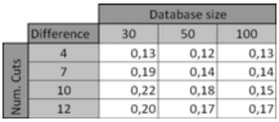


Fig. 7.2. Access ratio using $d_{max} = D_{max}/4$



		Database size		
	Difference	30	50	100
Num. Cuts	4	0,13	0,12	0,13
	7	0,19	0,14	0,14
	10	0,22	0,18	0,15
	12	0,20	0,17	0,17

Fig. 7.3. Difference of access ratio

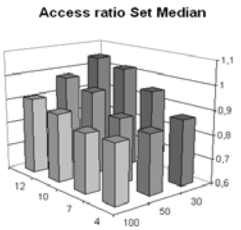


Fig. 8.1. Access ratio using $d_{max} = D_{max}/8$

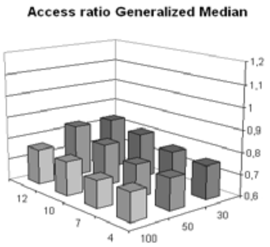
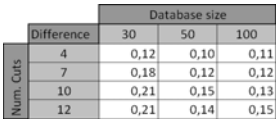


Fig. 8.2. Access ratio using $d_{max} = D_{max}/8$



		Database size		
	Difference	30	50	100
Num. Cuts	4	0,12	0,10	0,11
	7	0,18	0,12	0,12
	10	0,21	0,15	0,13
	12	0,21	0,14	0,15

Fig. 8.3. Difference of access ratio

Analyzing the experimental results, we conclude that the Generalized Median decreases the number of accesses in about 20%. As a consequence, we conclude that the Generalized Median has better representational power than the Set Median as routing objects in the m-trees. Note that the access ratio of some experiments on the Set Median is higher than one. That means that without any indexing structure, the run time would be lower.

7 Conclusions

We have presented a graph indexing technique based on metric trees and Median Graphs. Furthermore, we have compared the use of the Generalized Median Graph and the Set Median Graph as routing elements in the m-trees. We arrive at the conclusion that the construction of the m-tree is computationally harder using the Generalized Median Graph but better performance can be obtained while using them as routing elements. Experimental validation on a real database shows that the general overlap of the m-trees is lower when using the Generalized Medians instead of Set Median. Moreover, we have verified that the number of comparisons done while performing the queries is lower in the Generalized Medians than the Set Medians and so, the run time is also lower. With these results, we conclude that it is preferably to use Generalized Medians as routing elements in m-trees instead of Set Medians.

References

1. Gudivada, V.N., Raghavan, V.V.: Special issue on Content Based Image Retrieval Systems. Computer 28(9) (1995)
2. Tao, Y., Grosky, W.I.: Spatial Colour Indexing: A Novel approach for Content-Based Image Retrieval. In: Proc. IEEE International Conference Multimedia Computing and Systems (1999)
3. Smith, J.R., Samet, H.: VisualSEEk: A Fully Automated Content-Based Image Query System. In: Proc. ACM Multimedia, pp. 87–98 (1996)
4. Berretti, S., Del Bimbo, A., Vicario, E.: Efficient Matching and Indexing of Graph Models in Content-Based Retrieval. IEEE Transactions on Pattern Analysis and Machine Intelligence 23(10), 1089–1105 (2001)

5. Zhao, J.L., Cheng, H.K.: Graph Indexing for Spatial Data Traversal in Road Map Databases. Computers & Operations Research 28, 223–241 (2001)
6. Serratosa, F., Alquézar, R., Sanfeliu, A.: Function-described graphs for modeling objects represented by attributed graphs. Pattern Recognition 36(3), 781–798 (2003)
7. Serratosa, F., Alquézar, R., Sanfeliu, A.: Synthesis of Function-Described Graphs and clustering of Attributed Graphs. International Journal of Pattern Recognition and Artificial Intelligence 16(6), 621–655 (2002)
8. Sanfeliu, A., Serratosa, F., Alquézar, R.: Second-Order Random Graphs for modeling sets of Attributed Graphs and their application to object learning and recognition. International Journal of Pattern Recognition and Artificial Intelligence 18(3), 375–396 (2004)
9. Shasha, D., Wang, J.T.L., Giugno, R.: Algorithmics and applications of tree and graph searching. In: ACM SIGMOD-SIGACT-SIGART Symposium on Principles of Database Systems, pp. 39–52 (2002)
10. Yan, X., Yu, P.S., Han, J.: Graph indexing: a frequent structure-based approach. In: ACM SIGMOD International Conference on Management of Data, pp. 335–346 (2004)
11. Lee, S.Y., Hsu, F.: Spatial Reasoning and Similarity Retrieval of Images using 2D C-Strings Knowledge Representation. Pattern Recognition 25(3), 305–318 (1992)
12. He, H., Singh, A.K.: Closure-Tree: An Index Structure for Graph Queries. In: Proc. International Conference on Data Engineering, p. 38 (2006)
13. Ciaccia, P., Patella, M., Zezula, P.: M-tree: An Efficient Access Method for Similarity Search in Metric Spaces. In: Proc. 23rd VLDB Conference, pp. 426–435 (1997)
14. Jiang, X., Münger, A., Bunke, H.: On median graphs: Properties, algorithms and applications. IEEE Trans. on Pattern Analysis and Machine Intelligence 23(10), 1144–1151 (2001)
15. Ferrer, M., Valveny, E., Serratosa, F., Riesen, K., Bunke, H.: Generalized Median Graph Computation by Means of Graph Embedding in Vector Spaces. Pattern Recognition 43(4), 1642–1655 (2010)
16. Ferrer, M., Valveny, E., Serratosa, F.: Median graphs: A genetic approach based on new theoretical properties. Pattern Recognition 42(9), 2003–2012 (2009)
17. Neuhaus, M., Riesen, K., Bunke, H.: Fast Suboptimal Algorithms for the Computation of Graph Edit Distance. In: Yeung, D.-Y., Kwok, J.T., Fred, A., Roli, F., de Ridder, D. (eds.) SSPR 2006 and SPR 2006. LNCS, vol. 4109, pp. 163–172. Springer, Heidelberg (2006)
18. Riesen, K., Bunke, H.: IAM Graph Database Repository for Graph Based Pattern Recognition and Machine Learning. In: da Vitoria Lobo, N., Kasparis, T., Roli, F., Kwok, J.T., Georgiopoulos, M., Anagnostopoulos, G.C., Loog, M. (eds.) S+SSPR 2008. LNCS, vol. 5342, pp. 287–297. Springer, Heidelberg (2008)

A Novel Method for Counting Models on Grid Boolean Formulas

Carlos Guillén[1], Guillermo De Ita[1], and Aurelio López-López[2]

[1] Facultad de Ciencias de la Computación, BUAP
[2] Instituto Nacional de Astrofísica, Óptica y Electrónica
{cguillen,deita,allopez}@ccc.inaoep.mx

Abstract. We research on the possible orientations patterns of a grid graph G, and propose a method for counting certain combinatorial structures over the class of orientations of G. For example, our method can be applied for counting sink-free orientations of G, as well as it can be applied for solving the #2SAT problem for grid Boolean formulas.

Our proposal extends the classical transfer matrix method used for counting the number of independent sets in a grid.

Keywords: Grid Computing, Transfer Matrix Method, $\#2SAT$ Problem.

1 Introduction

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

In general, constraint satisfaction problems (CSP) are NP-hard; and considerable efforts, both practical and theoretical, have been made to identify tractable classes for that class of problems [7].

One application of the CSP has been for recognizing combinatorial patterns on graphs and to apply techniques developed for the CSP problem for solving different combinatorial problems on graphs. For example, given an undirected graph $G = (V, E)$, we can associate a monotone 2-CNF formula F_G with variables V, and where $F_G = \bigwedge_{(u,v)\in E}(u \vee v)$, a conjunctive normal form F_G is called *monotone* when each variable of F_G occurs with just one of its two signs.

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

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 322–331, 2010.

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

Other interesting problem to be modeled using 2-CNF's is about the orientation of a graph. Considering again, an undirected graph $G = (V, E)$, an orientation of G is an assignment of exactly one direction to each of the edges of G. Then, an orientation of $\{u, v\} \in E$ is (u, v) denoted as $u \rightarrow v$ or (v, u) denoted as $u \rightarrow v$. For an oriented edge $u \rightarrow v$, u is called the tail and v is called the head of the edge. The number of edges where u is a head is the in-degree of u and its out-degree is the number of edges where u is the tail. A node $u \in V$ with out-degree zero is called a sink of the graph.

An orientation O of a graph is *sink-free* if no node is a sink in O. There are important and classic problems related with recognize patterns and count combinatorial structures on the orientations, like: decision, construction, unique, listing, counting sink-free graph orientations and the acyclic orientations of the graph [5,3]. For example, the decision problem SFO on instance G is to determine whether G has a sink-free orientation, and the $\#SFO$ problem is to count the number of sink-free orientations of G.

Notice that an oriented edge $u \rightarrow v$ can be represented by the constraint $(\neg u \vee v)$ and then, problems related to oriented graphs can be considered as a restricted class of the CSP. In fact, Russ [11] has shown that the SFO problem is equivalent to determine the satisfiability of Boolean Conjunctive Formulas where each literal appears exactly once, problem knowing as Twice-SAT.

In this work, we consider a more general case of the patterns of orientations of an undirected edge. Given an undirected graph $G = (V, E)$, we associate to each edge $\{u, v\} \in E$ an ordered pair (s_1, s_2) of signs assigned as the labels of the edge. The signs s_1 and s_2 are related to the signs of the literals u and v respectively. For example, the clause $(\neg x \vee \neg y)$ determines the labeled edge: “$x\overset{-\;-}{\rule{2em}{0.4pt}}y$”.

Then, we have four different orientations for any edge $\{u, v\} \in E$; when $(s_1, s_2) = (-, +)$ the edge is type $u \rightarrow v$, if $(s_1, s_2) = (+, -)$ the edge is $v \rightarrow u$, both cases are called ordinary edges. The cases where $(s_1, s_2) = (+, +)$ denoted as $v \leftrightarrow u$, and where $(s_1, s_2) = (-, -)$ denoted as $v \rightarrow\leftarrow u$ are called skews edges.

This type of orientations generalize the class of problems which could be modeled and solved through methods applied in the area of Constraint Satisfaction Problems. We present a new matrix method for recognizing and counting the number of sink-free orientations of a planar grid graph G under this class of orientations, for solving its related constraint satisfaction problem; to count the number of models of a 2-CNF on formulas whose constrained graph is a grid graph.

The constraint satisfaction problem has been a helpful language to model processing on Grid graph which is one of the most important physical graph topology for modeling parallel and distributed computing.

2 The Transfer Matrix Method and the #2SAT Problem

An undirected planar grid graph of size $m \times n$ is a graph $G_{m,n}$ with vertex set $V = \{0, ..., m\} \times \{0, ..., n\}$ and edge set $E = \{\{u, v\} : u, v \in V \wedge \|u - v\| = 1\}$. where $\|\cdot\|$ is the euclidean norm of R^2. Let $I(G_{m,n})$ be the number of independent sets of $G_{m,n}$. There is a large volume of literature devoted to recognize and count structures in a grid graph, e.g., spanning trees, Hamiltonian cycles, independent sets, acyclic orientations, k-coloring, and so on [1,4,8,6].

In other line research, the transfer matrix method is a general technique which has been used to find exact solutions for a great variety of problems. For example, Calkin [4] used this method for computing the number of independent sets over a grid graph $G_{m,n}$.

Shortly, we describe the method used by Calkin as follows. Let $\mathcal{C}_m$ be the set of all $(m + 1)$-vectors $\mathbf{v}$ of $0's$ and $1's$ without two consecutive $1's$ (the number of these vectors is F_{m+2}, the $m + 2$-th Fibonacci number). Let T_m be an $F_{m+2} \times F_{m+2}$ symmetric matrix of $0's$ and $1's$ whose rows and columns are indexed by the vectors of $\mathcal{C}_m$. The entry of T_m in position $(\mathbf{u}, \mathbf{v})$ is 1 if the vectors $\mathbf{u}, \mathbf{v}$ are orthogonal, and is 0 otherwise, T_m is called the transfer matrix for $G_{m,n}$. Then, $I(G_{m,n})$ is the sum of all entries of the n-th power matrix T_m^n, i.e., $I(G_{m,n}) = \mathbf{1}^t T_m^n \mathbf{1}$, where $\mathbf{1}$ is the (F_{m+2})-vector whose entries are all $1's$.

For example, if $m = 2$ and $n = 3$ we have that $\mathcal{C}_2 = \{(0, 0, 0), (1, 0, 0), (0, 1, 0), (0, 0, 1), (1, 0, 1)\}$,

$$T_2 = \begin{pmatrix} 1&1&1&1&1 \\ 1&0&1&1&0 \\ 1&1&0&1&1 \\ 1&1&1&0&0 \\ 1&0&1&0&0 \end{pmatrix} \quad \textit{and} \quad T_2^3 = \begin{pmatrix} 17&12&13&12&9 \\ 12&7&10&8&5 \\ 13&10&9&10&8 \\ 12&8&10&7&5 \\ 9&5&8&5&3 \end{pmatrix}$$

therefore $I(G(2, 3)) = \mathbf{1}T_2^3\mathbf{1} = 227$.

The study of $I(G_{m,n})$ is closely related to the "hard-square model" used in statistical physics and, of particular interest is the so-called "hard-square entropy constant" defined as $\lim_{m,n\to\infty} I(G_{m,n})^{1/m\cdot n}$ [1]. Applications also include for instance tiling and efficient coding schemes in data storage [10].

Given a monotone Boolean formula F in 2-conjunctive normal form (2-CNF), we can associate an undirected signed graph $G_F = (V, E)$, called its constrained graph, where V is the set of variables of F and two vertices of V are connected by an edge in E if they belong to the same clause of F. We say that a 2-CNF formula is a *cycle, path, tree*, or *grid formula* if its constrained graph is a *cycle, path, tree*, or *grid* graph respectively.

It is known that the number of independent sets of G_F is the number of satisfying assignments (models) for monotone formulas F [9]. The number of models of a Boolean formula F is denoted as #SAT(F). The computation of #SAT(F) for formulas in 2-CNF is a classic #P-complete problem [3].

In order to extend the transfer matrix method for considering any kind of 2-CNF's, we have to deal with grid graphs with signed edges.

3 Oriented Grids

For each undirected edge $e = \{u, v\} \in E$ of an grid graph $G_{m,n}$, we consider four types of orientations for e: $(+,+)$, $(+,-)$, $(-,+)$ and $(-,-)$ (see figure 1).

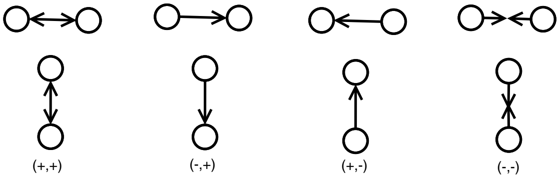


Fig. 1. Types of orientations

An oriented grid graph is a triplet $G = (V, E, \psi)$, where (V, E) is a graph (grid graph), and ψ is a function with domain E and range $\{+,-\} \times \{+,-\}$. The evaluation $\psi(e)$ is called the orientation of the edge $e \in E$.

Let $e = \{\mathbf{u}, \mathbf{v}\}$ be an edge of an oriented grid graph, if the vector $\mathbf{u} - \mathbf{v}$ is parallel to the vector $(0, 1)$, e is called column edge, if $\mathbf{u} - \mathbf{v}$ is parallel to the vector $(1, 0)$, e is called a row-edge.

A k-column of an oriented grid graph $G_{m,n}$ is the vertex-induced subgraph by the nodes

$$x_{k0}, ..., x_{km}$$

where $x_{ij} = (i, j)$. The vertex-induced subgraph by the nodes

$$x_{k0}, ..., x_{km}, x_{(k+1)0}, ..., x_{(k+1)m}$$

is denoted by $G_{m,k,k+1}$(see figure 2a).

From the k-columns and ℓ-rows of an oriented grid graph $G_{m,n}$ we define the vectors $\mathbf{s}_k, \mathbf{r}_k, \overrightarrow{\mathbf{r}}_k, \overleftarrow{\mathbf{r}}_k \in \{+,-\}^{2m}$ defined by

$$\mathbf{s}_k = (s'_{0k}, s_{1k}, s'_{1k}, ..., s_{m-1k}, s'_{m-1k}, s_{mk})$$

$$\mathbf{r}_k = (r'_{0k}, r_{1k}, r'_{1k}, ..., r_{m-1k}, r'_{m-1k}, r_{mk})$$

$$\overrightarrow{\mathbf{r}}_k = (r_{0k}, r_{1k}, r_{1k}, ..., r_{m-1k}, r_{m-1k}, r_{mk})$$

$$\overleftarrow{\mathbf{r}}_k = (r'_{0k}, r'_{1k}, r'_{1k}, ..., r'_{m-1k}, r'_{m-1k}, r_{mk})$$

The k-column nodes of $G_{m,n}$ induces a vector $\mathbf{x}_k \in \{0, 1\}^{2m}$ given by

$$\mathbf{x}_k = (x'_{k0}, x_{k1}, x'_{k1}, ..., x_{km-1}, x'_{km-1}, x_{km})$$

(see figure 2b).

The vector $\mathbf{s}_k$ is called the k-oriented vector induced by the column nodes from $G_{m,n}$. $\mathbf{x}_k$ is called the k-vector induced by the column nodes from $G_{m,n}$ and the pair $\langle \mathbf{s}_k, \mathbf{s}_{k+1} \rangle$ is called the sign vectors induced by $\mathbf{x}_k$ and $\mathbf{x}_{k+1}$.

A valuation of the nodes x_{ij} is a function φ with domain $\{0, ..., m\} \times \{0, ..., n\}$ and range $\{0, 1\}$. We define the operation $\cdot : \{+,-\} \times \{0, 1\} \to \{0, 1\}$ as +0=0,+1=1,-0=1, and -1=0. The operation

$$\odot : \{+,-\}^{2m} \times \{0, 1\}^{2m} \to \{0, 1\}$$

is defined by $(s_i) \odot (x_i) = (s_i \cdot x_i)$.

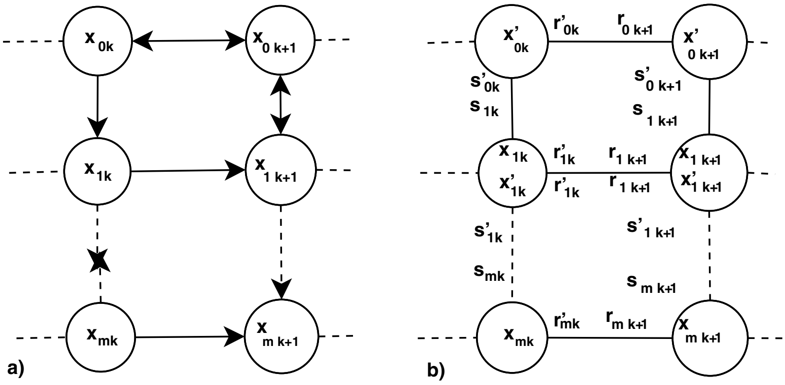


Fig. 2. a) Vertex-induced subgraph $G_{m,k,k+1}$ b) Vectors $\mathbf{s}_k, \mathbf{r}_k$

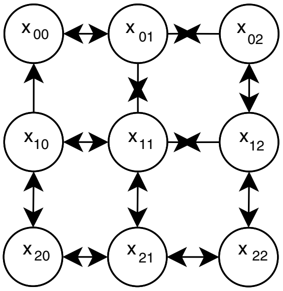


Fig. 3. Grid graph $G_{2,2}$

4 A Novel Matrix Method for Processing Grids

Let $\mathcal{F}_m$ be the set of all $2m$-vectors $\mathbf{v}$ of $0's$ and $1's$, and let $\mathcal{C}_m \subset \mathcal{F}_m$ be the set of all $2m$-vectors $\mathbf{v}$ of $0's$ and $1's$, such that $\mathbf{v}$ does not have two consecutive $1's$. The cardinality of $\mathcal{C}_m$ (denoted by $|\mathcal{C}_m|$) is F_{2m+1} (the $2m$+1-th Fibonacci number), while $|\mathcal{F}_m| = 2^{2m}$. Given $\mathbf{s} \in \{0,1\}^{2m}$, we define

$$\mathcal{F}_m^{\mathbf{s}} = \{\mathbf{e} \in \mathcal{F}_m : \mathbf{s} \odot \mathbf{e} \in \mathcal{C}_m\}$$

Following the idea proposed in [4], we define a matrix $T_k = T_{m,k}$, the transfer matrix of $G_{m,k,k+1}$ as follows. T_k is an $\mid \mathcal{F}_m^{\mathbf{s}_{k+1}} \mid \times \mid \mathcal{F}_m^{\mathbf{s}_k} \mid$ matrix of $0's$ and $1's$ whose rows and columns are indexed by vectors $(\mathbf{v}, \mathbf{u})$ of $\mathcal{F}_m^{\mathbf{s}^{k+1}} \times \mathcal{F}_m^{\mathbf{s}^k}$. The entry of T_k in position $(\mathbf{v}, \mathbf{u})$ is 1 if the vectors $\overrightarrow{\mathbf{r}}_k \odot \mathbf{u}$ and $\overleftarrow{\mathbf{r}}_{k+1} \odot \mathbf{v}$ are orthogonal, and is 0 otherwise.

Notice that if $\overrightarrow{\mathbf{r}}_k$ and $\overleftarrow{\mathbf{r}}_{k+1}$ have positive entries, then T_k is the transfer matrix used in the transfer method [4]. For example, if $G_{2,2}$ is the grid graph with labeled edges as illustrated in figure 3. For $G_{2,0,1}$, we have that $\mathbf{s}_0 = (+,-,+,+)$, $\mathbf{s}_1 = (-,-,+,+)$ and $\overrightarrow{\mathbf{r}}_0 = \overleftarrow{\mathbf{r}}_1 = (+,+,+,+)$, then $\mathcal{F}_2^{\mathbf{s}_0} = \{\mathbf{u}_1, \cdots, \mathbf{u}_4\}$ and $\mathcal{F}_2^{\mathbf{s}_1} = \{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3, \mathbf{v}_4\}$, where $\mathbf{u}_1 = (0,0,0,0)$, $\mathbf{u}_2 = (0,1,1,0)$, $\mathbf{u}_3 = (0,0,0,1)$,

$\mathbf{u}_4 = (1,1,1,0)$, $\mathbf{v}_1 = (1,0,0,0)$, $\mathbf{v}_2 = (0,1,1,0)$, $\mathbf{v}_3 = (1,0,0,1)$ and $\mathbf{v}_4 = (1,1,1,0)$. The transfer matrix $T_0 = (a_{ij})_{4\times 4}$, is a 4×4 matrix determined, for $1 \leq i,j \leq 4$, as $a_{ij} = 1$, if $(\overleftarrow{\mathbf{r}}_1 \odot \mathbf{v}_i) \cdot (\overrightarrow{\mathbf{r}}_0 \odot \mathbf{u}_j) = 0$ and $a_{ij} = 0$ otherwise. Since $\overrightarrow{\mathbf{r}}_0 = \overleftarrow{\mathbf{r}}_1 = (+,+,+,+)$, then $\overleftarrow{\mathbf{r}}_1 \odot \mathbf{v}_i = \mathbf{v}_i$ and $\overrightarrow{\mathbf{r}}_0 \odot \mathbf{u}_j = \mathbf{u}_j$. Then,

$$T_0 = \begin{pmatrix} 1\,1\,1\,0 \\ 1\,0\,1\,0 \\ 1\,1\,0\,0 \\ 1\,0\,1\,0 \end{pmatrix} \tag{1}$$

We have $\mathbf{s}_1 = (-,-,+,+)$, $\mathbf{s}_2 = (+,+,+,+)$, $\overrightarrow{\mathbf{r}}_1 = (-,-,-,+)$ and $\overleftarrow{\mathbf{r}}_2 = (-,+,+,+)$, then $\mathcal{F}_2^{\mathbf{s}_1} = \{\boldsymbol{\mu}_1,\ldots,\boldsymbol{\mu}_4\}$ and $\mathcal{F}_2^{\mathbf{s}_2} = \{\boldsymbol{\nu}_1,\ldots,\boldsymbol{\nu}_5\}$, where $\boldsymbol{\mu}_1$=(1,0,0,0), $\boldsymbol{\mu}_2$=(0,1,1,0), $\boldsymbol{\mu}_3$=(1,0,0,1), $\boldsymbol{\mu}_4$=(1,1,1,0), $\boldsymbol{\nu}_1$=(0,0,0,0), $\boldsymbol{\nu}_2$=(1,0,0,0), $\boldsymbol{\nu}_3$=(0,1,1,0), $\boldsymbol{\nu}_4$=(0,0,0,1) and $\boldsymbol{\nu}_5$=(1,0,0,1). Then,

$$\{\overrightarrow{\mathbf{r}}_1 \odot \boldsymbol{\mu} : \boldsymbol{\mu} \in \mathcal{F}_2^{\mathbf{s}_1}\} = \{(0,1,1,0),(1,0,0,0),(0,1,1,1),(0,0,0,0)\}$$

and

$$\{\overleftarrow{\mathbf{r}}_2 \odot \boldsymbol{\nu} : \boldsymbol{\nu} \in \mathcal{F}_2^{\mathbf{s}_2}\} = \{(1,0,0,0),(0,0,0,0),(1,1,1,0),(1,0,0,1),(0,0,0,1)\}$$

The transfer matrix $T_1 = (b_{ij})_{5\times 4}$, is such that, for $1 \leq i \leq 5$ and $1 \leq j \leq 4$, $b_{ij} = 1$, if $(\overleftarrow{\mathbf{r}}_2 \odot \boldsymbol{\nu}_i) \cdot (\overrightarrow{\mathbf{r}}_1 \odot \boldsymbol{\mu}_j) = 0$ and $b_{ij} = 0$ otherwise. Then

$$T_1 = \begin{pmatrix} 1\,0\,1\,1 \\ 1\,1\,1\,1 \\ 0\,0\,0\,1 \\ 1\,0\,0\,1 \\ 1\,1\,0\,1 \end{pmatrix} \tag{2}$$

Remark 1. When we compare our method with the Calkin's method, if $\overrightarrow{\mathbf{r}}_k$ and $\overleftarrow{\mathbf{r}}_{k+1}$ have positive entries, then $T_k = T$ for all $k = 1, \ldots, n$, and then T will be the classic transfer matrix used in [4].

In the case, not necessarily monotone, of a formula F having a constrained an oriented grid graph $G_{m,n}$ and transfer matrices $T_0, \ldots, T_{n-1}$, is straightforward to conclude that the sum of all entries of the product matrix $T_{n-1} \cdots T_0$ is the number of satisfying assignment of F. This fact is expressed in the following theorem.

Theorem 1. *Let F be a grid formula such that its constrained graph is an oriented grid graph $G_{m,n}$ $(1 \leq n)$, then the number of satisfying assignments of F is given by the sum of all of the entries of the product matrix $T_{n-1} \cdots T_0$, where T_k is the transfer matrix of $G_{m,k,k+1}$, $k = 0, \ldots, n-1$.*

Before detailing the proof, we consider the following example.
Example: Let $F = (x_0 \vee y_0) \wedge (\neg y_0 \vee \neg z_0) \wedge (z_0 \vee z_1) \wedge (z_1 \vee z_2) \wedge (z_2 \vee y_2) \wedge (y_2 \vee x_2) \wedge (x_2 \vee x_1) \wedge (\neg x_1 \vee x_0) \wedge (x_1 \vee y_1) \wedge (\neg y_1 \vee z_1) \wedge (\neg y_1 \vee \neg y_0) \wedge (y_1 \vee y_2)$. The constrained graph of F is the oriented grid graph $G_{2,2}$ with depicted in Figure

3. Then, from last example, T_0 and T_1 are the transfer matrices given in (2) and (3) respectively. Now, we have that the product matrix T_1T_0 is the following

$$T_1T_0 = \begin{pmatrix} 3\,2\,2\,0 \\ 4\,2\,3\,0 \\ 1\,0\,1\,0 \\ 2\,1\,2\,0 \\ 3\,1\,3\,0 \end{pmatrix}$$

therefore, $\#\mathrm{SAT}(F) = 30$.

If $F_{m,n}$ denotes a grid formula having as constrained graph a grid $G_{m,n}$, for $n > 0$, we can write

$$F_{m,n} = (\bigwedge_{i=0}^{n} C_i) \wedge (\bigwedge_{\ell=0}^{n-1} R_\ell) \tag{3}$$

where

$$C_i = \bigwedge_{k=0}^{m-1} (s'_{ki}x_{ki} \vee s_{k+1,i}x_{k+1,i}) \tag{4}$$

$s'_{ki}, s_{k+1,i} \in \{+, -\}$,

$$R_\ell = \bigwedge_{j=0}^{m} (r'_{j\ell}x_{j\ell} \vee r_{j,\ell+1}x_{j,\ell+1}) \tag{5}$$

$r'_{j\ell}, r_{j,\ell+1} \in \{+, -\}$. Here, the formulas C_i and R_ℓ are called *column-formula* and *row-formula* respectively. Notice that for $m, n > 0$

$$F_{m,n} = F_{m,n-1} \wedge C_n \wedge R_{n-1},\ F_{m,0} = C_0,\ F_{0,n} = R_0. \tag{6}$$

For $i = 0, ..., n-1$, we define

$$F_{m,i,i+1} = C_i \wedge C_{i+1} \wedge R_i \tag{7}$$

Note that

$$F_{m,n} = \bigwedge_{i=0}^{n-1} F_{m,i,i+1} \tag{8}$$

If $\phi : \{x_{0i}, \ldots, x_{mi}\} \to \{0, 1\}$ is an assignment of values for the variables of C_i (partial assignments of the variables of $F_{m,n}$), is denoted by the $(m+1)$-vector $(\phi(x_{0i}), ..., \phi(x_{mi}))$. Also, observe that can be considered as a partial assignment on the nodes of a k column induced vector from the oriented grid graph $G_{m,n}$.

$$\mathbf{x}_k = (x'_{k0}, x_{k1}, x'_{k1}, ..., x'_{k,m-1}, x_{km})$$

where $\phi(x_{ki}) = \phi(x'_{ki})$ for $i = 1, ..., m-1$. That is, an assignment for the variables of C_i can be seen as a vector in $\{0, 1\}^{2m}$.

To prove theorem 1, first, we characterize the partial assignments of the variables of $F_{m,n}$, such that satisfy each column-formula C_i (lemma 1). Second, we characterize the pairs of assignments that satisfy the formula (8), i.e. satisfy two consecutive column-formulas C_i, C_{i+1} and the respective row-formula R_i (lemma 2). Finally, we prove that all matrix of partial assignments derived from the lemmas 1 and 2, satisfies the formula $F_{m,n}$. Next, for simplicity, we omit the index i of $v_{ji}, x_{ji}, s_{ji}, r_{ji}, s'_{ji}$ and r'_{ji}.

Lemma 1. *The vector* $\mathbf{x} \in \{0,1\}^{2m}$ *satisfies the formula (4) if and only if* $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}}$, *where* $\overline{\mathbf{x}} = -\mathbf{x}$ *and* $\mathbf{s}$ *is the sign vector of* C_i.

Proof. If we assume that $\mathbf{x} = (x'_0, x_1, x'_1 ..., x'_{m-1} x_m)$ satisfies the formula (4) and $\mathbf{s} = (s'_0, s_1, s'_1, ..., s'_{m-1}, s_m)$, then $(s'_\ell x_\ell \vee s_{\ell+1} x_{\ell+1}) = 1$ for all $\ell \in \{0, ..., m-1\}$, that is equivalent to $(s'_\ell \overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1}) \neq (1,1)$.

It is straightforward to verify that the vector

$$(s'_0\overline{x}_0, s_1\overline{x}_1, s'_1\overline{x'}_1, ..., s'_{m-1}\overline{x}_{m-1}, s'_m\overline{x}_m) = \mathbf{s} \odot \overline{\mathbf{x}}$$

does have no two consecutive 1's, since any case is implied by conditions $\phi(x_{ki}) = \phi(x'_{ki})$ for $i = 1, ..., m-1$ and $(s'_\ell\overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1}) \neq (1,1)$. Therefore, $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}}$.

Suppose that $\mathbf{s} \odot \mathbf{x} \in \mathcal{C}_m$, for $\ell = 0, ..., m$ then $(s'_\ell\overline{x}_\ell, s_{\ell+1}\overline{x}_{\ell+1})$ does not have two consecutive $1's$. The vector $\mathbf{x}$ satisfies the column-formula C_i (equation (4)), otherwise, there is $\ell \in \{0, ..., m-1\}$ such that $s'_\ell x_\ell \vee s_{\ell+1}x_{\ell+1} = 0$, then $s'_\ell\overline{x}_\ell = 1$ and $s_{\ell+1}\overline{x}_{\ell+1} = 1$ (contradiction). □

Lemma 2. *The pair* $(\mathbf{x}, \mathbf{y}) \in \{0,1\}^{4m}$ *satisfies* $F_{m,i,i+1}$ *if and only if* $(\overline{\mathbf{x}}, \overline{\mathbf{y}}) \in \mathcal{F}_m^{\mathbf{s}_i} \times \mathcal{F}_m^{\mathbf{s}_{i+1}}$ *and* $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$.

Proof. Suppose that $\mathbf{x} = (x'_0, x_1, x'_1 ..., x'_{m-1}, x_m)$ and $\mathbf{y} = (y'_0, y_1, y'_1 ..., y'_{m-1}, y_m)$ are such that $(\mathbf{x}, \mathbf{y})$ satisfies $F_{m,i,i+1}$. From lemma 1, $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}_i}$ and $\overline{\mathbf{y}} \in \mathcal{F}_m^{\mathbf{s}_{i+1}}$, we must prove that $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$.

By hypothesis $(r'_{ij}x_j \vee r_{i+1,j}y_j) = 1$ for all $j = 0, ..., m$, then $r'_{ij}\overline{x}_j \wedge r_{i+1,j}\overline{y}_j = 0$ for all $j = 0, ..., m$, therefore $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$.

If $\overline{\mathbf{x}} \in \mathcal{F}_m^{\mathbf{s}_i}$ and $\overline{\mathbf{y}} \in \mathcal{F}_m^{\mathbf{s}_{i+1}}$, from lemma 1, $\mathbf{x}$ satisfies C_i and $\mathbf{y}$ satisfies C_{i+1}. Now, if $(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$, then $r'_{ij}\overline{x}_j \cdot r_{i+1,j}\overline{y}_j = 0$ for all $j = 0, ..., m$, hence $(r'_{ij}x_j \vee r_{i+1,j}y_j) = 1$ for all $j = 0, ..., m$. Therefore $(\mathbf{x}, \mathbf{y})$ satisfies the row-formula R_j (equation (5)) for $j = 0, ..., m$. □

Remark 2. Notice that if $\mathcal{F}_m^{\mathbf{s}_i} = \{\mathbf{x}_0^i, ..., \mathbf{x}_{r_i}^i\}$, then the sum of the entries of the matrix $T_i = (a_{kl}^i)_{r_{i+1} \times r_i}$ where $a_{kl}^i = 1$ if $(\overrightarrow{\mathbf{r}}_{i+1} \odot \mathbf{x}_k^{i+1}) \cdot (\overleftarrow{\mathbf{r}}_i \odot \mathbf{x}_l^i) = 0$ and $a_{kl}^i = 0$ otherwise, is $\#SAT(F_{m,i,i+1})$, that is, there is a bijection between the set of non zero entries of T_i and the set of satisfying assignments of $F_{m,i,i+1}$. Therefore, from previous lemma we have that $\mathbf{1}^t T_i \mathbf{1} = \#SAT(F_{m,i,i+1})$, where T_i is the transfer matrix of the column i to the column $i+1$ of $G_{m,n}$ (the constrained graph of $F_{m,n}$). Finally, we prove the theorem 1.

Proof (Theorem 1). From equation (8), it is clear that the vector $(\mathbf{x}_0, ..., \mathbf{x}_n)$ $\in \{0,1\}^{2m(n+1)}$ satisfies the formula $F_{m,n}$ if and only if $(\mathbf{x}_i, \mathbf{x}_{i+1})$ satisfies $F_{m,i,i+1}$ for $i = 0, ..., n-1$. By lemma 2, $(\bar{\mathbf{x}}_i, \bar{\mathbf{x}}_{i+1}) \in \mathcal{F}_m^{\mathbf{s}_i} \times \mathcal{F}_m^{\mathbf{s}_{i+1}}$ and

$$(\overleftarrow{\mathbf{r}}_i \odot \overline{\mathbf{x}}) \cdot (\overrightarrow{\mathbf{r}}_{i+1} \odot \overline{\mathbf{y}}) = 0$$

for $i = 0, ..., n-1$. Let $a^i_{l_{i+1}l_i}$ be the entry of the transfer matrix T_i in position

$$(\bar{\mathbf{x}}_{i+1}, \bar{\mathbf{x}}_i) \in \mathcal{F}_m^{\mathbf{s}_{i+1}} \times \mathcal{F}_m^{\mathbf{s}_i}$$

Then, by definition of T_i and previous analysis,

$$(\mathbf{x}_0, ..., \mathbf{x}_n) \in \{0,1\}^{(2m)(n+1)}$$

satisfies the formula $F_{m,n}$ if and only if

$$(\bar{\mathbf{x}}_0, ..., \bar{\mathbf{x}}_n) \in \mathcal{F}_m^{\mathbf{s}_0} \times \cdots \times \mathcal{F}_m^{\mathbf{s}_n} \text{ and } a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 1$$

Therefore $\#SAT(F_{m,n})$ is the cardinality of the set

$$\{(\bar{\mathbf{x}}_0, \cdots, \bar{\mathbf{x}}_n) \in \mathcal{F}_m^{\mathbf{s}_0} \times \cdots \times \mathcal{F}_m^{\mathbf{s}_n} : a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 1\}$$

Taking into account all the terms $a^{n-1}_{l_n l_{n-1}} \cdots a^0_{l_1 l_0} = 0$, we obtain

$$\#SAT(F_{m,n}) = \sum_{(l_0,\ldots,l_n)\in I_0\times\cdots\times I_n} a^{n-1}_{l_n l_{n-1}} \cdots a^1_{l_2 l_1} \cdot a^0_{l_1 l_0} = \mathbf{1}^t T_{n-1} \cdots T_0 \mathbf{1}$$

where $I_k = \{0, ..., t_k\}$, $t_k =\mid \mathcal{F}_m^{\mathbf{s}_k} \mid$ for $k = 0, ..., n$. □

In [2] we show a possible application of our method for modeling a distributed work among a team formed by 16 remote collaborators, and which is represented by a network grid. Some constraints are defined on the orientations which distribute the flow of tasks among the members of the team. Our method is applied for counting the number of different paths of the oriented grid where such paths violate the defined constraints.

5 Conclusions

We have considered the different pattern orientations of an undirected edge, and applying those pattern on a grid graph G, we develop a new matrix method for recognizing and counting the number of sink-free orientations of G for solving its related constraint satisfaction problem; to count the number of models of a 2-CNF on formulas whose constrained graph is a grid graph.

The type of orientations considered here generalize the class of problems which could be modeled and solved through methods applied in the area of Constraint Satisfaction Problems.

References

1. Baxter, R.J.: Planar Lattice Gases with Nearest-Neighbour Exclusion. Annals of Combinatorics 3, 191–203 (1999)
2. Guillén, C., Vera, E., López-López, A., De Ita, G.: Applying the Transfer Matrix Method for Supervising Lines in a Network Grid. In: Proc. REV 2008 (2008), www.rev-conference.org

3. Barbosa, V.C., Ferreira, R.G.: On the phase transitions of graph coloring and independent sets. Physica A: Statistical Mechanics and its Applications 343, 401–423 (2004)
4. Calkin, N.J.: The Number of Independent Sets in a Grid Graph. SIAM Journal on Discrete Mathematics 11(1), 54–60 (1998)
5. Gärtner, B., Morris, W., Rüst, L.: Unique Sink Orientations of Grids. Algorithmica 51(2), 200–235 (2008)
6. Golin, M.J., Leung, Y.C., Wang, Y., Yong, X.: Counting Structures in Grid Graphs, Cylinders and Tori Transfer Matrices: Survey and New Results. In: ALENEX/ANALCO, pp. 250–258 (2005)
7. Grohe, M., Marx, D.: Constraint Solving via Fractional Edge Covers. In: Soda'06, Miami (2006)
8. Reinhardt, E.: The Fibonacci Number of a Grid Graph and a New Class of Integer Sequences. JIS Journal of Integer Sequences 88(2), 1–16 (2005) Article 05.2.6
9. Roth, D.: On the Hardness of Approximate Reasoning. Artificial Intelligence, 273–302 (1996)
10. Roth, R.M., Siegel, P.H., Wolf, J.K.: Efficient Coding Schemes for the Hard-Square Model. IEEE Trans. Inform. Theory 47, 1166–1176 (2001)
11. Russ, B.: Randomized Algorithms: Approximation, Generation, and Counting, Distinguished dissertations. Springer, Heidelberg (2001)
12. Vadhan Salil, P.: The complexity of Counting in Sparse, Regular, and Planar Graphs. SIAM Journal on Computing 31(2), 398–427 (2001)
13. Valiant, L.G.: The complexity of enumeration and reliability problems. SIAM J. Comput. 8(3) (1979)

Sentence to Document Level Emotion Tagging – A Coarse-Grained Study on Bengali Blogs

Dipankar Das and Sivaji Bandyopadhyay

Department of Computer Science and Engineering, Jadavpur University, India
dipankar.dipnil2005@gmail.com, sivaji_cse_ju@yahoo.com

Abstract. This paper presents the identification of document level emotions from the sentential emotions obtained at word level granularity. Each of the Bengali blog documents consists of a topic and corresponding user comments. Sense weight based average scoring technique for assigning sentential emotion tag follows the word level emotion tagging using Support Vector Machine (SVM) approach. Cumulative summation of sentential emotion scores is assigned to each document considering the combinations of some heuristic features. An average F-Score of 59.32% with respect to all emotion classes is achieved on 95 documents on the development set by incorporating the best feature combination into account. Instead of assigning a single emotion tag to a document, each document is assigned with the best two emotion tags according to the ordered emotion scores obtained. The best two system assigned emotion tags of each document are compared against best two human annotated emotion tags. Evaluation of 110 test documents yields an average F-Score of 59.50% with respect to all emotion classes.

Keywords: Document, SVM, Emotion Tagging, Heuristic Features.

1 Introduction

From the classification of reviews [18] or newspaper articles [15] to Question Answering systems [6] and modern Information Retrieval systems [4], all the areas are increasingly incorporating emotion analysis within their scope. It sometimes important to track users' emotion expressed in online forums or blogs or twitters for different applications such as sentiment review, customer management, stock exchange prediction etc. Blogs are the communicative and informative repository of text based emotional contents in the Web 2.0 [15]. Researches on emotion show that blogs play the role of a substrate to analyze the reactions of different emotional enzymes. In general, sentence level emotion identification plays an important role to track emotions or to find out the cues for generating such emotions or to properly identify it. Sentences as the information units of any document identify the overall document level emotion whereas emotion of individual sentence in the documents is based on the emotions expressed by the word(s) [3, 16].

In the present task, each of the Bengali blog documents containing individual topic and corresponding user comments is collected from the web blog archive

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 332–341, 2010.

(www.amarblog.com). Each of the blog documents is annotated with Ekman's (1993) [17] six basic emotion tags. The method adopted for assigning sentential emotion tag is based on word level constituents follows the same approach as in [9]. But, instead of using Conditional Random Field (CRF) [2] based classifier, Support Vector Machine (SVM) [5] is used for word level emotion tagging in the present task. It is observed that SVM outperforms CRF in word level emotion tagging. The sense weight based average scoring technique [9] is applied for assigning sentential emotion tags based on word level emotion tagged constituents. The document level emotion tagging with emotion scores from emotion tagged sentences is carried out based on some combinations of heuristic features (e.g. emotion tag of the title sentence or end sentence of a topic, emotion tags assigned to an overall topic, most frequent emotion tags expressed in user comment portions of a document, identical emotions that appear in the longest series of tagged sentences etc.). The best two emotion tags are assigned to a document based on the ordered maximum emotion scores obtained. The development set gives the best average *F-Score* of 59.32% after applying all possible feature combinations. Evaluation is carried out against the best two annotated emotion tags of 110 test documents containing 1298 user comments sections. Average *F-Score* of 59.50% has been achieved with respect to all emotion classes.

The rest of the paper is organized as follows. Section 2 describes the related work done in this area. Corpus preparation followed by sentence level emotion tagging is discussed in Section 3. Document level emotion tagging is mentioned in Section 4. Section 5 describes subsequent emotion class wise evaluation with respect to the best two emotion tags. Finally Section 6 concludes the paper.

2 Related Work

Several efforts have been attempted by the natural language processing researchers to identify emotion at different level of granularities. In [16], work on opinions mining at word, sentence and document levels from news and web blog articles has been conducted along with opinion summarization. The present work is similar only in the sense that both of the works adopted the granular fashion to represent and solve the problem related to opinion or emotion. But tour approach differs in assigning document level emotion tags along with the knowledge of heuristic features.

In [13, 14], *Support Vector Machine* (SVM) based classifier was used on the blog data to classify the documents according to the mood of the author during writing. The authors used emoticons in LiveJournal posts to train a mood classifier at the document level. Another related article described in [15] has focused on the classification of news articles into the readers' emotions instead of the authors'. The work discussed in [8] used Yahoo! Kimo Blog as corpora to build emotion lexicons. In their studies, emoticons are used to identify emotions associated with textual keywords. In the present task, the perspective of reader is selected only to judge the emotional counterpart. The emoticons of Bengali blog documents are also considered in the present task.

The task described in [12] adopted different unsupervised, supervised and semi supervised strategies to identify and classify emotions. The researchers of [8] have done an experiment of the emotion classification task on web blog corpora using

SVM and CRF machine learning techniques. It is observed that the CRF classifiers outperform SVM classifiers in case of document level emotion detection. In contrast, the present task shows that SVM outperforms over CRF in word level emotion detection as CRF suffers from sequence labeling issues used for tagging emotion to discrete word tokens. However, most of the related work has been conducted for English. Bengali is less privileged and less computerized than English. Works on emotion analysis in Bengali have started recently [9, 10]. The present work differs in the respect that the knowledge extracted from different heuristic features and their combinations play a novel role in document level emotion tagging.

3 Word to Sentence Level Emotion Tagging

3.1 Corpus Preparation

As Bengali is very resource constraint language, the emotion annotated Bengali blog corpus [9] is being developed manually. A small portion has been considered in the present task. The blog documents stored in the format as shown in Figure 1 are retrieved from the web blog archive (www.amarblog.com). Each of the blog documents is assigned with a unique identifier (*docid*) followed by a section devoted for topic section and several sections devoted for different users' comments. Each of the sections of users' comment that is associated with corresponding user id (*uid*) may also contain the comments of other users. Such overlapped comments are differentiated using their corresponding user ids (*uid*).

```
-<DOC docid = xyz>
        -<Topic>…. </Topic>
        -<User Comments>
                 -<U uid=1>… </U>
                 -<U uid=2>… </U>
                 -<U uid=3>….
                       -<U uid=1>…</U> </U>…
      </User Comments>
</DOC>
```

Fig. 1. General structure of a blog document

Total of 24678 sentences are employed in the present task. Although we have considered 95 and 110 documents as our development and test sets respectively but 300 and 200 development and test sentences of [9] are used in the present task for conducting different experiments and comparative evaluations of sentential emotion tagging. Out of total 1200 sentences collected from 14 different comic related blog documents, rest 700 sentences are considered for SVM based training of the word level emotion tagging system.

3.2 Sentential Emotion Tagging

In the present task, the technique adopted for assigning sentential emotion tags is similar with the approach that was considered in [9]. But, instead of using Conditional Random Field (CRF), the Support Vector Machine (SVM) based classifier is used in the present task to classify each word into any of the Ekman's(1993) six emotion categories. Feature plays a crucial rule in any machine-learning framework. Hence, among 10 active singleton features of [9], 9 features have been employed to accomplish the current task. Instead of *SentiWordNet* emotion word, the *Bengali WordNet Affect* [11] has been used in this task to identify feature for emotion words. Different unigram and bi-gram context features (word level as well as POS tag level) have been applied. The features are as follows.

- POS information (adjective, verb, noun, adverb)
- First sentence in a topic or title sentence [7]
- Emotion words of Bengali WordNet Affect (e.g. ক্ষ্যামা(khyama) [pardon])
- Reduplication (e.g., *bhallo bhallo* [good good], *khokhono khokhono* [when when] etc.)
- Question words (*ki* [what], *keno* [why] etc.)
- Colloquial / Foreign words (e.g., *kshyama* [pardon] etc.) and foreign words (e.g. Thanks, *gossya* [anger] etc.)
- Special punctuation symbols (!,@,?..)
- Quoted sentence ("you are 2 good man")
- Sentence Length (>=8, <15)
- Emoticons (☺, 😐, ☹ ..)

The CRF classifier generally performs the classification task for sequence labeling problem, and thus it carries out word level classification task with a significant loss of word level emotional constituents. As SVM gives better performance in discrete (e.g. word) information tagging, the improvement is found in the word level emotion classification task. The comparative results of CRF and SVM for word level emotion tagging are shown in Table 1. The improvement of word level emotion tagging system also carries its effects to the sentential emotion tagging.

Table 1. Word level emotion tagging accuracies (%) for six emotion classes on the Development and Test set using CRF and SVM

Emotion Classes	Development Set (# words)	Test Set (#words)	
		CRF-based model	**SVM-based model**
Happy	61.17 (185)	67.67 (106)	69.55 (106)
Sad	64.63 (151)	63.12 (143)	65.34 (143)
Anger	62.85 (136)	51.00 (70)	56.15 (70)
Disgust	68.66 (100)	49.75 (65)	53.35 (65)
Fear	63.38 (96)	52.46 (37)	54.78 (37)
Surprise	79.05 (230)	68.23 (204)	69.37 (204)

The default emotion tag weights for six emotion types are taken from [9, 10]. In the method, the basic six words "*happy*", "*sad*", "*anger*", "*disgust*", "*fear*" and "*surprise*" are chosen as the seed words corresponding to each emotion type. The positive and negative scores in the English SentiWordNet [1] for each synset in which each of these seed words appear are retrieved and the average of the scores is fixed as the *Sense_Tag_Weight* (STW) of that particular emotion tag.

The present work differs from the approach in [9] while assigning emotion scores and sentential emotion tags to the blog sentences. In the present method, the emotion tagged words have been considered instead of depending on the fixed emotion tag weights assigned for the words. For supporting the task, a Bengali *SentiWordNet* is being developed by replacing each word entry in the synonymous set of the English *SentiWordNet* by its possible set of Bengali synsets using a synset based English to Bengali bilingual dictionary being developed as part of the EILMT[1] project.

Each word tagged with a particular emotion type is searched in the Bengali *SentiWordNet* and the positive and negative scores of the word are retrieved from the *SentiWordNet*. The average of the scores is fixed as the *Sense_Tag_Weight* (STW) for the emotion tag assigned to the word. If an emotion tagged word is not found in the Bengali *SentiWordNet*, the default weight calculated earlier is assigned for that word. The total *Sense_Tag_Weight* (STW) for each emotion tag *i* is calculated by summing up the STWs of all assigned emotion tags with type *i*. Stemming is included during the searching process. Bengali, like any other Indian languages, is morphologically very rich. Different suffixes may be attached to a word depending on the various features (e.g. the features for a Bengali verb are Tense, Aspect, and Person). A Bengali stemmer uses a suffix list to identify the stem form of the word.

Apart from the search technique, the sentential emotion tagging is carried out in a similar fashion that was adopted by [9]. Each sentence is assigned with a *Sense_Weight_Score* (SWS) for each emotion type. The weight is calculated by dividing the total STW of all occurrences of that emotion tag in the sentence by the total STW of all types of emotion tags present in that sentence. The sentence is assigned with the emotion tags for which the sentence level *Sense_Weight_Score* (SWS) is highest. The sentences are tagged as *neutral* type if for all emotion tags, the total *Sense_Weight_Scores* (SWS) produce zero (0) emotion score. The post processing strategies [9] related to negative words have been incorporated in the present system. The comparative results of the CRF based model with SVM for sentence level emotion tagging is shown in Table 2.

Table 2. CRF and SVM based test set accuracies (in %) per emotion class for sentential emotion tagging

Emotion Class (Total # sentences)	CRF-based model	SVM- based model
happy (40)	65.28	66.05
sad (41)	66.42	68.12
angry (32)	60.28	62.77
disgust (21)	52.18	53.54
fear (23)	57.14	60.11
surprise (43)	66.45	69.82

[1] English to Indian Languages Machine Translation (EILMT) is a TDIL project undertaken by the consortium of different premier institutes and sponsored by MCIT, Govt. of India.

4 Document Level Emotion Tagging

Assigning a single emotion tag to a particular document does not always bear the actual emotions present in that document. This module identifies document level emotion tags and their associated weights based on the sentence level emotion tags along with the contribution of some heuristic feature combinations. *Emotion_Weight_Score* (EWS) based technique applied on the sentence level emotion tags produces six possible emotion scores for a document. But, the evaluation is carried out for the best two emotion tags only.

4.1 Calculation of Document Level Emotion Tag Weights

Each document is assigned with an *Emotion_Weight_Score* (EWS) for each of the six emotion tags. In general, the document level *Emotion_Weight_Score* (EWS) for a particular emotion tag is calculated by summing up the total *Sense_Weight_Scores* (SWS) of all occurrences of the sentential tags corresponding to that emotion category, i.e., $\text{EWS}i = \sum \text{SWS}i$, where SWS*i* is the sentence level *Sense_Weight_Score* (SWS) for the emotion tag *i* in the document. Each document is assigned with two document emotion tags DET*i* and DET*j*, for which EWS*i* is the highest and EWS*j* is the second highest *Emotion_Weight_Score*. $\text{SET}i = [\text{Max } i\text{=1 to 6}(\text{EWS}i)]$ and $\text{SET}j = [\text{Max } j\text{=1 to 6 } \&\& \ j \neq i \ (\text{EWS}j)]$. But, the document level emotion tagging incorporates the heuristic features and their combinations into consideration.

4.3 Heuristic Features

Document level emotion identification depends not only on the emotion expressed in the sentential constituents but also on the combination of different characteristic features of that document (e.g. the blog structure). Irrespective of linguistic attributes alone, blog itself contains some special inherent features that help in identifying emotion at sentence as well as document level. In the present task, the following 7 active features identified heuristically for the document level emotion tagging.

(I) Emotion tags of the title sentence (95, 110)
(II) Emotion tags of the end sentence of a topic (95, 110)
(III) Emotion tags assigned to an overall topic (95, 110)
(IV) Emotion tags for user comment portions of a document (1156, 1298)
(V) Most frequent emotion tags identified from the document (95, 110)
(VI) Identical emotions that appear in the longest series of tagged sentences (67, 61) (Yang et al., 2007)
(VII) Emotion tags of the largest section among all of the user comments' sections (1274, 1322)

The numeric figures in brackets after each heuristic feature denote the number of times the corresponding feature has appeared in the development and test set respectively. Each of the documents contains a title and a topic and hence the frequencies of the first three features are same as the number of documents. The development and test documents contain a total of 1156 and 1298 number of sections specified for user comments only. The emotions reflected inside user comment sections are helpful for predicting the emotions at document level. The emphasis is also given to the frequency of the emotion tags identified at document level. Sometimes comments portion of a user contain other user comments in a nested fashion. Hence, one or more comment sections contain a large number of emotional sentences than other sections. This feature enhances the continuity of emotion in document for predicting the overall emotions expressed in the document. The contributions of the features alone and in combination with other features have been evaluated on 95 developmental documents. Information related to the frequency of different features is shown in Table 3 on basis of importance. It has been observed that the emotion classes e.g. *fear*, *disgust* contain less frequent information regarding features I, II and V.

Table 3. Frequencies of seven features per emotion class in the development and test sets respectively

Emotion	Information (Feature id, #Frequency)	
Classes	Development Set	Test Set
happy	(I, 24), (II, 22), (III, 27), **(IV, 201)**, (V, 43), (VI, 9), (VII, 287)	(I, 28), (II, 26), (III, 31), **(IV, 219)**, (V, 54), (VI, 12), (VII, 308)
sad	(I, 29), (II, 22), (III, 28), **(IV, 177)**, (V, 32), (VI, 12), (VII, 264)	(I, 23), (II, 25), (III, 33), **(IV, 203)**, (V, 49), (VI, 17), (VII, 288)
angry	(I, 22), (II, 23), (III, 21), **(IV, 243)**, (V, 36), (VI, 11), (VII, 253)	(I, 27), (II, 29), (III, 27), **(IV, 278)**, (V, 43), (VI, 12), (VII, 276)
disgust	(I, 8), (II, 12), (III, 11), **(IV, 122)**, (V, 21), (VI, 10), (VII, 64)	(I, 19), (II, 21), (III, 16), **(IV, 145)**, (V, 32), (VI, 4), (VII, 79)
fear	(I, 10), (II, 8), (III, 15), **(IV, 110)**, (V, 27), (VI, 12), (VII, 44)	(I, 17), (II, 11), (III, 12), **(IV, 134)**, (V, 31), (VI, 5), (VII, 67)
surprise	(I, 28), (II, 37), (III, 26), **(IV, 215)**, (V, 51), (VI, 13), (VII, 362)	(I, 21), (II, 24), (III, 23), **(IV, 218)**, (V, 56), (VI, 11), (VII, 304)

5 Evaluation

Emotion tags are assigned to each document by counting the total number of sentential emotion tags and summing up the associated average weight scores. The best two tags are assigned to the document based on the maximum and next to maximum emotion scores obtained. The *F-score* value is calculated on the development set for each of the features and individual contribution of each feature is measured. It is found that the contribution of each feature is not uniform and can be fairly distinguished according to the level of importance. For example, the combination of topic as well as user comments is identified as a contributory feature pair denoted by the experiment id *ii (8)*. The contributions of all the features are not mentioned but the detailed experimental evaluation and associated results regarding features are shown in Table 4. The emotion tags corresponding to maximum and next to maximum *Emotion_Weight_Scores* (EWS) of a document are considered as the probable candidate emotion tags. The set, namely, **GSDT** (Gold Standard Document Tag) contains two emotion tags assigned to a document in the gold standard annotated corpus and is defined as {*dmax1, dmax2*}. Document level emotion tagging module has generated the set **SGDT** (System Generated Document Tag) that contains two probable candidate emotion tags to a document based on their ordered *Emotion_Weight_Scores* (EWS) and the set is described as {*dmax1′, dmax2′*}. The emotion tags *dmax1*and *dmax2* correspond to the maximum and next to maximum scores for two emotion tags assigned to a document in the gold standard corpus and *dmax1′* and *dmax2′* are the maximum and next to maximum scores for two emotion tags to a document generated by the system.

Table 4. F-scores (in %) of different heuristic features and their combinations on development set

Expt. Id.	Current Feature Set	F-Score (in %)
i	(1) Emotion tags of the title sentence	31.12
	(2) Emotion tags of the end sentence of a topic	28.25
	(3) Emotion tags assigned to an overall topic	48.87
	(4) Emotion tags for user comment portions of a document	52.66
	(5) Most frequent emotion tags identified from the document	53.95
	(6) Identical emotions that appear in the longest series of tagged sentences	37.29
	(7) Emotion tags of the largest section among all of the user comments' sections.	35.11
ii	(8). i(3)+i(4)	57.32
	(9). i(3)+i(5)	56.55
	(10). i(3)+i(7)	55.42
	(11). i(4)+i(5)	54.87
	(12). i(4)+i(6)	53.25
	(13). i(4)+i(7)	55.57
	(14). ii(8)+i(6)	58.54
iii	(15). ii(8)+i(7)	58.04
	(16). ii(11)+i(6)	56.21
	(17). ii(11)+i(7)	56.55
	(18). iii(14)+i(5)	59.32
iv	(19). iii(13)+i(5)	58.70
	(20). iii(15)+iii(16)	58.02

The *F-Score* for each emotion tag pair is measured by considering the number of system generated document tags that are matched correctly with annotated tags. The final average *F-Score* is calculated for each emotion class considering any four combinations of the two sets. The tagged documents are evaluated against the manually annotated 95 gold standard documents of the development set and then applied on 110 test documents. It is observed that 59.32% *F-score* has been achieved with these four combinations on a development set. The corresponding feature combination that gives best *F-Score* on the development set is applied for 110 test documents and finally an average *F-Score* of 59.50% is achieved. It is also observed that the emotions expressed in the title of the document do not convey the actual emotions expresses inside the document. Hence, the coarse grained evaluation based on the heuristic features helps to predict the document level emotions extensively. An important observation is that, as the number of feature instances varies in the emotion classes, they have imposed an impact on the document level tagging for that emotion classes respectively. It has to be mentioned that the performance of the system in terms of *F-Score* has not improved significantly by conducting the extended evaluation with adding of more than two tags in GSDT and SGDT sets.

Table 5. Evaluation of Document Level Emotion Tagging

Tag Combination of GSDT and SGDT		**Test Documents (#110)**	
		F-Score	**Average F-Score (in %)**
	{d*max1* , *dmax1′*}	61.23	
happy	{*dmax1* , *dmax2′* }	58.11	
	{*dmax2* , *dmax1′* }	57.08	**58.74**
	{*dmax2* , *dmax2′* }	58.56	
	{d*max1* , *dmax1′*}	60.98	
sad	{*dmax1* , *dmax2′* }	61.08	
	{*dmax2* , *dmax1′* }	59.77	**60.78**
	{*dmax2* , *dmax2′* }	61.32	
	{d*max1* , *dmax1′*}	61.57	
angry	{*dmax1* , *dmax2′* }	59.22	
	{*dmax2* , *dmax1′* }	59.69	**60.25**
	{*dmax2* , *dmax2′* }	60.54	
	{d*max1* , *dmax1′*}	57.87	
disgust	{*dmax1* , *dmax2′* }	58.06	
	{*dmax2* , *dmax1′* }	58.17	**58.40**
	{*dmax2* , *dmax2′* }	59.51	
	{d*max1* , *dmax1′*}	57.34	
fear	{*dmax1* , *dmax2′* }	57.81	
	{*dmax2* , *dmax1′* }	59.37	**58.19**
	{*dmax2* , *dmax2′* }	58.25	
	{d*max1* , *dmax1′*}	60.33	
surprise	{*dmax1* , *dmax2′* }	60.81	
	{*dmax2* , *dmax1′* }	61.37	**60.69**
	{*dmax2* , *dmax2′* }	60.25	

6 Conclusion

In the present task, the sentence level emotion tags are used to assign document level emotion tags. The resulting document level emotion tagger can be used in an emotion based information retrieval system where retrieved documents will match the user defined query word(s) and emotion specification. The idea of assigning two emotion tags to the documents can then be related to the ranking of the retrieved sentences and documents. Emotion analysis related to the effect of metaphors (especially in blogs) is the research area to be explored in future. But the clause level analysis of the complex emotional sentences may be an area for further studies.

References

1. Esuli, A., Sebastiani, F.: SENTIWORDNET: A Publicly Available Lexical Resource for Opinion Mining. In: LREC-06 (2006)
2. McCallum, A., Pereira, F., Lafferty, J.: Conditional Random Fields: Probabilistic Models for Segmenting and labeling Sequence Data. ISBN, 282–289 (2001)
3. Vincent, B., Xu, L., Chesley, P., Srhari, R.K.: Using verbs and adjectives to automatically classify blog sentiment. In: Proceedings of AAAI-CAAW-06, the Spring Symposia (2006)
4. Pang, B., Lee, L.: Foundations and Trends in Information Retrieval 2, vol. 1-2, pp. 1–135 (2008)
5. Cortes, C., Vapnik, V.: Support-Vector Network. Machine Learning 20, 273–297 (1995)
6. Claire, C., Janyce, W., Theresa, W., Litman Diane, J.: Combining Low-Level and Summary Representations of Opinions for Multi-Perspective Question Answering. New Directions in Question Answering, 20–27 (2003)
7. Alm, C.O., Roth, D., Sproat, R.: Emotions from text: machine learning for text-based emotion prediction. In: Proceedings of the Conference on HLT (EMNP) Vancouver, British Columbia, Canada, pp. 579–586 (2005)
8. Yang, C., Lin, K.H.-Y., Chen, H.-H.: Emotion classification Using Web Blog Corpora. In: IEEE, WIC, ACM International Conference on Web Intelligence, pp. 275–278 (2007)
9. Das, D., Bandyopadhyay, S.: Emotion Tagging – A Comparative Study on Bengali and English Blogs. In: ICON-09, India, pp. 177–184 (2009)
10. Das, D., Bandyopadhyay, S.: Word to Sentence Level Emotion Tagging for Bengali Blogs. In: ACL-IJCNLP 2009, Suntec, Singapore, pp. 149–152 (2009)
11. Das, D., Bandyopadhyay, S.: Developing Bengali WordNet Affect for Analyzing Emotion. In: ICCPOL-2010, California, USA (2010)
12. Sebastiani, F.: Machine learning in automated text categorization. ACM Computing Surveys 34(1) (2002)
13. Mishne, G.: Experiments with Mood Classification in Blog Posts. In: Proceedings of 1st Workshop on Stylistic Analysis of Text for Information Access (2005)
14. Mishne, G., de Rijke, M.: Capturing Global Mood Levels using Blog Posts. In: Proceedings of AAAI, Spring Symposium on Computational Approaches to Analysing Weblogs, pp. 145–152 (2006)
15. Lin, K.H.-Y., Yang, C., Chen, H.-H.: What Emotions News Articles Trigger in Their Readers? In: Proceedings of SIGIR, pp. 733–734 (2007)
16. Ku, L.-W., Liang, Y.-T., Chen, H.-H.: Opinion extraction, summarization and tracking in news and blog corpora. In: AAAI-2006 Spring Symposium on Computational Approaches to Analyzing Weblogs, AAAI Technical Report, pp. 100–107 (2006)
17. Ekman, P.: Facial expression and emotion. American Psychologist 48(4), 384–392 (1993)
18. Turney, P.D.: Thumbs Up or Thumbs Down? Semantic Orientation Applied to Unsupervised Classification of Reviews. In: Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, pp. 417–424 (2002)

Clustering Weblogs on the Basis of a Topic Detection Method

Fernando Perez-Tellez[1], David Pinto[2], John Cardiff[1], and Paolo Rosso[3]

[1] Social Media Research Group, Institute of Technology Tallaght Dublin, Ireland
fernandopt@gmail.com, John.Cardiff@ittdublin.ie
[2] Benemérita Universidad Autónoma de Puebla, Mexico
dpinto@cs.buap.mx
[3] Natural Language Engineering Lab, ELiRF, Universidad Pólitecnica de Valencia, Spain
prosso@dsic.upv.es

Abstract. In recent years we have seen a vast increase in the volume of information published on weblog sites and also the creation of new web technologies where people discuss actual events. The need for automatic tools to organize this massive amount of information is clear, but the particular characteristics of weblogs such as shortness and overlapping vocabulary make this task difficult. In this work, we present a novel methodology to cluster weblog posts according to the topics discussed therein. This methodology is based on a generative probabilistic model in conjunction with a Self-Term Expansion methodology. We present our results which demonstrate a considerable improvement over the baseline.

Keywords: Clustering, Weblogs, Topic Detection.

1 Introduction

In recent years the World Wide Web has shown huge changes as a tool of socialization, bringing up new services and applications such as weblogs, wikis as part of the Web 2.0 technologies. The blogosphere is a new medium of expression, becoming more popular all around the world. We can find weblogs in all subjects from sports, games to politics and finance.

In order to manage the large amount of information published in the blogosphere, there is a clear need for systems that provide automatic organization of its content, in order to exploit the information more efficiently and retrieve only the information required for a particular user. Document clustering –the assignment of documents to previously unknown categories— has been used for this purpose [20]. We consider it more appropriate to employ clustering rather than classification, since the latter would require providing tags of categories in advance and in real scenarios we usually deal with information from the blogosphere without knowing the correct category tag.

The focus of this research work is to study a novel approach for clustering weblog posts according to their topics of discussion. For this purpose, we have based our approach in a topic detection method. Topic detection and tracking is a well-studied

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 342–351, 2010.

area [2] [3], which focuses on extraction of significant topics and events from news articles. We consider the topic detection task as the problem of finding the most prominent topics in a collection of documents; in general terms, identifying a set of words that constitute topics in a collection of documents.

The main contribution in this work is a novel methodology of clustering weblog posts based on a topic detection model for text in conjunction with a Self-Term Expansion methodology [16]. In our approach we treat the weblog content purely as raw text, identifying the different topics inside of the documents and using this information in the clustering process.

In [15], the features of weblogs are discussed, for instance, weblogs can be characterized as very short texts and with a general writing style. These are undesirable characteristics from a clustering perspective, as not enough discriminative information is provided. In order to tackle the particular characteristics of weblogs, we employ an expansion methodology, the Self-Term Expansion Methodology [16], that does not use external resources, relying only on information included in the corpus itself then. Our hypothesis states that the application of this methodology can improve the quality of topic clusters, and further that the improvement will be more significant where the corpus is composed of well-delimited categories which share a low percentage of vocabulary (wide domain corpus).

The methodology we present consists of four parts. Firstly, it improves the representation of the text by means of a Self-Term Enriching Technique. External resources are not employed because we consider it difficult to identify appropriate linguistic resources for information such weblogs. Secondly, a Term Selection Technique is applied in order to select the most important and discriminative information of each category thereby reducing processing time for the next two steps. The third step is the use of the Latent Dirichlet Allocation method [5], which is a generative probabilistic model for discrete data. We use this model to construct a set of reference vectors which can be used as categories prototypes for a better and faster clustering process. Finally, we use the well-known Jaccard coefficient [14] as a similarity measure to form the clusters.

The rest of this paper is organized as follows. Section 2 presents the related work. Section 3 describes the dataset used in the experiments. Section 4 explains our approach and the techniques used in our research work. Section 5 shows the obtained results. Section 6 provides an analysis of results and, finally, in Section 7 we present the conclusions.

2 Related Work

There are previous attempts on topic detection in online documents such as in [8], where the authors present a topic detection system composed of three modules that attempt to model events and reportage in news. The first module (pre-processing) is used to select and weight the features, i.e., words that are representative of short events. The clustering module is a hybrid technique that uses a slow accurate hierarchical method with a fast partitional algorithm. Finally, the last module is the presentation module which displays each cluster to the user.

The task of finding a set of topic in a collection of documents has also been attempted in [21]; the authors based their approach on the identification of clusters of keywords that are taken as representation of topics. They have employed the well-known k-means algorithm to test some distance measures based on a distribution of words. The experiments were conducted using Wikipedia articles, reporting acceptable results, but the calculation of the distributions seems to be computational expensive.

Topic detection is also addressed in [18], where the authors present a method which uses blogger's interests in order to extract topic words from weblogs. In this approach the authors assume that topic words are words commonly used by bloggers who share the same interests, and they use these topic words to compute similar interests between each two bloggers by using the cosine similarity measure. A topic score is assigned to each word. The processing time is also a problem in this approach, as they have pointed out, and the optimization for some of their calculations is needed.

Recently, the clustering of weblogs has become an active topic of research; for instance in [13] the authors build a word-page matrix by downloading weblog pages and have applied the k-means clustering algorithm with different weights assigned to the title, body, and comment parts. In [1], the authors use weblog categories to build a category relation graph in order to join different categories; they use edges in the category relation graph to represent similarity between different categories and they represent nodes as categories. They also consider different values of link strengths and level of directories.

Our approach is focused on detecting the topic clusters contained in the corpus itself, and the novel aspect is based on using a topic detection method to identify possible references that could be used in the clustering process, and the expansion methodology in order to improve the representation of the weblogs.

3 Description of Dataset

In this section, we describe the corpus used in our experiments. The corpus is a subset of the ICWSM 2009 Spinn3r Blog Dataset[1], the content of the data includes metadata such as the blog's homepage, timestamps, etc. The data is in XML format and according to the Spinn3r crawling[2] documentation; it is further arranged into tiers, approximating search engine ranking to some degree.

Even if the Spinn3r blog dataset contains several blogs sites in a number of different languages, we only focused the experiments carried out on the "Yahoo Answers", weblog site[3] – in which people share what they know and ask questions on any topic that matters to the user, in order to be answered by other users. We have extracted from this corpus two distinct subsets (see Fig. 1). The first subset contains 10 categories with 25,596 posts and vocabulary size of 66,729. It may be considered as "narrow domain", since the vocabulary in the categories is similar. The second

[1] The corpus was initially made available for the 2009 Data Challenge at the 3rd International AAAI Conference on Weblogs and Social Media, `http://www.icwsm.org/2009/data/`

[2] `http://spinn3r.com/documentation/`

[3] `http://answers.yahoo.com/`

subset contains 10 categories with 48,477 posts and a vocabulary size of 122,960 terms. As opposed to the narrow domain subset, it may be considered "wide domain" because its categories have a low overlapping vocabulary.

	Category name	Posts	Category name	Posts
Subset 1 (Narrow Domain)	Cell_Phones_Plans	1,543	Video_Online_Games	6,578
	Computer_Networking	1,337	Maintenance_Repairs	1,973
	Programming_Design	2,466	Security	1,583
	Laptops_Notebooks	2,153	Music_Music_Players	1,640
	Software	4,800	Other_-_Internet	1,523
Subset 2 (Wide Domain)	Singles_Dating	20,498	Celebrities	2,219
	Software	4,800	Marriage_Divorce	2,956
	Womens_Health	4,262	Languages	1,914
	Politics	2,527	Elections	3,628
	Dogs	3,205	Books_Authors	2,468

Fig. 1. Topics of discussion of the two datasets (narrow and wide domain)

Clustering of narrow domains brings additional challenges to the clustering process. Moreover, the shortness of this kind of data will make this task more difficult. The purpose of constructing two subsets with these characteristics is to demonstrate the effectiveness of our method across both wide and narrow domains, and also to test the relative effectiveness of the approach in each case.

Regarding the categories tags, they were only used for gold standard construction purposes, and provide a better idea of the subsets used in our experiments. The posts are treated as raw text, i.e. we have not used any additional information provided by the XML tags. As a preprocessing step, we have removed stop words –high-frequency word that has not significant meaning in a phrase– and punctuation symbols as well.

4 Methodology Proposed

In this section, we present the techniques used in our approach in order to improve the quality of clusters. This methodology clusters weblog posts using prototypes as reference, therefore, we have also called this approach prototype/topic based clustering. Our approach is composed of three steps: the Self-Term Expansion Methodology (S-TEM), which consists of a Self-Term Enriching Technique and a Term Selection Technique. This is followed by the application of the Latent Dirichlet Allocation model and the prototype/topic based clustering process.

4.1 Self-Term Expansion Methodology

The Self-Term Expansion Methodology [16] comprises a twofold process: the Self-Term Enriching Technique, which is a process of replacing terms with a set of co-related terms, and a Term Selection Technique with the role of identifying the relevant features. The idea behind Term Expansion has been studied in previous works such as [17] and [9] in which external resources have been employed. Term expansion has been used in many areas of natural language processing as in word disambiguation in [4], in which WordNet [7] is used in order to expand all the senses of a word. However, in the particular case of the S-TEM methodology, we use only the information being clustered to perform the term expansion, i.e., no external resource is employed.

The technique consists of replacing terms of a web post with a set of co-related terms. We consider it particularly important to use the intrinsic information of the data set itself. A co-occurrence list is calculated from the target dataset by applying the Pointwise Mutual Information (*PMI*) [14]. *PMI* provides a value of relationship between two words; however, the level of this relationship must be empirically adjusted for each task. In this work, we found *PMI* equal or greater than 3 to be the best threshold. This threshold was established empirically. In other experiments [16], a threshold of 6 was used; however, in weblog documents correlated terms are rarely found. This list will be used to expand every term of the original corpus.

The Self-Term Enriching Technique is defined formally in [16] as follows: Let $D = \{d_1, d_2, \ldots , d_n\}$ be a document collection with vocabulary $V(D)$. Let us consider a subset of $V(D) \times V(D)$ of co-related terms as $RT = \{(t_i, t_j) | t_i, t_j \in V(D)\}$ The RT expansion of D is $D' = \{d'_1, d'_2, \ldots , d'_n\}$, such that for all $d_i \in D$, it satisfies two properties: 1) if $t_j \in d_i$ then $t_j \in d'_i$, and 2) if $t_j \in d_i$ then $t'_j \in d'_i$, with $(t_j , t'_j) \in RT$. If RT is calculated by using the same target dataset, then we say that D' is the Self-Term Expansion version of D. The degree of co-occurrence between a pair of terms is determined by a co-ocurrence method, this method is based on the assumption that two words are semantically similar if they occur in similar contexts [10].

The Term Selection Technique helps us to identify the best features for the clustering process. However, it is also useful to reduce the computing time of the clustering algorithms. In particular, we have used Document Frequency (DF) [19], which assigns the value DF(t) to each term t, where DF(t) means the number of posts in a collection, where t occurs. The Document Frequency technique assumes that low frequency terms will rarely appear in other documents; therefore, they will not have significance on the prediction of the class of a document.

4.2 Latent Dirichlet Allocation Model

In general, a topic model is a hierarchical Bayesian model that associates each document to a probability distribution over topics. The *Latent Dirichlet Allocation* (LDA) model [5] is derived from the idea of discovering short descriptions of the members of a collection, in particular discrete data, in order to allow efficient processing of huge collections, while keeping the essential statistical relationships that may be used in other tasks such as classification.

There are other sophisticated approaches that use dimensionality reduction techniques such as *Latent Semantic Indexing* (LSI) [6], which can achieve significant compression in large corpora using single value decomposition of the X matrix to identify a linear subspace in the space of *tf-idf* features by capturing most of the variance in the corpora. An alternative model is *probabilistic Latent Semantic Index* (pLSI) [11], in which the main idea is to model each word in a document as a sample from a mixture model, in which the components of the mixture are multinomial random variables that can be viewed as words generated from topics. However, LDA may be seen as a step forward with respect to LSI and pLSI.

The LDA model is based on a supposition that the words of each document arise from a mixture of topics, each of that is a distribution over the vocabulary. This method has been used for automatically extracting the topical structure of large document collections, in other words, it is a generative probabilistic model of a corpus that uses different distributions over a vocabulary in order to describe the document collection.

4.3 Clustering Weblog Posts Using the Prototypes as References

The prototype/topic based clustering methodology is outlined in Fig. 2. We start from having the corpus as raw text. Then we apply the S-TEM approach to the original posts. In the Term Selection Technique we have selected from 10% to 90% of vocabulary after the enriching process, in order to confirm which percentage provides the best information to LDA Method.

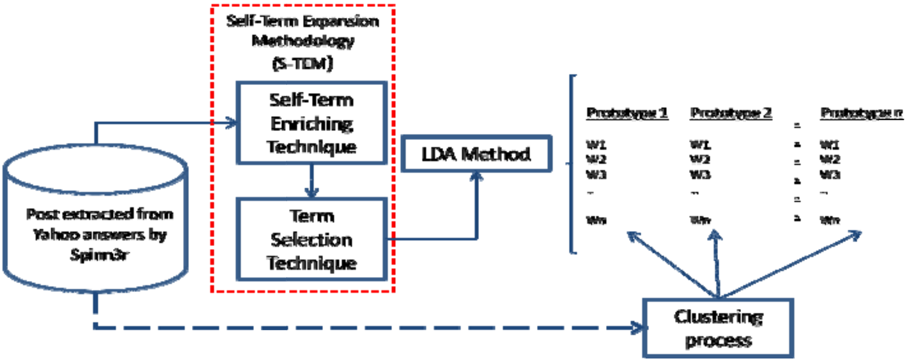

Self-Term Expansion Methodology (S-TEM)
Self-Term Enriching Technique
Term Selection Technique
LDA Method
Post extracted from Yahoo answers by Spinn3r
Prototype 1
Prototype 2
Prototype n
W1
W2
W3
Wn
W1
W2
W3
Wn
W1
W2
W3
Wn
Clustering process

Fig. 2. Methodology proposed "prototype/topic based clustering"

The LDA method generates the prototypes, i.e., vectors that will contain topics discussed on the posts. We expect to have a reference for each category in order to generate the clusters, one for each prototype. In this step, LDA requires as input the number of possible topics, in our case we have fixed this parameter to ten, which is the number of categories in each subset. We have also varied the number of terms selected from 100 to 3,000 in order to confirm the best and minimum number of terms for the clustering task.

Finally, the clustering process will compare each original post (unexpanded) with each prototype; every post will be assigned to one cluster according to the most similar prototype (highest value in the clustering process). We have chosen the Jaccard coefficient because its simplicity and relative fast clustering process. In our case, we have compared each original post against each prototype and the highest similarity measure with the prototypes get the post in its cluster.

5 Experiments

In this section, we present the experiments and results using the approach proposed in this research work. These experiments were carried out over the two subsets described in Section 3.

5.1 Wide Domain Subset

Fig. 3 presents a comparison of our approach against the baseline for the wide domain corpus. We have obtained the baseline by generating the prototypes with the LDA method from the original posts, i.e., without using the S-TEM methodology in the construction of the prototypes, and finally, clustering the posts with the Jaccard coefficient. We have summarized the results in the graph showing the minimum, maximum and average F-measure value obtained from the different percentage of vocabulary selected (from 10% to 90% with steps of 10%) with the Term Selection Technique in the S-TEM methodology.

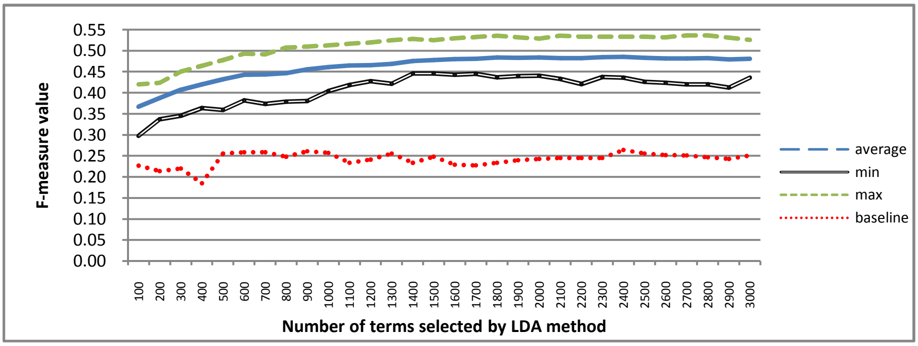


Fig. 3. Clustering results using the "wide" domain corpus

The objective of using this selection is to reduce the noise (terms included in more than one category that can be highly correlated with discriminative information) generated by the enriching technique and to highlight the most important features of each category. We have obtained the best results when we have selected 10% of vocabulary (achieving an F-measure value of 0.53). It means that after the enriching process, it only needs 10% of the vocabulary to generate the best prototypes. We have also confirmed that in all the cases we have outperformed the baseline (0.26 in the best case). We have limited the number of terms selected by the LDA method from 100 to 3,000 terms per topic in order to confirm the minimum number of terms for the prototype which can give us acceptable results in the clustering process. Furthermore, by reducing the number of terms, we can reduce the processing time for the clustering task.

5.2 Narrow Domain Subset

In Fig. 4 we present the improvement that the S-TEM methodology provides to this clustering approach for the narrow domain corpus. In this particular case the gap between the baseline and the average is smaller.

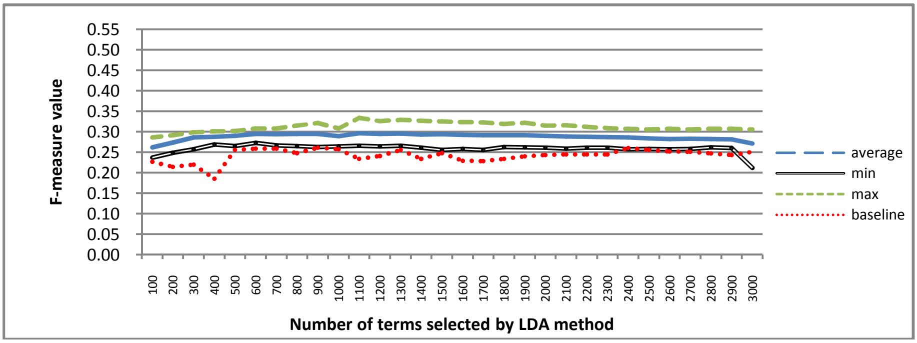


Fig. 4. Clustering results using the "narrow" domain corpus

In other words, the performance of our methodology is not as high as that obtained with wide domain, but in any case we still achieve an improvement. We consider that the reduced improvement in this domain is due to the fact that when the enrichment process expands the corpus, it introduces some noisy terms, i.e., terms that share many categories in this kind of domain. Even if we have used the Term Selection Technique to avoid this noisy information, it is difficult to highlight the discriminative information of each category. All of this makes the clustering task more difficult. Therefore, the size of the each document (in this case, weblog posts) is another important factor involved in this complex clustering process.

6 Analysis of Results

In this section, we discuss the results obtained in the experiments. As we expected we have obtained the best results with the wide domain corpus, because the categories share a low percentage of vocabulary. On the other hand, the narrow domain has a very high overlapping vocabulary between categories, which is a very important factor reflected in the clustering process. We have found out that the S-TEM methodology can help the generation of prototypes because the LDA has taken advantage of the expansion methodology. The improvement of the representation that S-TEM gives to the narrow domain posts is less because of the high overlapping vocabulary, and also the noise introduced by the enriching process derived from the Pointwise Mutual Information that is based on the frequency of correlated terms. It is also important to mention that we have outperformed the baseline in both cases (narrow and wide domain).

An additional aspect found in our experiment and shown in Figures 3 and 4 is that using nearly a thousand terms per category in the prototypes is good enough to get acceptable result the clustering process this may impact in the processing time due to we can manage relatively low-dimension vectors.

7 Conclusions and Further Work

We have presented a novel methodology to cluster weblogs based on a generative probabilistic model (LDA) in conjunction with an enriching methodology (S-TEM) applied to two different kind of corpus, one considered as "narrow" domain with very similar categories, and other considered as "wide" domain with low overlapping vocabulary or dissimilar categories.

We have confirmed that our approach works well with wide domain corpora obtaining 0.53 in F-measure with just 10% of the vocabulary to generate the best prototypes and it has also shown improved results (albeit with a smaller gain) with narrow domains. Finally, due to the simplicity of the clustering method used, our approach has shown acceptable ranges in the processing time.

In future work, we plan to modify our approach and cluster the expanded posts used in the generation of the prototypes with the objective of giving better information to the clustering process and improve representation of the post in particular in narrow domain. We are also interested in working on the scalability of our approach in order to be able to manage data sets with huge number of documents and classes. To further this aim, we are intending to adapt the approach described in [12].

Acknowledgments. The work of the fourth author has been partially supported by the TEXTENTERPRISE 2.0 TIN2009-13391-C04-03 research project and the work of the first author by the Mexican Council of Science and Technology (CONACYT).

References

1. Agrawal, N., Galan, M., Liu, H., Subramanya, S.: Clustering blogs with collective wisdom. In: Proc. of the International Conference on Web Engineering, pp. 336–339. IEEE Computer Society, USA (2008)
2. Allan, J., Carbonell, J.G., Doddington, G., Yamron, J., Yang, Y.: Topic Detection and Tracking Pilot Study: Final Report. In: Proc. DARPA Broadcast News Transcription and Understanding Workshop (1998)
3. Allan, J., Papka, R., Lavrenko, V.: On-line new event detection and tracking. In: Proc. SIGIR International Conference on Research and Development in Information Retrieval, pp. 37–45. ACM, NY (1998)
4. Banerjee, S., Pedersen, T.: An adapted Lesk algorithm for word sense disambiguation using WordNet. In: Gelbukh, A. (ed.) CICLing 2006. LNCS, vol. 3878, pp. 136–145. Springer, Heidelberg (2006)
5. Blei, D.M., Ng, A.Y., Jordan, M.I.: Latent Dirichlet Allocation. The Journal of Marchine Learning Research, JMLR.org 3, 993–1022 (2003)
6. Deerwester, S., Dumais, S., Landauer, T., Furnas, G., Harshman, R.: Indexing by latent semantic analysis. Journal of American Society of Information Science 41, 391–407 (1990)
7. Fellbaum, C.: WordNet: An Electronic Lexical Database. MIT Press, Cambridge (1998)
8. Flynn, C., Dunnion, J.: Topic Detection in the News Domain. In: Proc. of the 2004 International Symposium on Information and Communication Technologies, pp. 103–108. ACM, New York (2004)
9. Grefenstette, G.: Explorations in Automatic Thesaurus Discovery. Kluwer Ac., Dordrecht (1994)
10. Harris, Z.: Distributional structure. Word 10(23), 146–162 (1954)

11. Hofman, T.: Probabilistic latent semantic indexing. In: Proc. of the Twenty-Second Annual International SIGIR Conference, pp. 50–57. ACM, NY (1999)
12. Karp, R.M., Rabin, M.O.: Efficient Randomized Pattern-Matching Algorithms. IBM Journal of Research and Development 31(2), 249–260 (1987)
13. Li, B., Xu, S., Zhang, J.: Enhancing Clustering Blog Documents by Utilizing Author/Reader Comments. In: ACM Southeast Regional Conference, pp. 94–99 (2007)
14. Manning, D.C., Schutze, H.: Foundations of Statistical Natural Language Processing. MIT Press, Cambridge (1999)
15. Perez-Tellez, F., Pinto, D., Cardiff, J., Rosso, P.: Characterizing Weblog Corpora. In: Horacek, H., Métais, E., Muñoz, R., Wolska, M. (eds.) NLPIS 2010. LNCS, vol. 5723, pp. 299–300. Springer, Heidelberg (2010)
16. Pinto, D.: On Clustering and Evaluation of Narrow Domain Short-Text Corpora. PhD dissertation, Universidad Politecnica de Valencia, Spain (2008)
17. Qiu, Y., Frei, H.P.: Concept based query expansion. In: Proc. of the 16th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 160–169. ACM, New York (1993)
18. Sekiguchi, Y., Kawashima, H., Okuda, H., Oku, M.: Topic Detection from Blog Documents Using Users' Interests. In: Proc. of the 7th International Conference on Mobile Data Management (2006)
19. Spärck, J.K.: A statistical interpretation of term specificity and its application in retrieval. Journal of Documentation 28, 11–21 (1972)
20. Steinbach, M., Karypis, G., Kumar, V.: A comparison of document clustering techniques. In: KDD Workshop on Text Mining (2000)
21. Wartena, C., Brussee, R.: Topic Detection by Clustering Keywords. In: Proc. of the 19th International Conference on Database and Expert Systems Application, pp. 54–58. IEEE Computer Society, USA (2008)

A Naïve Bayes Approach to Cross-Lingual Word Sense Disambiguation and Lexical Substitution*

David Pinto, Darnes Vilariño, Carlos Balderas,
Mireya Tovar, and Beatriz Beltrán

Faculty of Computer Science
B. Autonomous University of Puebla, Mexico
{dpinto,darnes,mtovar,bbeltran}@cs.buap.mx

Abstract. Word Sense Disambiguation (WSD) is considered one of the most important problems in Natural Language Processing [1]. It is claimed that WSD is essential for those applications that require of language comprehension modules such as search engines, machine translation systems, automatic answer machines, second life agents, etc. Moreover, with the huge amounts of information in Internet and the fact that this information is continuosly growing in different languages, we are encourage to deal with cross-lingual scenarios where WSD systems are also needed. On the other hand, Lexical Substitution (LS) refers to the process of finding a substitute word for a source word in a given sentence. The LS task needs to be approached by firstly disambiguating the source word, therefore, these two tasks (WSD and LS) are somehow related. In this paper, we present a naïve approach to tackle the problem of cross-lingual WSD and cross-lingual lexical substitution. We use a bilingual statistical dictionary, which is calculated with Giza++ by using the EUROPARL parallel corpus, in order to calculate the probability of a source word to be translated to a target word (which is assumed to be the correct sense of the source word but in a different language). Two versions of the probabilistic model are tested: unweighted and weighted. The results were compared with those of an international competition, obtaining a good performance.

1 Introduction

Word Sense Disambiguation is a task that consists in selecting the correct sense of a given ambiguous word in a given context. There are several approaches that have been proposed for WSD [1], however, the problem of automatic WSD has not been resolved. Competitions such as Senseval[1] and recently SemEval[2] have also motivated the generation of new systems for WSD, providing an interesting

* This work has been partially supported by the CONACYT project #106625, as well as by the PROMEP/103.5/09/4213 grant.

[1] http://www.senseval.org/

[2] http://nlp.cs.swarthmore.edu/semeval/
http://semeval2.fbk.eu/

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 352–361, 2010.

environment for testing those systems. Despite the WSD task has been studied for a long time, the expected feeling is that WSD should be integrated into real applications such as mono and multi-lingual search engines, machine translation systems, automatic answer machines, etc [1]. Different studies on this issue have demonstrated that those applications benefit from WSD. For instance, the case of machine translation [2,3].

Even if the problem of WSD is difficult when dealing in only one language, when we consider its cross-lingual version (C-WSD), this problem becomes to be much more complex. In this case, it is needed not only to find the correct translation, but this translation must consider the contextual senses of the original sentence (in a source language), in order to find the correct sense (in the target language) of the source word.

For the experiments carried out in this paper, we have considered English as the source language and Spanish as the target language. We do not use an inventory of senses, as the most of the WSD systems do. Instead, we attempt to find those senses automatically by means of a bilingual statistical dictionary which is calculated on the basis of the IBM-1 translation model[3], by using the EUROPARL parallel corpus[4]. In this way, we obtain a set canditate translations for the source ambiguous word and applying a probabilistic model we may rank those translations in order to determine the most probable word/sense for the ambiguous word.

We have also considered the problem of Cross-lingual Lexical Substitution(C-LS) for the experiments presented in this paper. The aim was to test the results obtained in C-WSD to solve the problem of C-LS. In general, the C-LS problem may be defined as follows: given a paragraph and a source word, the goal is to provide several correct translations for that word in a given language, with the constraint that the translations fit the given context in the source language. We consider this task to be a step forward of the English lexical substitution task from SemEval-2007 [4], but this time the problem is considered in a cross-lingual scenario.

The rest of this paper is structured as follows. Section 2 presents the two datasets used in the experiments. In Section 3 we define the probabilistic model used as classifier for both, the cross-lingual WSD and LS. The experimental results are shown in Section 4 together with a discussion of findings. Finally, the conclusions and further work are given in Section 5.

2 Datasets

For the experiments conducted on cross-lingual word sense disambiguation we have used 25 polysemous English nouns. We selected five nouns (movement, plant, occupation, bank and passage), each with 20 example instances, for conforming a development corpus. The remaining polysemous nouns (twenty) were considered for a test corpus. In the case of the test corpus, we used 50 instances per noun. A list of the ambiguous nouns of the test corpus may be seen in Table 1.

[3] We used Giza++ (http://fjoch.com/GIZA++.html)

[4] http://www.statmt.org/europarl/

Table 1. Test set for the cross-lingual WSD task

Noun name			
coach	education	execution	figure
job	post	pot	range
rest	ring	mood	soil
strain	match	scene	test
mission	letter	paper	side

Table 2. Development set for the cross-lingual lexical substitution task

Polysemous word name									
take	v	stand	v	run	v	manage	v	lie	v
forget	v	fix	v	find	v	fear	v	bar	v
well	r	tight	r	severely	r	nearly	r	finally	r
wild	n	stand	n	gall	n	film	n	examination	n
dark	n	cross	n	can	n	bar	n	wild	a
rough	a	rich	a	reasonable	a	outdoor	a	neat	a
nasty	a	grim	a	cross	a	bright	a		

On the other hand, for the experiments carried out on the cross-lingual lexical substitution we employed two corpora: the development corpus and the test corpus. In Table 2 we may see the different polysemous words used in the development corpus. Whereas, Table 3 shows the different ambiguous words used. As may be seen in the case of the C-LS task, we have considered other grammatical categories different than nouns. We denoted verbs with *v*, nouns with *n*, adjectives with *a* and adverbs with *r*.

3 A Naïve Bayes Approach to WSD and LS

In this section it is presented an overview of the presented system, but also we further discuss the particularities of the general approach for each task evaluated. We will start this section by explaining the manner we deal with the C-WSD problem.

3.1 Cross-Lingual Word Sense Disambiguation

In Figure 1 we may see the complete process of approaching the problem of cross-lingual WSD.

We have approached the cross-lingual word sense disambiguation task by means of a probabilistic system which considers the probability of a word sense (in a target language), given a sentence (in a source language) containing the ambiguous word. In particular, we used the Naive Bayes classifier in two different ways. First, we calculated the probability of each word in the source language of being associated/translated to the corresponding word (in the target language).

Table 3. Test set for the cross-lingual lexical substitution task

Polysemous word name									
work	v	wind	v	touch	v	throw	v	tap	v
strike	v	skip	v	show	v	shed	v	see	v
return	v	render	v	put	v	pass	v	order	v
let	v	hold	v	go	v	fire	v	drop	v
draw	v	dismiss	v	clear	v	clean	v	check	v
charge	v	carry	v	call	v	burst	v	bring	v
acquire	v	yet	r	right	r	only	r	now	r
late	r	hard	r	forward	r	closely	r	away	r
around	r	about	r	way	n	test	n	strain	n
side	n	shot	n	shade	n	scene	n	ring	n
rest	n	range	n	pulse	n	pot	n	post	n
paper	n	mission	n	match	n	mass	n	lead	n
investigator	n	girl	n	function	n	figure	n	field	n
execution	n	coach	n	cap	n	bug	n	board	n
blow	n	account	n	tender	a	strong	a	straight	a
stiff	a	special	a	solid	a	soft	a	serious	a
rude	a	raw	a	profound	a	poor	a	open	a
new	a	live	a	light	a	liberal	a	informal	a
heavy	a	good	a	fundamental	a	fresh	a	flat	a
extended	a	dry	a	clear	a	civil	a	blue	a

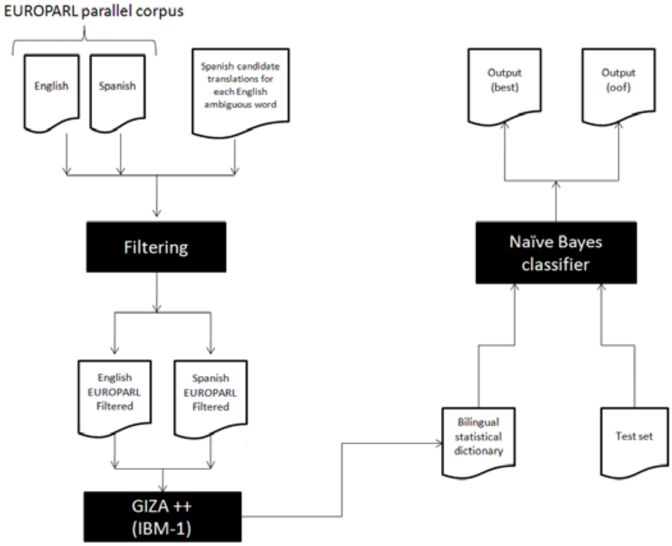

EUROPARL parallel corpus
English
Spanish
Spanish candidate translations for each English ambiguous word
Filtering
English EUROPARL Filtered
Spanish EUROPARL Filtered
GIZA ++ (IBM-1)
Bilingual statistical dictionary
Test set
Naïve Bayes classifier
Output (best)
Output (oof)

Fig. 1. An overview of the presented approach for cross-lingual word sense disambiguation

The probabilities were estimated by means of a bilingual statistical dictionary which is calculated using the Giza++ system over the EUROPARL parallel corpus. We filtered this corpus by selecting only those sentences which included some senses of the ambiguous word which were obtained by translating this ambiguous word on the Google search engine. The second approach considered a weighted probability for each word in the source sentence. The closer a word of the sentence to the ambiguous word, the higher the weight given to it.

In other words, given an English sentence $S = \{w_1, w_2, \cdots, w_k, \cdots, w_{k+1}, \cdots\}$ with the ambiguos word w_k in position k. Let us consider N candidate translations of w_k, $\{t_1^k, t_2^k, \cdots, t_N^k\}$ obtained somehow (we will further discuss about this issue in this section). We are insterested on finding the most probable candidate translations for the polysemous word w_k. Therefore, we may use a Naïve Bayes classifier which considers the probability of t_i^k given w_k. A formal description of the classifier is given as follows.

$$p(t_i^k|S) = p(t_i^k|w_1, w_2, \cdots, w_k, \cdots) \tag{1}$$

$$p(t_i^k|w_1, w_2, \cdots, w_k, \cdots) = \frac{p(t_i^k)p(w_1, w_2, \cdots, w_k, \cdots|t_i^k)}{p(w_1, w_2, \cdots, w_k, \cdots)} \tag{2}$$

We are interested on finding the argument that maximizes $p(t_i^k|S)$, therefore, we may avoid calculating the denominator. Moreover, if we assume that all the different translations are equally distributed, then Eq. (2) must be approximated by Eq. (3).

$$p(t_i^k|w_1, w_2, \cdots, w_k, \cdots) \approx p(w_1, w_2, \cdots, w_k, \cdots|t_i^k) \tag{3}$$

The complete calculation of Eq. (3) requires to apply the chain rule. However, if we assumed that the words of the sentence are independent, then we may rewrite Eq. (3) as Eq. (4).

$$p(t_i^k|w_1, w_2, \cdots, w_k, \cdots) \approx \prod_{j=1}^{|S|} p(w_j|t_i^k) \tag{4}$$

The best translation is obtained as shown in Eq. (5). Nevertheless the position of the ambiguous word, we are only considering a product of the probabilites of translation. Thus, we named this approach, the *unweighted version.* Algorithm 1 provides details about implementation.

$$BestSense_u(S) = \arg\max_{t_i^k} \prod_{j=1}^{|S|} p(w_j|t_i^k) \tag{5}$$

with $i = 1, \cdots, N$.

A second approach (*weighted version*) is also proposed as shown in Eq. (6). Algorithm 2 provides details about implementation.

$$BestSense_w(S) = \arg\max_{t_i^k} \prod_{j=1}^{|S|} p(w_j|t_i^k) * \frac{1}{k - j + 1} \tag{6}$$

with $i = 1, \cdots, N$.

Algorithm 1. An unweighted naïve Bayes approach to cross-lingual WSD

```
Input: A set $Q$ of sentences: $Q = \{S_1, S_2, \cdots\}$;
Dictionary $= p(w|t)$: A bilingual statistical dictionary;
Output: The best word/sense for each ambiguous word $w_j \in S_l$
for $l = 1$ to $|Q|$ do
    for $i = 1$ to $N$ do
        $P_{l,i} = 1$;
        for $j = 1$ to $|S_l|$ do
            foreach $w_j \in S_l$ do
                if $w_j \in Dictionary$ then
                    $P_{l,i} = P_{l,i} * p(w_j|t_i^k)$;
                else
                    $P_{l,i} = P_{l,i} * \epsilon$;
                end
            end
        end
    end
end
return $\arg\max_{t_i^k} \prod_{j=1}^{|S|} p(w_j|t_i^k)$
```

With respect to the N candidate translations of the polysemous word w_k, $\{t_1^k, t_2^k, \cdots, t_N^k\}$, we have used of the Google translator[5]. Google provides all the possible translations for w_k with the corresponding grammatical category. Therefore, we are able to use those translations that match with the same grammatical category of the ambiguous word. Even if we attempted other approaches such as selecting the most probable translations from the statistical dictionary, we confirmed that by using the Google online translator we obtain the best results. We consider that this result is derived from the fact that Google has a better language model than we have, because our bilingual statistical dictionary was trained only with the EUROPARL parallel corpus.

The experimental results of both, the *unweighted* and the *weighted* versions of the presented approach for cross-lingual word sense disambiguation are given in Section 4.

3.2 Cross-Lingual Lexical Substitution

In Figure 2 we may see the complete process of approaching the problem of cross-lingual lexical substitution. Notice that this task is complemented by the WSD solver.

This module is based on the cross-lingual word sense disambiguation system. Once we knew the best word/sense (Spanish) for the ambiguous word (English), we lemmatized the Spanish word. We searched, at WordNet, the synonyms of this word (sense) that agree with the grammatical category (noun, verb, etc) of the query (source polysemous word).

[5] http://translate.google.com.mx/

Algorithm 2. A weighted naïve Bayes approach to cross-lingual WSD

```
Input: A set $Q$ of sentences: $Q = \{S_1, S_2, \cdots\}$;
Dictionary $= p(w|t)$: A bilingual statistical dictionary;
Output: The best word/sense for each ambiguous word $w_j \in S_l$
for $l = 1$ to $|Q|$ do
    for $i = 1$ to $N$ do
        $P_{l,i} = 1$;
        for $j = 1$ to $|S_l|$ do
            foreach $w_j \in S_l$ do
                if $w_j \in Dictionary$ then
                    $P_{l,i} = P_{l,i} * p(w_j|t_i^k) * \frac{1}{k-j+1}$;
                else
                    $P_{l,i} = P_{l,i} * \epsilon$;
                end
            end
        end
    end
end
return arg $\max_{t_i^k} \prod_{j=1}^{|S|} p(w_j|t_i^k) * \frac{1}{k-j+1}$
```

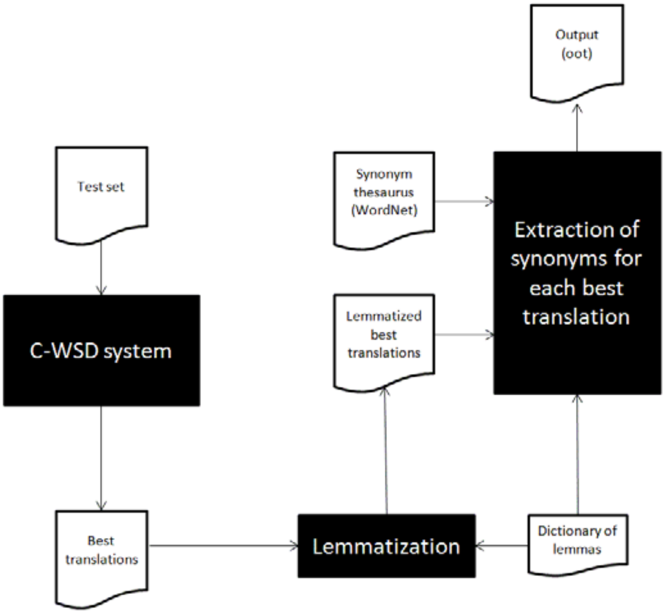

Output
(oot)
Test set
Synonym thesaurus (WordNet)
Extraction of synonyms for each best translation
C-WSD system
Lemmatized best translations
Best translations
Lemmatization
Dictionary of lemmas

Fig. 2. An overview of the presented approach for cross-lingual lexical substitution

Table 4. Description of runs

Run name	*Description*
FCC-WSD1 :	Best translation (one target word) / unweighted version
FCC-WSD2 :	Ten best translations (ten target words - *oof*) / unweighted version
FCC-WSD3 :	Best translation (one target word) / weighted version
FCC-WSD4 :	Ten best translations (ten target words - *oof*) / weighted version

4 Experimental Results

In this section we present the obtained results for both, the cross-lingual word sense disambiguation task and the cross-lingual lexical substitution task.

4.1 Cross-Lingual Word Sense Disambiguation

In Table 5 we may see the results we have obtained with the different versions of the presented approach. In particular, we have tested four different runs which correspond to two evaluations for each different version of the probabilistic classifier. The description of each run is given in Table 4.

In the same Table we can find a comparison of our runs with others approaches presented at the SemEval-2 competition. The *UvT* team submitted four runs (UvT-WSD1 and UvT-WSD2 for the both *best* and the *oof* evaluation) which make use of a k-nearest neighbour classifier to build one word sense for each target ambiguous word, and select translations from a bilingual dictionary obtained by executing the GIZA package on the EUROPARL parallel corpus [5]. The University of Heidelberg participated submitting other four runs (UHD-1 and UHD-2 for both the *best* and the *oof* evaluation). They approached the cross-lingual word sense disambiguation by finding the most appropriate translation in different languages on the basis of a multilingual co-ocurrence graph, which is automatically induced from the target words aligned contexts found in the EUROPARL and JRC-Arquis parallel corpora [5]. Finally, there was another team which submitted two runs: ColEur1 (*best* evaluation) and ColEur2 (*oof* evaluation) with a supervised approach that uses the translations obtained with GIZA from the EUROPARL parallel corpus in order to distinguish between senses in the English source sentences [5]. In general, we may see that all the teams used the GIZA software in order to find a bilingual statistical dictionary. Therefore, the main differences among all these approaches are in the way that they represents the original ambiguous sentence (including the pre-processing stage), and the manner the teams filter the results obtained by GIZA.

We obtained a better performance with those runs that were evaluated with the ten best translations than with those that were evaluated with only the best ones. This fact lead us to consider in further work to improve the ranking of the translations found by our system. On other hand, the unweighted version

Table 5. Evaluation of the cross-lingual word sense disambiguation task

System name	*Precision (%)*	*Recall (%)*
UvT-WSD1	23.42	23.42
UvT-WSD2	19.92	19.92
FCC-WSD1	15.09	15.09
FCC-WSD3	14.43	14.43
UHD-1	20.48	16.33
UHD-2	20.2	16.09
ColEur1	19.78	19.59

a) Best translation

System name	*Precision (%)*	*Recall (%)*
UvT-WSD1	42.17	42.17
UvT-WSD2	43.12	43.12
FCC-WSD2	40.76	40.76
FCC-WSD4	38.46	38.46
UHD-1	38.78	31.81
UHD-2	37.74	31.3
ColEur2	35.84	35.46

b) Five best translations (oof)

of the proposed classifier improved the weighted one. This behavior was unexpected, because in the development dataset, the results were opposite. We got a better performance than other systems, and those runs that outperformed our system runs did it by around 3% of precision and recall in the case of the oof evaluation.

4.2 Cross-Lingual Lexical Substitution

In Table 6 we may see the obtained results for the cross-lingual lexical substitution task. The obtained results are low in comparison with the best one (the complete description of all the runs may be found in [6]). Since this task relies on the C-WSD task, then a lower performance on the C-WSD task will conduct to a even lower performance in C-LS. Firstly, we need to improve the C-WSD solver. In particular, we need to improve the ranking procedure in order to obtain a better translation of the source ambiguous word. Moreover, we consider that the use of language modeling would be of high benefit, since we could test whether or not a given translation together with the terms in its context would have high probability in the target language.

Table 6. Evaluation of the cross-lingual lexical substitution task (the ten best results - *oot*)

System name	*Precision (%)*	*Recall (%)*
UvT-v	58.91	58.91
UvT-g	55.29	55.29
UBA-W	52.75	52.75
WLVUSP	48.48	48.48
UBA-T	47.99	47.99
USPWLV	47.6	47.6
ColSlm	43.91	46.61
ColEur	41.72	44.77
TYO	34.54	35.46
IRST-1	31.48	33.14
FCC-LS	23.9	23.9
IRSTbs	8.33	29.74

5 Conclusions and Further Work

In this paper we have presented a system for cross-lingual word sense disambiguation and cross-lingual lexical substitution. The approach uses a Naïve Bayes classifier which is feed with the probabilities obtained from a bilingual statistical dictionary. Two different versions of the classifier, unweighted and weighted were tested. The results were compared with those of an international competition, obtaining a good performance. As further work, we need to improve the ranking module of the cross-lingual WSD classifier. Moreover, we consider that the use of a language model for Spanish would highly improve the results on the cross-lingual lexical substitution task.

References

1. Aguirre, E., Edmonds, P.: Word Sense Disambiguation, Text, Speech and Language Technology. Springer, Heidelberg (2006)
2. Chan, Y., Ng, H., Chiang, D.: Word sense disambiguation improves statistical machine translation. In: Proceedings of the 45th Annual Meeting of the Association of Computational Linguistics, pp. 33–40 (2007)
3. Carpuat, M., Wu, D.: Improving statistical machine translation using word sense disambiguation. In: Proceedings of the 2007 Joint Conference on Empirical Methods in Natural Language Processing and Computational Natural Language Learning (EMNLPCoNLL), pp. 61–72 (2007)
4. McCarthy, D., Navigli, R.: English lexical substitution task. In: SemEval 2007 Proceedings of the 4th International Workshop on Semantic Evaluations, pp. 48–53 (2007)
5. Mihalcea, R., Sinha, R., McCarthy, D.: Semeval-2010 task2: cross-lingual lexical substitution. In: Proceedings of the Fifth International Workshop on Semantic Evaluations (SemEval-2010). Association for Computational Linguistics (2010)
6. Lefever, E., Hoste, V.: Semeval-2010 task3:cross-lingual word sense disambiguation. In: Proceedings of the Fifth International Workshop on Semantic Evaluations (SemEval-2010). Association for Computational Linguistics (2010)

Supervised Learning for Semantic Classification of Spanish Collocations

Alexander Gelbukh and Olga Kolesnikova

Center for Computing Research, National Polytechnic Institute
Mexico City, 07738, Mexico
www.gelbukh.com, kolesolga@gmail.com

Abstract. The meaning of word combination such as give a book or lend money can be obtained by mechanically combining the meaning of the two constituting words: to give is to hand over, a book is a pack of pages, then to give a book is to hand over a pack of pages. However, the meaning of such word combinations as give a lecture or lend support is not obtained in this way: to give a lecture is not to hand it over. Such word pairs are called collocations. While their meaning cannot be derived automatically from the meaning of their constituents, we show how to predict the meaning of a previously unseen word combination using semantic regularities we observe in a training set of collocations whose meaning has been specified manually.

1 Introduction

In natural language, individual words can either combine freely or form so-called collocations. The meaning of a free word combination can be understood from the meaning of its constituents. In particular, in automatic text understanding, semantic representation of a free word combination can be constructed by concatenating two definitions. For example, *to make* is 'to create', *a dress* is 'a piece of clothes', thus *to make a dress* is 'to create a piece of clothes'. However, the combination *to make an announcement* means 'to communicate information' but not 'to create information'. That is, simply concatenating the two definitions does not work here: while one word's meaning is literal, the other's is quite different. Such word combinations are called collocations. For example: *to make an announcement*, *to give a lecture*, *to pose an obstacle*.

The word used in a collocation in its literal, or typical, meaning is called the **base** of the collocation, and the other word, the **collocate**. In a collocation (e.g., *make an announcement*), the collocate (*make*) acquires a meaning ('communicate') different from its typical meaning ('create'), i.e., the meaning it has in free word combinations (e.g., *make a dress*).

Collocations present a difficulty in automatic text understanding because the dictionaries usually do not include as senses all meanings that a word can have in collocations. One solution to this problem is to include all such meanings in the dictionary: with each word, add senses corresponding to all the meanings this word can have as a collocate in some collocation, along with the corresponding base. This, however, would be time and space consuming. What is more, such list of senses

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 362–371, 2010.

would always be incomplete, since new collocations constantly appear in the language. A better solution is to build a system capable of predicting those new meanings on the fly.

In this paper we examine the latter approach. We look for semantic patterns in collocations and train a system to detect them, in order to predict a meaning of previously unseen collocations.

It has been observed in linguistic research that different collocates may express the same meaning. For example, consider collocations *deliver a lecture*, *take a walk*, *pose an obstacle*. Though the nouns functioning as bases in these collocations (*lecture*, *walk*, *obstacle*) show diverse semantics, the collocates expressed by the verbs have the same semantic content, namely 'to do, perform, carry out something'. In collocations *plant a garden*, *give surprise*, *create a difficulty*, all the verbs have the meaning 'to cause that something come into existence'. Thus, the first group of collocation has the semantic pattern "do what is denoted by the base" and the second group, "cause that what denoted by the base come into existence". Semantic patterns can be specified using a formalism called **lexical function**[1].

Lexical function (LF) is defined in lexical semantics [2] as a function that associates a word with a corresponding word such that the latter expresses a given abstract meaning indicated by the name of lexical function. The concept of lexical function was introduced in linguistics as an element of functional models of natural language called *Meaning-Text Models*. The framework of the said models is *Meaning-Text Theory*. For details, see [6] and [3]. About 70 lexical functions have been identified. Each LF has been given a name in the form of an abbreviated Latin word whose meaning corresponds to the semantics of collocation. LF takes an input, termed the *keyword,* and produces the output, termed the *value* of LF. The notation of LF also includes information of syntactic structure of utterances where the keyword is used together with the LF's value in a collocation. The syntactic structure is encoded with a string of integers put as an index after the LF's name. Integer values specify semantic roles of LF's keyword. For example, 1 stands for the agent, 2 stands for the patient, etc. Positions of integers in the string signify syntactic functions. First position in the string stands for subject, the second for direct object, etc. For example, the string *12* says that the subject is the agent, and the direct object is the patient. We illustrate the above said with the examples given in Table 1, where K stands for *keyword*. Meanings of LFs are taken from [7].

Table 1 presents simple lexical functions which capture a single semantic element. There are collocations where the collocate may express a complex meaning including more than one semantic element. The semantic structure of such collocations are represented by complex semantic functions. Examples of some complex LFs are given in Table 3 of Section 4.

The rest of the paper is organized as follows. Section 2 defines the problem; Section 3 sketches related work and presents state of the art results. Section 4 describes the dataset and classification methods. The results obtained in our experiments are discussed in Section 5. Finally, Section 6 gives conclusions and speaks of future work.

[1] Not to be confused with a similar term used in computer programming.

2 Problem

Our task is to examine performance of supervised learning algorithms for classification of Spanish collocations according to the typology of LFs. The system is trained on a manually build corpus of verb-noun collocations annotated with LFs. Then the system is tested for recognition of eight LFs chosen for the experiments and the class of free word combinations (FWC), which gives us the total of nine semantic classes. We aim at detecting classifiers whose performance is best for each of these semantic classes.

Table 1. Examples of lexical functions

LF	Meaning	Keyword	Value	Syntactic structure	Example
Oper$_1$	Lat. *operari* – '*to do, carry out*'	*support*	*lend*	Subject is the *agent* of K.	*The company lends support to charity.*
Oper$_2$		*resistance*	*meet*	Subject is the *patient* of K.	*Allied Forces meet resistance in Afganistan.*
Labor$_{12}$	Lat. *laborare* – '*to work, toil*'	*control*	*keep under*	Subject is the agent of K, direct object is the patient of K.	*Sometimes people can not keep stress under control.*
Real$_1$	Lat. *realis* – '*real*'. The values are verbs meaning '*to fulfill the requirement of K*'.	*obligation*	*fulfill*	The same as for Oper$_1$.	*Adult children must fulfill their obligation to care for elderly parents.*
Stop$_2$	Lat. *stuppare* – '*to stop up, to plug*'. The values are verbs meaning '*to stop functioning*'.	*breath*	*loose*	The same as for Oper$_2$.	*She suddenly lost her breath and turned very pale.*

3 Related Work

It was mentioned in the Introduction that the concept of lexical functions was first elaborated in the frame of Meaning-Text Theory. Some research has been done on automatic detection of lexical functions. L. Wanner [13] proposed to view the task of LF detection as automatic classification of collocations according to LF typology. To fulfill this task, the nearest neighbor machine learning technique was used. Datasets included Spanish verb-noun pairs annotated with nine LFs. Every example in the datasets was represented by its hyperonyms retrieved from the Spanish part of EuroWordNet [12]. Every hyperonym was accompanied by its Basic Concepts and Top Concepts. A candidate instance was assigned that LF whose prototype was the most similar to the instance. Similarity was measured using path length in hyperonym hierarchy. These experiments gave the average F-measure of about 70%.

For classification of Spanish verb-noun collocations by LFs, Wanner *et al.* [14] applied four machine learning methods, namely, Nearest Neighbor technique, Naïve Bayesian network, Tree-Augmented Network Classification technique and a decision tree classification technique based on the ID3-algorithm. Experiments were carried out for two groups of verb-noun collocations. The first group included collocations where nouns belonged to the semantic field of emotions. In the second group, nouns were field-independent. In Section 5, we compare our results with those of [14] for verb-noun bigrams with field-independent nouns.

Alonso Ramos *et al.* [1] extracted collocations *support verb + object* from FrameNet corpus of examples [8]. Then they checked if the extracted collocations are of Oper$_n$. Their algorithm employ some syntactic, semantic and collocation annotations in the FrameNet corpus which serve as LF indicators. The proposed algorithm was tried out on a set of 208 instances and showed 76% accuracy. The authors come to a conclusion that it is feasible to extract and classify collocations according to LFs using semantically annotated corpora. Since the formalism of lexical function represents the correspondence between the keyword's semantic valency and of syntactic patterns together with semantic contents of collocation, such a conclusion sounds rather reasonable.

4 Data and Methodology

Our approach is based on supervised machine learning algorithms as implemented in the WEKA version 3-6-2 toolset [4, 15, 10]. Table 2 lists 68 classifiers tested for distinguishing between lexical functions. Table 3 presents LFs chosen for the experiments. LF meaning descriptions are taken from [7] and [13]; K stands for *keyword*. Examples are taken from the list of verb-noun pairs used in the experiments. The verb is the value of a corresponding LF, the noun is the keyword. Since examples are not grammatically well-formed utterances, the words are put in the form they appear in dictionary entries, with the exception of the verbs for Func$_0$ as explained further. The use of articles in the verb-noun pairs may vary depending on the context in which they are used in speech.

Now we give an explanation why the verbs for Func0 are not used in their dictionary form in Table 3. This touches upon a peculiarity of the Spanish syntax. In Spanish, the pattern *verb + noun*, where the noun is the subject of the verb noun, is not as rare as in English, where the subject typically precedes the verb. For example, it is common to see such headings in newspapers as, for instance, *Sube inflación en Eurozona a 1.4%*, lit. *Rises inflation in Eurozone by 1.4%* (cited from the newspaper *Excelsior*, issue of April 16, 2010). Verbs which are values of Func0 are used in utterances with their keywords as the subjects, so to give an example of a verb-noun pair where the verb is the value of Func0, we have to put the verb in the form that corresponds to the dictionary form of the noun, i.e., the verb has the form of 3rd person, singular number, present tense, indicative mood. The same is valid for the English translation. The words in the example *hace un mes* are always used in this word order. The corresponding English translation is *a month ago*. In English, there is no verb which is the exact equivalent of the verb *hacer* in the expression *hace un mes*, so *hacer* is translated not with a verb but the adverb *ago*.

For the experiments, we used the a list of the 1000 most frequent verb-noun pairs extracted automatically from the Spanish Web Corpus in the Sketch Engine [5]. All collocations in this list were annotated with lexical functions by human experts, who also tagged all the words in the collocations with word senses of the Spanish WordNet [9, 12]. The words in the pairs which are free word combinations (FWC) were also tagged with word senses and viewed as belonging to its own semantic class. Thus in our experiments, we treat FWC as a lexical function.

Table 2. WEKA classifiers trained to distinguish between lexical functions

AODE
AODEsr
BayesianLogisticRegression
BayesNet
HNB
NaiveBayes
NaiveBayesSimple
NaiveBayesUpdateable
WAODE
LibSVM
Logistic
RBFNetwork
SimpleLogistic
SMO
VotedPerceptron
Winnow
IB1
IBk
KStar
LWL
AdaBoostM1
AttributeSelectedClassifier
Bagging
ClassificationViaClustering
ClassificationViaRegression
CVParameterSelection
Dagging
Decorate
END
EnsembleSelection
FilteredClassifier
Grading
LogitBoost
MultiBoostAB
MultiClassClassifier
MultiScheme
OrdinalClassClassifier
RacedIncrementalLogitBoost
RandomCommittee
RandomSubSpace
RotationForest
Stacking
StackingC
ThresholdSelector
Vote
HyperPipes
VFI
ConjunctiveRule
DecisionTable
JRip
NNge
OneR
PART
Prism
Ridor
ZeroR
ADTree
BFTree
DecisionStump
FT
Id3
J48
J48graft
LADTree
RandomForest
RandomTree
REPTree
SimpleCart

Input files were constructed in the Attribute-Relation File Format (ARFF) [11] accessible by WEKA classifiers. For each verb-noun pair, we used binary feature representation. For each word in the list of annotated verb-noun pairs, all its hyperonyms were retrieved from the Spanish WordNet referenced above, and the word itself is considered as the zero-level hyperonym. This gives 654 features to represent the nouns, and 280 features to represent the verbs. Each verb-noun pair in the training set is represented as a vector:

$$v_1, v_2, ..., v_{654}, n_1, n_2, ..., n_{280}, LF,$$

where v_n, n_k can be 0 or 1, and *LF* is a categorical feature having the value *yes* for positive instances of LF for which classification is done, and *no* for negative instances. Negative instances are collocations that belong to all other LFs in the list of collocations except to the LF chosen for classification.

The performance of WEKA classifiers was evaluated by comparing the values of precision, recall, and F-measure using 10-fold cross-validation. The precision is the proportion of the examples which truly have class *x* among all those which were classified as class *x*. The recall is the proportion of examples which were classified as class *x*, among all examples which truly have class *x*. The F-measure is:

$$F = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}} \tag{1}$$

Table 3. Lexical functions chosen for the experiments

LF	Meaning	Collocation: LF value + keyword	
		Spanish	**English translation**
Oper_1	Lat. *operare* – ‘to do, perform’. Experience (if K is an emotion), carry out K.	*alcanzar un objetivo* *aplicar una medida* *corregir un error* *satisfacer una necesidad*	*achieve a goal* *apply a measure* *correct a mistake* *satisfy a necessity*
Oper_2	Undergo K, be source of K	*aprender una lección* *obtener una respuesta* *recibir ayuda* *sufrir un cambio*	*learn a lesson* *get an answer* *receive help* *suffer a change*
IncepOper_1	Lat. *incipere* – ‘to begin’. Begin to do, perform, experience, carry out K.	*adoptar una actitud* *cobrar importancia* *iniciar una sesión* *tomar posición*	*take an attitude* *acquire importance* *start a session* *obtain a position*
ContOper_1	Lat. *continuare* – ‘to continue’. Continue to do, perform, experience, carry out K.	*guardar silencio* *mantener el equilibrio* *seguir un modelo* *llevar una vida (ocupada)*	*keep silence* *keep one’s balance* *follow an example* *lead a (busy) life*
Func_0	Lat. *functionare* – ‘to function’. K exists, takes place, occurs.	*el tiempo pasa* *hace un mes* *una posibilidad cabe* *la razón existe*	*time flies* *a month ago* *there is a possibility* *the reason exists*
CausFunc_0	Lat. *causare* – ‘to cause’. Do something so that K begins occurring.	*encontrar respuesta* *establecer un sistema* *hacer campaña* *producir un efecto*	*find an answer* *establish a system* *conduct a campaign* *produce an effect*
CausFunc_1	A person/object, different from the agent of K, does something so that K occurs and has effect on the agent of K.	*abrir camino* *causar daño* *dar respuesta* *producir un cambio*	*open the way* *cause damage* *give an answer* *produce a change*
Real_1	Lat. *realis* – ‘*real*’. To fulfill the requirement of K, to act according to K.	*contestar una pregunta* *cumplir el requisito* *solucionar un problema* *utilizar la tecnología*	*answer a question* *fulfill the requirement* *solve a problem* *use technology*

5 Experimental Results

In Table 4, we present the main results obtained in our experiments. For each LF, we list three classifiers that have shown the best performance.

Table 4. Results showed by WEKA classifiers on the training set of lexical functions

LF	#	Classifier	P	R	F	Time	
						Train	Test
Oper1	157	BayesianLogisticRegression	0.879	0.866	**0.873**	1.22	0.50
		Id3	0.879	0.861	0.870	2.53	0.67
		SMO	0.862	0.866	0.864	5.19	0.39
Oper2	16	J48	0.923	0.571	**0.706**	0.72	0.45
		PART	0.923	0.571	0.706	0.72	0.45
		AttributeSelectedClassifier	0.923	0.571	0.706	1.58	0.42
IncepOper1	14	SMO	0.813	0.650	0.722	2.44	0.50
		NNge	0.923	0.600	0.727	1.61	1.17
		Prism	0.750	0.800	**0.774**	1.47	0.45
ContOper1	11	J48	0.833	0.769	0.800	0.30	0.44
		FilteredClassifier	0.833	0.769	0.800	0.38	0.47
		DecisionTable	0.909	0.769	**0.833**	10.0	0.30
Func0	22	AttributeSelectedClassifier	0.636	0.636	0.636	1.63	0.42
		HyperPipes	0.636	0.636	0.636	0.03	0.45
		BFTree	0.667	0.727	**0.696**	13.7	0.27
CausFunc0	102	REPTree	0.750	0.648	0.695	0.88	0.41
		EnsembleSelection	0.744	0.659	0.699	67.5	2.19
		JRip	0.747	0.705	**0.725**	0.97	0.39
CausFunc1	60	J48	0.842	0.696	0.762	1.22	0.42
		OrdinalClassClassifier	0.842	0.696	0.762	1.22	0.42
		END	0.842	0.696	**0.762**	1.39	0.42
Real$_1$	45	Id3	0.600	0.574	0.587	2.33	0.66
		NNge	0.614	0.574	0.593	2.64	2.75
		FT	0.650	0.553	**0.598**	12.7	15.1
FWC	198	SMO	0.656	0.623	0.639	4.67	0.36
		BayesianLogisticRegression	0.658	0.629	0.643	0.89	0.45
		Prism	0.639	0.702	**0.669**	25.9	0.34
Total:	625			*Average best:*	0.737		

Often, in classification experiments, the baseline is the performance of ZeroR classifier. ZeroR is a trivial algorithm that always predicts the majority class. It happens that the majority class in our training sets is always the class of negative instances. Even in the case of the LF which has the largest number of positive instances in the training set (198), the number of negative instances is still larger (427). Therefore, the ZeroR does not classify any test instances as positives, which gives always recall of 0 and undefined precision. Thus ZeroR is too bad a baseline to be considered.

In Table 4, the column marked by # specifies the number of positive instances for each LF. Recall that the whole dataset consists of all instances for all LFs mixed together and contains 635 items, so the number of negative instances is 625 minus the value of the # column. For each classifier, the amount of time taken to build the model is given as well as the time taken to test the model on the training set. Note that these figures are meaningful because all LFs used the same set of 625 collocations as its training set and the same test sets in the tenfold cross-validation procedure.

As it is seen from Table 4, no single classifier is the best one for detecting all LFs. For each LF, the highest result is achieved by a different classifier. However, Prism reaches the highest F-score for both IncepOper$_1$ and FWC, though recall that FWC (free word combinations) is not a lexical function but is considered as an independent class along with LFs. The maximum F-measure of 0.873 is achieved by BayesianLogisticRegression classifier for Oper$_1$. The lowest best F-measure of 0.598 is shown by FT for Real$_1$. The average F-measure (calculated over only the nine best results, one for each LF) is 0.737.

The maximum time taken to build a model on the training data is shown by EnsembleSelection classifier for CausFunc$_0$, and the minimum time, by HyperPipes for Func$_0$. The maximum time taken to test the model is given by FT for Real$_1$, and the minimum time, by BFTree for Func$_0$.

We observed no correlation between the number of instances in the training set and the results obtained from the classifiers. For example, a low result is shown for the class FWC which has the largest number of positive examples. On the contrary, the second top result is achieved for LF ContOper$_1$, with the smallest number of positive examples. The minimum F-measure is obtained for Real$_1$ whose number of positive examples is about 77% smaller than the largest number of positive examples (FWC) and about 71% smaller than the number of positive examples for Oper$_1$, the detection of which was the best.

Table 5. State of the art results for some LFs taken from [14]

LF	NN			NB			ID3			TAN			Our
	P	R	F	P	R	F	P	R	F	P	R	F	F
Oper1	0.65	0.55	0.60	0.87	0.64	**0.74**	0.52	0.51	0.51	0.75	0.49	0.59	**0.87**
Oper2	0.62	0.71	**0.66**	0.55	0.21	0.30	N/A			0.55	0.56	0.55	**0.71**
ContOper1	N/A			N/A			0.84	0.57	**0.70**	N/A			**0.83**
CausFunc0	0.59	0.79	**0.68**	0.44	0.89	0.59	N/A			0.45	0.57	0.50	**0.73**
Real1	0.58	0.44	**0.50**	0.58	0.37	0.45	N/A			0.78	0.36	0.49	**0.60**

Average: 0.75

For comparison, Table 5 gives the state of the art results reported in [14] for LF classification using machine learning techniques. Out of nine LFs mentioned in [14] we give in Table 5 only those five that we used in our experiments, i.e., that are represented in Table 4. Also, as we have explained in Section 3, [14] reports the results for two different datasets: one for a narrow semantic field (that of emotions) and another for a field-independent (general) dataset. Since our dataset is also general, comparing them with a narrow-field dataset would not be fair, so in Table 5 we only give the field-independent figures from [14].

Not all methods have been applied in [14] for all LFs; if a method was not applied for a particular LF, the corresponding cells are marked as N/A. In this table, NN stands for the Nearest Neighbor technique, NB for Naïve Bayesian network, ID3 is a decision tree classification technique based on the ID3-algorithm, and TAN for the Tree-Augmented Network Classification technique; P, R, and F are as in Table 4. In fact [14] did not give the value of recall, so we calculated it using (1). The last column repeats the best F-measure results from Table 4, for convenience of the reader. For each LF, the best result from [14], as well as the overall best result (including our experiments), are marked in boldface.

As seen from Table 5, for all LFs our experiments gave significantly higher figures than those reported in [14]. The best average F-measure from [14] is 0.66, while our experiments demonstrate the best average F-measure of 0.75. However, the comparison is not fair because different datasets have been used: the exact dataset used in [14] is unfortunately not available anymore;[2] ours is available from [16].

6 Conclusions and Future Work

We have shown that it is feasible to apply machine learning methods (specifically, those implemented in the WEKA toolkit) for predicting the meaning of unseen Spanish verb-noun collocations. Specifically, we trained a classifier to assign the semantic classes to a previously unseen collocation according to the formalism of lexical functions [7].

As features, both previous works and this work used the set of all hyperonyms of a word taken from WordNet [9, 12]. With this we re-confirmed that the set of hyperonyms can be used to describe lexical meaning and discriminate word senses.

Our experiments achieved the average F-measure of 74% (calculated basing on the best classifier for each of the nine LFs). This significantly outperforms the previously reported result of 66% [14] (our average result for the same subset of five LFs is 75%). However, the comparison is not fair because we used a dataset different from the one that has been used in [14], which is no longer available.

In the future, we plan to test other classification methods that were not examined in our experiments, both WEKA's modules that we did not yet test and methods not included in WEKA. We also plan to study the effect of other features, such as WordNet glosses. Finally, we intent to experiment with a word space models representing various similarity measures between collocations.

Acknowledgements. We are greatly thankful to Leo Wanner for useful and detailed discussions and for providing us lists of verb-noun collocations with partial manual markup. His kind help was very inspiring for us. We are also grateful to Adam Kilgarriff and Vojtěch Kovář for providing us a list of most frequent verb-noun pairs from the Spanish Web Corpus of the Sketch Engine, www.sketchengine.co.uk.

The work was done under partial support of Mexican Government: SNI, COFAA-IPN, PIFI-IPN, CONACYT grant 50206-H, and SIP-IPN grant 20100773.

References

1. Alonso Ramos, M., Rambow, O., Wanner, L.: Using semantically annotated corpora to build collocation resources. In: Proceedings of LREC, Marrakesh, Morocco, pp. 1154–1158 (2008)

[2] Personal communication with L. Wanner.

2. Apresjan, Ju. D.: Selected Works, Lexical Semantics, vol. 1. Vostochnaya Literatura Publishers, Moscow (1995) (in Russian)
3. Bolshakov, I.A., Gelbukh, A.F.: On Contemporary Status of the Meaning-Text Model. In: Guzman, A., Menchaka, R. (eds.) Selected Papers CIC-1999, CIC, IPN, Mexico City, pp. 17–25 (1999)
4. Hall, M., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: The WEKA Data Mining Software: An Update. SIGKDD Explorations 11(1) (2009)
5. Kilgarriff, A., Rychly, P., Smrz, P., Tugwell, D.: The Sketch Engine. In: Proceedings of EURALEX 2004, pp. 105–116 (2004)
6. Mel'čuk, I.A.: A Theory of the Meaning-Text Type Linguistic Models. Nauka Publishers, Moscow (1974) (in Russian)
7. Mel'čuk, I.A.: Lexical Functions: A Tool for the Description of Lexical Relations in a Lexicon. In: Wanner, L. (ed.) Lexical Functions in Lexicography and Natural Language Processing, pp. 37–102. Benjamins Academic Publishers, Amsterdam (1996)
8. Ruppenhofer, J., Ellsworth, M., Petruck, M., Johnson, C.R., Scheffczyk, J.: FrameNet II: Extended Theory and Practice. ICSI Berkeley (2006), http://framenet.icsi.berkeley.edu/book/book.pdf
9. Spanish WordNet, http://www.lsi.upc.edu/~nlp/web/index.php?Itemid=57&id=31&option=com_content&task=view (last viewed March 26, 2010)
10. The University of Waikato Computer Science Department Machine Learning Group, WEKA download, http://www.cs.waikato.ac.nz/~ml/weka/index_downloading.html (last viewed March 26, 2010)
11. The University of Waikato Computer Science Department Machine Learning Group, Attribute-Relation File Format, http://www.cs.waikato.ac.nz/~ml/weka/arff.html (last viewed March 26, 2010)
12. Vossen, P. (ed.): EuroWordNet: A Multilingual Database with Lexical Semantic Networks. Kluwer Academic Publishers, Dordrecht (1998)
13. Wanner, L.: Towards automatic fine-grained classification of verb-noun collocations. Natural Language Engineering 10(2), 95–143 (2004)
14. Wanner, L., Bohnet, B., Giereth, M.: What is beyond Collocations? Insights from Machine Learning Experiments. In: EURALEX (2006)
15. Witten, I.H., Frank, E.: Data Mining: Practical machine learning tools and techniques, 2nd edn. Morgan Kaufmann, San Francisco (2005)
16. http://www.Gelbukh.com/lexical-functions

Recognizing Textual Entailment with Statistical Methods

Miguel Angel Ríos Gaona[1], Alexander Gelbukh[1], and Sivaji Bandyopadhyay[2]

[1] Center for Computing Research, National Polytechnic Institute, Mexico
mriosb08@sagitario.cic.ipn.mx, gelbukh@gelbukh.com
[2] Computer Science & Engineering Department, Jadavpur University, Kolkata 700 032 India
sivaji_cse_ju@yahoo.com

Abstract. In this paper we propose a new cause-effect non-symmetric measure applied to the task of Recognizing Textual Entailment .First we searched over a big corpus for sentences which contains the discourse marker "because" and collected cause-effect pairs. The entailment recognition is based on measure the cause-effect relation between the text and the hypothesis using the relative frequencies of words from the cause-effect pairs. Our measure outperformed the baseline method, over the three test sets of the PASCAL Recognizing Textual Entailment Challenges (RTE). The measure shows to be good at discriminate over the "true" class. Therefore we develop a meta-classifier using a symmetric measure and a non-symmetric measure as base classifiers. So, our meta-classifier has a competitive performance.

1 Introduction

One of the biggest challenges in Natural Language Processing (NLP) is to provide a computer with the linguistic knowledge necessary to successfully perform language-based tasks. For example, the query "What does Peugeot manufacture?" a Question Answering (QA) system must be able to recognize, or infer, and answer which may be expressed differently from the query. Thus from text "Chrétien visited Peugeot's newly renovated car factory" entails the hypothesized answer from "Peugeot manufactures cars". A fundamental phenomenon in NLP is the variability of a semantic expression, which the same meaning could be expressed or infer from different text.

A task underlying this phenomenon is the ability to Recognize Textual Entailment. This task is defined as a directional relationship between pair of text expressions, denoted by T -the entailing "Text" and H -the entailed "Hypothesis". We say that T entails H if the meaning of H can be inferred from the meaning of T as could typically de interpreted by people [2].

Moreover, many NLP tasks have strong links to entailment: in Summarization (SUM), a summary should be entailed by the text; Paraphrases (PP) can be seen as mutual entailment between a text T and a hypothesis H; in Information Extraction (IE), the extracted information should also be entailed by the text; in QA the answer obtained for one question after the Information Extraction (IR) process must be entailed by the supporting snippet of text.

J.A. Carrasco-Ochoa et al. (Eds.): MCPR 2010, LNCS 6256, pp. 372–381, 2010.

To address this task, different methods have been proposed, with various degrees of success. The classification of methods depends on the level of representation of the T-H pair. Therefore the common criteria for entailment recognition were similarity between T and H, or the coverage of H by T in lexical representation methods and lexical syntactic representation methods, and the ability to infer H from T, in the logical representation approach. Zanzotto et al also measured the similarity between different T-H pairs, crosspair similarity. Some works [6] tried to detect non-entailment, by looking for various kinds of mismatch between the text and the hypothesis.

In this paper we propose a new cause-effect non-symmetric measure for entailment recognition based on the causal relation between the text and the hypothesis. The causal relation is measure by using the relative frequencies of words in a cause-effect set. These sets are extracted from a corpus by searching sentences containing the discourse marker “because”. Finally, we applied our method on a meta-classifier.

The paper is structured as follows. An overview of the related work in Section 2, Section 3 describes the proposed measure. Section 4 we shown experiments, and a comparison with previous results. Finally the conclusions are presented in Section 5.

2 Related Work

The RTE approaches can be classified depending in which textual entailment phenomena address or the type of representation (*levels of language*) of the T-H pair.

Thus each type of representation has operations in order to establish the entailment decision (e.g., word matching in the lexical level, tree edit distance in the syntactic level). The principal operations are similarity measures between T-H pair representations. But many of the similarity measures are symmetric. So a symmetric measure can not capture some of the aspects in the T→H relation. Because of if we altered the entailment relation (i.e., H→T) a symmetric function will give us the same score. Therefore methods like [9] propose a non-symmetric similarity measure, used in RTE-1 Challenge.

Glickman [3] uses as definition: T entails H iff $P(H \mid T) > P(H)$. The probabilities are calculated on the base of Web. The accuracy of the system is the best for RTE-1 (56%).

Another non-symmetric method is that of Kouylekov [7], who uses the definition: T entails H if and only if there exists a sequence of transformations applied to T such that H is obtained with a total cost below of a certain threshold. The following transformations are allowed: Insertion: insert a node from the dependency tree of H into the dependency tree of T; Deletion: delete a node from the dependency tree of T; Substitution: change a node in the T into a node of H. Each transformation has a cost and the cost of edit distance between T and H, $\mathrm{ed}(T, H)$ is the sum of costs of all applied transformations. The entailment score of a given pair is calculated as

$$\mathrm{score}(T,H) = \mathrm{ed}(T,H),$$

where $\mathrm{ed}(\cdot,H)$ is the cost of inserting the entire tree H. If this score is bigger than a learned threshold, the relation T →H holds. The accuracy of method is of 0.56.

In [9] an even "more non-symmetric" is proposed: when the edit distance (which is a Levenshtein modified distance) fulls the relation:

$$\mathrm{ed}(T,H) < \mathrm{ed}(H,T),$$

Then the relation T→H holds.

Other teams use a definition which in terms of representation of knowledge as feature structures could be formulated as: T entails H iff H subsumes T [9]. Even the method used in [2] is a non-symmetric one, as the definition used is: T entails H iff H is not informative in respect to T.

A method of establishing the entailment relation could be obtained using a non-symmetric measure of similarity between two texts presented by Corley and Mihalcea [1], the authors define the similarity between the texts T_i and T_j with respect to T_i as:

$$sim\left(T_i,T_j\right)_{Ti} = \frac{\sum_{pos}\left(\sum_{wk\in ws_{pos}^{ti}}\left(\max Sim\left(w_k\right)\times idf\left(w_k\right)\right)\right)}{\sum_{pos}\sum_{wk\in ws_{pos}^{ti}} idf\left(w_k\right)}$$

Here the sets of open-class words (nouns, verbs, adjective and adverbs) in each text segment are denoted by WST_i PoS (PoS: Part of Speech) and WST_j PoS. For a word w_k with a given PoS in T_i, the highest similarity of the words with the same pos in the other text T_j is denoted by maxSim(w_k).

Starting with this text-to-text similarity metric, we derive a textual entailment recognition system by applying the lexical refutation theory presented above. As the hypothesis H is less informative than the text T, for a TRUE pair the following relation will take place:

$$\mathrm{sim}(T,H) \times T < \mathrm{sim}(T,H) \times H$$

This relation can be proven using the lexical refutation [9]. A draft is the following: to prove T→H it is necessary to prove that the set of formulas {T; negH} is lexical contradictory (they denote also by T and negH the sets of disjunctive clauses of T and negH).

3 Proposed Methods

A causal relation refers to the relation between a cause and its effect or between regularly correlated events. One type of coherence relation we used is cause-effect, illustrated above. For example: (1) states the cause for the effect given in (2).

1. *There was bad weather at the airport*
2. *and so our flight got delayed.*

The causal relation subsumes the cause and the explanation relations in Hobbs [3]. Hobbs's cause relation holds if a discourse segment stating a cause occurs before a discourse segment stating an effect; an explanation relation holds if a discourse segment stating an effect occurs before a discourse segment stating a cause. The causal relation is encoded by adding a direction. In a graph, this can be represented by a directed arc going from cause to effect.

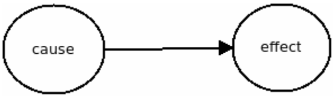


Fig. 1. Cause effect graph

Thus from Fig. 1 the causality is a directional relationship such as the relationship between a T-H pair. A non-symmetric similarity measure based on the count of co-occurrences of causal lexical pairs could be as follows: If a word *x* is a necessary cause of a word *y*, then the presence of *y* necessarily implies the presence of *x*.

3.1 Causal Non-symmetric Measure

The hypothesis behind our method is based on treat the T-H pair as a causal relation. Where the text T is a cause and the hypothesis H is its effect (i.e., T causes H).

The general scheme of the method is showed in Fig. 2:

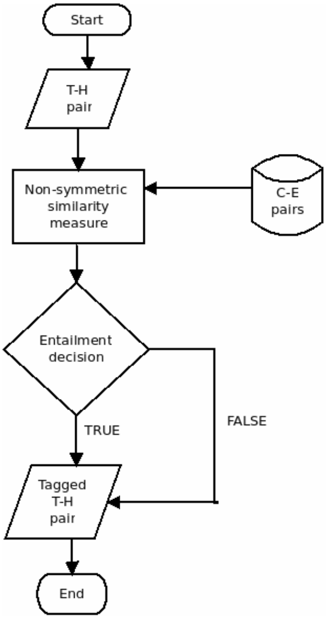


Fig. 2. General data flow of our system

In Fig. 2 we show the general data flow of the proposed method. The non-symmetric similarity measure is based on the count of co-occurrences of causal lexical pairs from a C-E pairs extracted from a corpus.

Algorithm 1. New non-symmetric similarity measure

```
For each word $t_i$ in $T$
 For each word $h_j$ in $H$
  $ce_j$=causal frequency($t_i$,$h_j$)
  $e_j$=causal frequency($h_j$)
 $max_i$ = argmax($ce_j$/$e_j$)
nonsymetric($T$,$H$)= Σ $max_i$
```

As we se in the Algorithm 1 the first causal frequency function is the count of words t_i and h_i related by the cue phrase (For example, a sentence, h…because…t) in a corpus of C-E pairs and the second causal frequency function is the count of word h_i in the C-E pairs, which gives us a non-symmetric score. Because the co-occurrences of T causes H is not the same like H causes T.

To each T-H pair the system measures the causal relation between them and then decides if the pair is true or false given a certain entailment decision.

Algorithm 2. Entailment decision

```
if non-symmetric($T$,$H$) > non-symetric($H$,$T$) then TRUE
else FALSE
```

In Algorithm 2 we show that the entailment decision basically penalize a T—H pair when the H→T relation is stronger than the T→H relation. Therefore the hypothesis H is more probably an effect than the text T. Therefore it is more probable that the text T implies the hypothesis H.

3.2 Symmetric and Non-symmetric Meta-classifier

It has been observed for related systems that a combination of separately trained features in the machine learning component can lead to an overall improvement in system performance, in particular if features from a more informed component and shallow ones are combined.

One of the main problems when machine-learning classifiers are employed in practice is to determine whether classifications assigned to new instances are reliable. The meta-classifier approach is one of the simplest approaches to this problem. Given a base classifiers, the approach is to learn a meta-classifier that predicts the correctness of each instance classification of the base classifiers. The sources of the meta-training data are the training instances. The meta-label of an instance indicates reliable classification, if the instance is classified correctly by a base classifier; otherwise, the meta-label indicates unreliable classification. The meta-classifier plus the base classifiers form one combined classifier. The classification rule of the combined classifier is to assign a class predicted by the base classifier to an instance if the meta-classifier decides that the classification is reliable.

Thus some questions on how to design a meta-classifier are:

- What type of base classifiers do we have to learn for meta-classifier, for what type of data?
- What is the role of the accuracy of the base classifiers in the whole scheme?
- How do we have to represent meta-data?
- How can we have to generate meta-data?

4 Experimental Setting

In this subsection we explain at detail some of the blocks in the Fig 2. First the preprocessing we used to represent the T-H pair and second the data used to create the C-E pairs.

The preprocessing we used in each T-H pair is as follows:

- Tokenize.
- Quit stop words.

Normally, an early step of processing is to divide the input text into units called tokens where each is either a *word* or something else like a number or a punctuation mark. This process is referred to as the treatment of punctuation varies.

The system has just stripped the punctuation out. We consider as word any object within the occurrence of a withespace. The withespace is the main clue used in English (RTE benchmark is in English). Finally the system quits any stops words from a stoplist. Common stop words are *the*, *from* and *could*. These words have important semantic functions in English, but they rarely contribute information if the criterion is a simple word-by-word match.

The data we used to collect the frequency of the causal lexical pairs came from sentences which contain the cue phrase *because.*). The sentences were striped in two parts: one corresponding to the cause and one corresponding to its effect to finally form the cause-effect pairs. The sentences were extracted from the Sketch Engine system over a big corpus (ukWAC from the Sketch Engine[1]). The Sketch Engine is a corpus query system which allows the user to view word sketches, thesaurally similar words, and ‘sketch differences’, as well as the more familiar Corpus Query Systems (CQS).

The answers to the questions of how to design a meta-classifier are as follows:

- We used symmetric and non-symmetric measures as base classifiers.
- We chose the best symmetric measure (we optimize accuracy).
- We represented the T-H pairs as a BoW.
- We used as meta-data the RTE Challenge test sets.

For the symmetric base classifier we tested between the cosine, word overlap, and the Bleu algorithm. Thus the cosine measure was the bet of all.

[1] `http://www.sketchengine.co.uk/`

5 Experimental Results

As we see in previous sections we varied the entailment decision in order to prove some differences between the uses of our non-symmetric measure. The experiment 1 was tested over the RTE-1 Challenge test set:

- Experiment 1: The system penalizes a pair if the H→T relation is greater than T→H relation.
- Experiment 2: The system determines the entailment decision based on a meta-classifier.

  The outline of the information displayed on each experiment is the next one:
- Contingency matrix.
- Evaluation matrix.
- Comparison with previous wok.
- Accuracy depending on task.

First, we present the method applied to the RTE-1. The contingency table, Table 3 show how many times the method misclassified the T-H pairs (i.e. *fp* and *tn*) and how many times the method its right. From this table we can obtain some measures to evaluate the entailment decision.

Table 3. RTE-1 contingency matrix

	true	false
true	257	245
false	143	155

Table 3 also shows that our approach tends to say true.

Table 4. RTE-1 evaluation measures

Accuracy	Precision	Recall	F-measure
0.51	0.51	0.64	0.57

From Table 4 this approach obtains a better recall than precision. Therefore the entailment decision got right the proportion of the target items that the system selected.

Table 5. RTE-1 comparison with previous results

Method	Accuracy
GLICKMAN	0.56
LEVENSHTEIN	0.53
C-E	0.51
BLEU	0.49

To compare our approach with previous works we use the accuracy measure (i.e. the most common measure in the RTE Challenge).The proposed measure is compared to non-symmetric measures. We compare out approach with:

- Bleu algorithm RTE baseline [8]
- Probabilistic measure [3]
- Levenshthein modified measure [9]

In Table 5 the results are show. Thus the best one is Glickman. Our measure is the last one compare to the non-symmetric measures. Our measure only outperforms the Bleu algorithm.

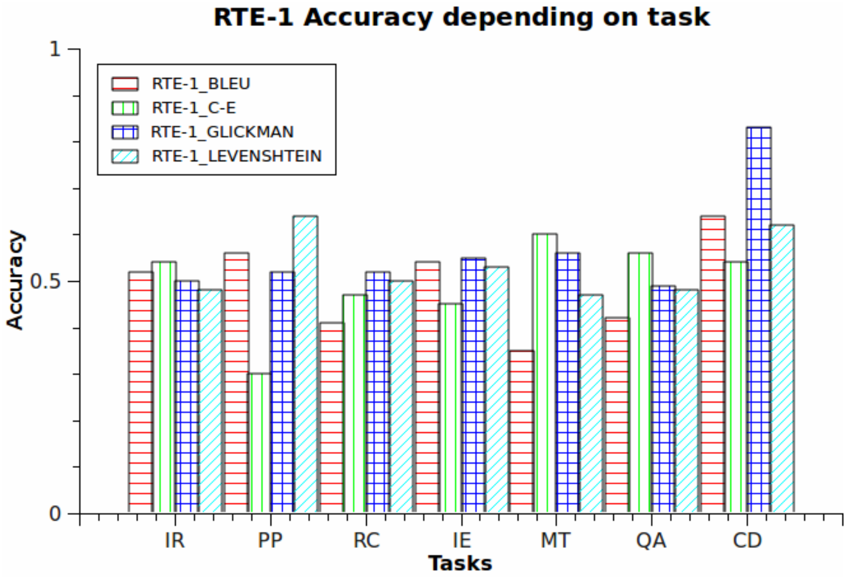


Fig. 3. RTE-1 comparison with previous results by tasks

The results of our approach were the lowest between the non-symmetric measures in general. So if we make a comparison depending on each task. We see that our measure outperforms the other non-symmetric measures in some of the tasks. These tasks are:

- QA.
- IR.
- MT.

The results of the meta-classifier over the RTE Challenge are: In the RTE-1 and RTE-2 the results did not achieve great differences against the Experiment 1. Thus in the RTE-3 the system achieve the best accuracy of all our experiments with 0.61.

In the RTE-3 we achieve the better results for our approach, comparing it to the other results in our research. Thus the results to the RTE-3 are competitive to other participants on the same Challenge.

The percentage of the coverage of the different base classifiers over the RTE-1 development data is as follows: Most of the T-H pairs could be resolved either by the symmetric and the non-symmetric measures (36.62%). Following the examples resolved by the symmetric measure (29.38%) and the non-symmetric at last (14.12%). Finally the 18.88% of the instances could not be resolved by any measure.

Table 6. RTE-3 meta-classifier contengiency matrix

	true	false
true	264	163
false	146	227

Table 7. RTE-3 meta-classifier evaluation measure

Accuracy	Precision	Recall	F-measure
0.61	0.61	0.64	0.63

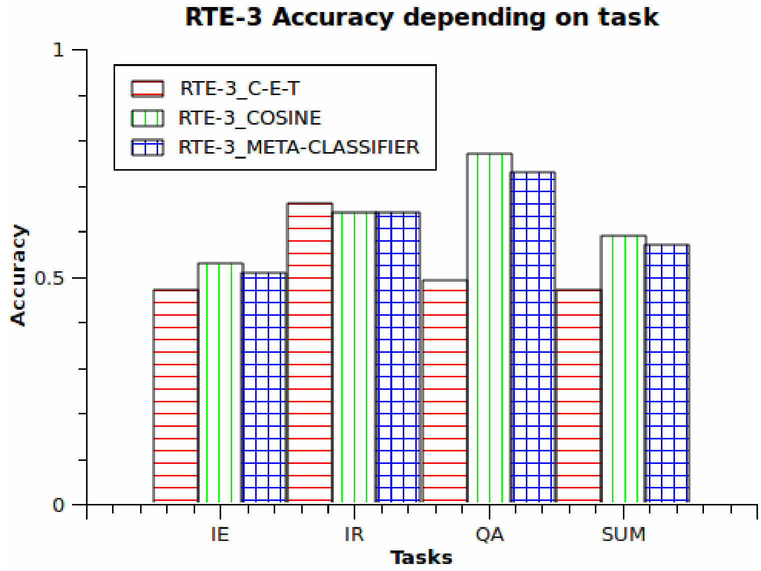


Fig. 4. RTE-3 meta-classifier comparison with base classifiers by tasks

6 Conclusion and Future Work

We proposed a non-symmetric similarity measure to the RTE task. Therefore our unsupervised method is no language dependent.

We have shown that our measure has a lower accuracy than the state of the art methods and outperforms the RTE baseline. These results are significant because they are based on a very simple algorithm that relies on co-occurrences of causal pairs.

We once more confirmed that the web could be used as a lexical resource for RTE (i.e. The Sketch Engine developers have built their corpora from the Web). Also our meta-classifier has a competitive accuracy of 0.61; the average accuracy for the RTE-3 is of 0.61.

In our future work we will explore the use of different meta-features for the meta-classifier, as well as linguistically-motivated meta-features (such as a syntactic unit) and evaluate our method against the RTE machine learning approaches.

References

1. Corley, C., Mihalcea, R.: Measuring the semantic similarity of texts. In: Proceedings of the ACL Workshop on Empirical Modeling of Semantic Equivalence and Entailment, Ann Arbor, pp. 13–18 (June 2005)
2. Dagan, I., Glickman, O.: Probabilistic textual entailment: Generic applied modeling of language variability. In: PASCAL workshop on Text Understanding (2004)
3. De Salvo Braz, R., Girju, R., Punyakanok, V., Frentiu, D.M.: An Inference Model for Word Sense Disambiguation. In: Proceedings of KEPT 2007, Knowledge Engineering Principles and Techniques, Workshop on Recognising Textual Entailment, vol. I (2007)
4. Glickman, O., Dagan, I., Koppel, M.: Web Based Probabilistic Textual Entailment. In: Proceedings of the PASCAL Challenges Workshop on Recognising Textual Entailment (2005)
5. Hobbs, J.R.: Ontological promiscuity. In: Proceedings of the 23rd annual meeting on Association for Computational Linguistics (1985)
6. Inkpen, D., Kipp, D., Nastase, V.: Machine Learning Experiments for Textual Entailment. In: Proceedings of the Second Challenge Workshop Recognising Textual Entailment, Venice, Italy (2006)
7. Kouylekov, M., Magnini, B.: Tree Edit Distance for Recognizing Textual Entailment: Estimating the Cost of Insertion. In: Proceedings of the Second PASCAL Challenges Workshop on Recognising Textual Entailment, Venice, Italy (2006)
8. Pérez, D., Alfonseca, E.: Application of the Bleu algorithm for recognising textual entailments. In: Proceedings of the First Challenge Workshop Recognising Textual Etailment, Southampton, U.K., April 11-13, pp. 9–12 (2005)
9. Tatar, D., Gabriela, S., Andreea-Diana, M., Rada, M.: Textual Entailment as a Directional Relation. Journal of Research and Practice in Information Technology (2009)

Author Index