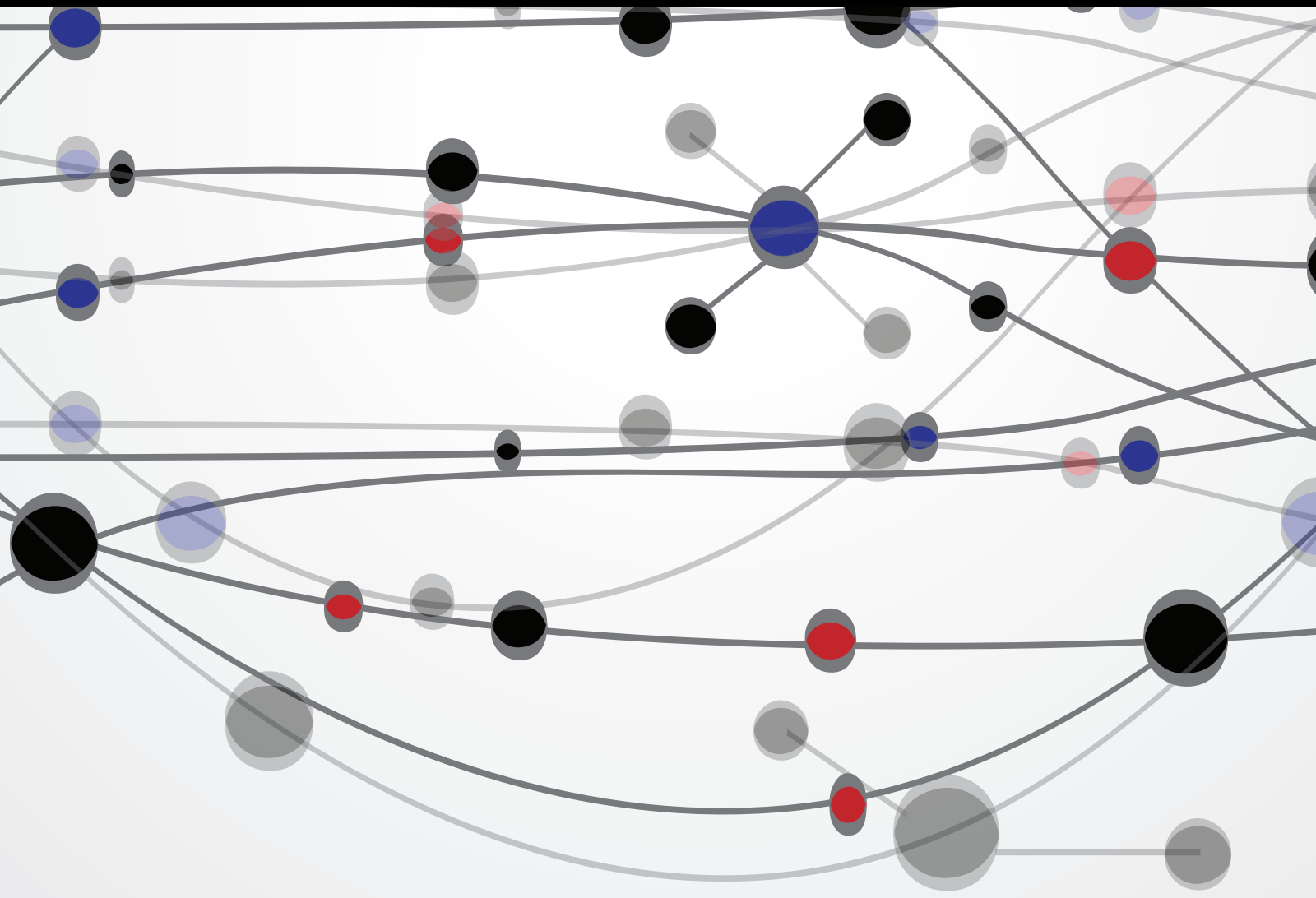
The Scientific World Journal

Emerging Trends in Soft Computing Models in Bioinformatics and Biomedicine

Guest Editors: Yudong Zhang, Saeed Balochian, and Vishal Bhatnagar



Emerging Trends in Soft Computing Models in Bioinformatics and Biomedicine

The Scientific World Journal

Emerging Trends in Soft Computing Models in Bioinformatics and Biomedicine

Guest Editors: Yudong Zhang, Saeed Balochian, and Vishal Bhatnagar

Contents

Emerging Trends in Soft Computing Models in Bioinformatics and Biomedicine, Yudong Zhang, Saeed Balochian, and Vishal Bhatnagar
Volume 2014, Article ID 683029, 3 pages

Attribute Selection Impact on Linear and Nonlinear Regression Models for Crop Yield Prediction, Alberto Gonzalez-Sanchez, Juan Frausto-Solis, and Waldo Ojeda-Bustamante
Volume 2014, Article ID 509429, 10 pages

Kruskal-Wallis-Based Computationally Efficient Feature Selection for Face Recognition, Sajid Ali Khan, Ayyaz Hussain, Abdul Basit, and Sheeraz Akram
Volume 2014, Article ID 672630, 6 pages

Intelligent Screening Systems for Cervical Cancer, Yessi Jusman, Siew Cheok Ng, and Noor Azuan Abu Osman
Volume 2014, Article ID 810368, 15 pages

Mining 3D Patterns from Gene Expression Temporal Data: A New Tricluster Evaluation Measure, David Gutiérrez-Avilés and Cristina Rubio-Escudero
Volume 2014, Article ID 624371, 16 pages

EEG Channel Selection Using Particle Swarm Optimization for the Classification of Auditory Event-Related Potentials, Alejandro Gonzalez, Isao Nambu, Haruhide Hokari, and Yasuhiro Wada
Volume 2014, Article ID 350270, 11 pages

A Fusion Method of Gabor Wavelet Transform and Unsupervised Clustering Algorithms for Tissue Edge Detection, Burhan Ergen
Volume 2014, Article ID 964870, 13 pages

Evolutionary Approach for Relative Gene Expression Algorithms, Marcin Czajkowski and Marek Kretowski
Volume 2014, Article ID 593503, 7 pages

New Fuzzy Support Vector Machine for the Class Imbalance Problem in Medical Datasets Classification, Xiaoqing Gu, Tongguang Ni, and Hongyuan Wang
Volume 2014, Article ID 536434, 12 pages

Chaotic Multiquenching Annealing Applied to the Protein Folding Problem, Juan Frausto-Solis, Ernesto Liñan-García, Mishael Sánchez-Pérez, and Juan Paulo Sánchez-Hernández
Volume 2014, Article ID 364352, 11 pages

Towards Application of One-Class Classification Methods to Medical Data, Itziar Irigoien, Basilio Sierra, and Concepción Arenas
Volume 2014, Article ID 730712, 7 pages

Multicompare Tests of the Performance of Different Metaheuristics in EEG Dipole Source Localization, Diana Irazú Escalona-Vargas, Ivan Lopez-Arevalo, and David Gutiérrez
Volume 2014, Article ID 524367, 9 pages

Adaptive Iterated Extended Kalman Filter and Its Application to Autonomous Integrated Navigation for Indoor Robot, Yuan Xu, Xiyuan Chen, and Qinghua Li
Volume 2014, Article ID 138548, 7 pages

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 683029, 3 pages
http://dx.doi.org/10.1155/2014/683029

Editorial

Emerging Trends in Soft Computing Models in Bioinformatics and Biomedicine

Yudong Zhang,[1] Saeed Balochian,[2] and Vishal Bhatnagar[3]

[1] *School of Computer Science and Technology, Nanjing Normal University, Nanjing, Jiangsu 210023, China*
[2] *Department of Electrical Engineering, Islamic Azad University, Gonabad Branch, Razavi Khorasan, Gonabad 96916-29, Iran*
[3] *Ambedkar Institute of Advanced Communication Technologies and Research, New Delhi 110031, India*

Correspondence should be addressed to Yudong Zhang; zhangyudongnuaa@gmail.com

Received 12 March 2014; Accepted 12 March 2014; Published 15 June 2014

Soft computing (SC) obtains inexact solutions to computationally hard tasks such as the NP-complete problems, for which there is no known algorithm that can obtain an exact solution in polynomial time. SC differs from conventional hard computing (HC) in that SC is tolerant of imprecision, uncertainty, partial truth, and approximation.

Recently, SC has attracted close attention of researchers and has also been applied successfully to solve problems in bioinformatics and biomedicine. Nevertheless, the amount of information from biological experiments and the applications involving large-scale high-throughput technologies is rapidly increasing nowadays. Therefore, the ability of being scalable across large-scale problems becomes an essential requirement for modern SC approaches.

The main objective of this special issue is to provide the readers with a collection of high quality research articles that address the broad challenges in bioinformatics and biomedicine of SCs and reflect the emerging trends in state-of-the-art SC algorithms.

The special issue received 25 high quality submissions from different countries all over the world. All submitted papers followed the same standard (peer-reviewed by at least three independent reviewers) as applied to regular submissions to The Scientific World Journal. Due to the limited space, 14 papers were finally included. The primary guideline was to demonstrate the emerging trends of SC algorithms and applications in bioinformatics and biomedicine.

Y. Xu et al. (Southeast University and Qilu University of Technology) propose an adaptive iterated extended Kalman (AIEKF) to improve the accuracy of data fusion for inertial navigation systems (INS)/wireless sensors networks (WSNs) integrated navigation system. In their mode, the iterated extended Kalman (IEKF) combines the advantages of the AEKF and the IEKF by embedding the noise statistics estimator. Their proposed method is effective to reduce the mean root-mean-square error (RMSE) of position by about 92.53%, 67.93%, 55.97%, and 30.09% compared with the INS-only, WSN, EKF, and IEKF.

The paper authored by S. R. Shahamiri and S. S. B. Salim (University of Malaya) provides a dysarthric multinetworks speech recognizer (DM-NSR) model using a realization of multiviews multilearners approach called multinets artificial neural networks, which tolerates variability of dysarthric speech. Their proposed DM-NSR approach is presented as both speaker-dependent (SD) and speaker-independent (SI) paradigms. The results show that the DM-NSR recorded improve recognition rate by up to 20% and the error rate is reduced by up to 9.32% over the reference model.

J. Frausto-Solis et al. (UPEMOR, UAdeC, and UNAM) propose chaotic multiquenching annealing algorithm (CMQA) that is applied to protein folding problem (PFP). CMQA is divided into three phases: (i) multiquenching phase (MQP), (ii) annealing phase (AP), and (iii) dynamical equilibrium phase (DEP). MQP enforces several stages of quick quenching processes that include chaotic functions. The chaotic functions can increase the exploration potential of solutions space of PFP. AP phase implements a simulated annealing algorithm (SA) with an exponential cooling function. MQP and AP are delimited by different ranges of temperatures; MQP is applied for a range of temperature,

which goes from extremely high values to very high values; AP searches for solutions into a range of temperatures from high values to extremely low values. DEP phase finds the equilibrium in a dynamic way by applying least squares method. CMQA is tested with several instances of PFP.

In the paper by X. Gu et al. (Changzhou University), they present a fuzzy support machine (FSVM) for the class imbalance problem (FSVM-CIP) that can be seen as a modified class of FSVM by extending manifold regularization and assigning two misclassification costs for two class. FSVM-CIP can be used to handle the class imbalance problem in the presence of outliers/noise and enhance the locality maximum margin. Five real-world medical datasets, breast, heart, hepatitis, BUPA liver, and Pima diabetes from the UCI medical database, are employed to illustrate the method presented. Experimental results on these datasets show the outperformed or comparable effectiveness of FSVM-CIP.

A. Gonzalez et al. (Nagaoka University of Technology) propose a method for classifying P300 event-related potentials (ERPs) using a combination of Fisher discriminant analysis (FDA) and a multiobjective hybrid real-binary particle swarm optimization (MHPSO). Their algorithm searches for the set of EEG channels and classifier parameters that simultaneously maximize the classification accuracy and minimize the number of used channels. Results show that their proposed method achieves higher classification accuracy than that achieved by traditional methods while using fewer channels.

A. S. Vieira et al. (University of Vigo) offer a dimensionality reduction technique on text datasets based on a clustering method to group documents, and a simple hidden Markov model to represent them. They apply the new method on the OSHUMED and TREC benchmark text corpora using the k-NN and SVM classifiers. The results obtained are very satisfactory and demonstrate the suitability of their proposed technique for the problem of dimensionality reduction and document classification.

D. Gutiérrez-Avilés and C. Rubio-Escudero (University of Seville) present an evaluation measure for triclusters called mean square residue 3D (MSR_{3D}), based on the homogeneity of the tricluster. The measure is based on the classic biclustering measure, mean square residue (MSR). MSR_{3D} is applied to both synthetic and real data and it has proved to be capable of extracting groups of genes with homogeneous patterns in subsets of conditions and times, and these groups have shown a high correlation level and they are also related in terms of their functional annotations extracted from the Gene Ontology project.

The paper by Y. Jusman et al. (University of Malaya) briefly reviews cervical screening techniques and their advantage and disadvantages. The digital data of the screening techniques are used as data for the computer screening system as replaced in the expert analysis. Four stages of the computer system, enhancement, features extraction, feature selection, and classification, are reviewed in detail. The computer system based on cytology data and electromagnetic spectra data achieves better accuracy than other data.

D. I. Escalona-Vargas et al. (Center for Research and Advanced Studies at Tamaulipas and Cinvestav at Monterrey) study the use of nonparametric multiple comparison statistical tests on the performance of simulated annealing (SA), genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE), when used for electroencephalographic (EEG) source localization. They evaluate the localization's performance in terms of metaheuristics' operational parameters and for a fixed number of evaluations of the objective function. Their results do not show significant differences in the metaheuristics' performance for the case of single source localization. In case of localizing two correlated sources, they find that PSO (ring and tree topologies) and DE perform the worst.

In the paper by M. Czajkowski and M. Kretowski (Bialystok University of Technology), they develop a specialized evolutionary algorithm (EA) for top-scoring pairs called EvoTSP which allows finding more advanced gene relations. They manage to unify the major variants of relative expression algorithms through EA and introduce weights to the top-scoring pairs. Experimental validation of EvoTSP on public available microarray datasets shows that their proposed solution significantly outperforms in terms of accuracy other relative expression algorithms and allows exploring much larger solutions.

A. Gonzalez-Sanchez et al. (ITESM and IMTA) evaluate the most common data-driven modeling techniques applied to yield prediction, using a complete method to define the best attribute subset for each model. Multiple linear regression, stepwise linear regression, M5' regression trees, and artificial neural networks (ANN) are ranked. The models are built using real data of eight crops sowed in an irrigation module of Mexico. To validate the models, three accuracy metrics are used: the root relative square error (RRSE), relative mean absolute error (RMAE), and correlation factor (R). Their results show that ANNs are more consistent in the best attribute subset composition between the learning and the training stages, obtaining the lowest average RRSE (86.04%), lowest average RMAE (8.75%), and the highest average correlation factor (0.63).

I. Irigoien et al. (Euskal Herriko Unibertsitatea UPV-EHU and Universitat de Barcelona) experimentally compare an approach to one class classification (OCC) based on a typicality test with reference state-of-the-art OCC techniques—Gaussian, mixture of Gaussians, naive Parzen, Parzen, and support vector data description—using biomedical datasets. They evaluate the ability of the procedures using twelve experimental datasets with no necessarily continuous data. The results of the comparison show the good performance of the typicality approach, which is available for high dimensional data; it is worth mentioning that it can be used for any kind of data (continuous, discrete, or nominal), whereas state-of-the-art approaches application is not straightforward when nominal variables are present.

S. A. Khan et al. (Shaheed Zulfikar Ali Bhutto Institute of Science and Technology) use two well-known methods, discrete wavelet transform (DWT) and weber local descriptor (WLD) to extract the face discriminative features. First, for both types of features, the recognition accuracy is separately measured. In the next step, both types of features are fused using the concatenation method to improve the accuracy

rate. To select more discriminative features and reduce data dimensions, computationally efficient algorithm (Kruskal-Wallis) is used. In the last step, three-classifier (SVM, KNN, and BPNN) ensemble is developed to improve the accuracy rate. Their proposed technique is more efficient in terms of time complexity as compared to GA and PSO. Yale face database is used for all experiments. Their proposed technique is highly robust to facial variations like occlusion, illumination, and expression change and computationally efficient as compared to existing methods.

Finally, B. Ergen (Firat University) proposes two edge detection methods for medical images by integrating the advantages of Gabor wavelet transform (GWT) and unsupervised clustering algorithms. The GWT is used to enhance the edge information in an image while suppressing noise. Following this, the k-means and Fuzzy c-means (FCM) clustering algorithms are used to convert a gray level image into a binary image. Their proposed methods are tested using medical images obtained through computed tomography (CT) and magnetic resonance imaging (MRI) devices and a phantom image. The results prove that their proposed methods are successful for edge detection, even in noisy cases.

Acknowledgments

We would like to express our gratitude to all of the authors for their contributions and to the reviewers for their effort in providing constructive comments and feedback. We hope this special issue offers a comprehensive and timely view of the area of emerging trends in soft computing models in bioinformatics and biomedicine and that it will offer stimulation for further research.

Yudong Zhang
Saeed Balochian
Vishal Bhatnagar

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 509429, 10 pages
http://dx.doi.org/10.1155/2014/509429

Review Article

Attribute Selection Impact on Linear and Nonlinear Regression Models for Crop Yield Prediction

Alberto Gonzalez-Sanchez,[1] Juan Frausto-Solis,[2] and Waldo Ojeda-Bustamante[1]

[1] *IMTA, Boulevard Cuauhnáhuac 8532, Colonia Progreso, 62550 Jiutepec, MOR, Mexico*
[2] *UPEMOR, Boulevard Cuauhnáhuac 566, Colonia Lomas del Texcal, 62550 Jiutepec, MOR, Mexico*

Correspondence should be addressed to Juan Frausto-Solis; juan.frausto@upemor.edu.mx

Received 6 December 2013; Accepted 10 February 2014; Published 26 May 2014

Academic Editors: S. Balochian and Y. Zhang

Efficient cropping requires yield estimation for each involved crop, where data-driven models are commonly applied. In recent years, some data-driven modeling technique comparisons have been made, looking for the best model to yield prediction. However, attributes are usually selected based on expertise assessment or in dimensionality reduction algorithms. A fairer comparison should include the best subset of features for each regression technique; an evaluation including several crops is preferred. This paper evaluates the most common data-driven modeling techniques applied to yield prediction, using a complete method to define the best attribute subset for each model. Multiple linear regression, stepwise linear regression, M5′ regression trees, and artificial neural networks (ANN) were ranked. The models were built using real data of eight crops sowed in an irrigation module of Mexico. To validate the models, three accuracy metrics were used: the root relative square error (RRSE), relative mean absolute error (RMAE), and correlation factor (R). The results show that ANNs are more consistent in the best attribute subset composition between the learning and the training stages, obtaining the lowest average RRSE (86.04%), lowest average RMAE (8.75%), and the highest average correlation factor (0.63).

1. Introduction

Crop yield prediction (CYP) is important for agricultural planning and resource distribution decision making [1]. Regrettably, CYP is a difficult task because many variables are interrelated [2]. Yield is affected by human producer decisions or activities (such as irrigated water, land, and crop rotation) or incontrollable factors (such as weather). Commonly, cropping planners use the previous yield as an estimation of future yield. Nevertheless, crop yield varies spatially and temporally with a nonlinear behavior, introducing large deviations from one year to another [3]. Thus, more efficient methods for CYP have been developed, in which crop growth and data-driven models are the most popular. Crop growth models, using site-specified experimental data, regional calibration, and plot level observations, are recognized as robust and efficient models. However, they are available only for some crops, with development time and cost being extremely large [3]. On the other hand, data-driven models work with high-level information and are built empirically without a deep knowledge about physical mechanisms which produce the data. Previous works suggest that data-driven models have better adaptability for cropping planning than crop growth methods due to their friendly implementation and performance [4].

Data driven models are widely applied using classical statistics and data-mining methods. Statistical models use parametric structures tuned with sum-of-squares residuals, validated by hypotheses test and confidence intervals. Most of the statistical applications for CYP have been linear [3], obtaining a range from bad to moderate results. Data mining applies machine learning techniques and nonparametric structures, in which validation uses prediction accuracy. Machine learning (ML) obtains nonlinear models from massive datasets [5]. Most common ML techniques for CYP are regression trees [6] and neural networks [3, 4, 7]. Despite the high site dependency, neural networks are widely recognized as robust models, obtaining good results for CYP

[7]. Comparisons between linear and nonlinear models for CYP show a small advantage in favor of nonlinear models [3, 7]. However, the attribute subset is usually the same for all the evaluated techniques. In practice, the explanatory attributes are selected from expertise assessment or previous publications, for instance, [3, 8]. However, the explanatory attributes may have a different impact on each technique, even using the same dataset [9]. A fairer comparison should include the best attribute subset for each technique, selected with some performance metrics [10]. Regrettably, only an exhaustive approach can guarantee the optimal subset for all regression techniques. Some CYP datasets have relatively few attributes and an exhaustive approach can be applied to model comparison purposes [8].

In this paper, a comparison between linear and nonlinear data-driven modeling techniques for CYP is presented. The best attribute subset for each technique is determined by measuring the predictive accuracy of each model subset. To obtain the optimal subset, a recursive algorithm finds all the feature combinations, building a regression model of each subset. The models are built using most samples of training datasets, leaving the more recent to measure the performance. The best subset for each technique is tested with samples representing future information which had not been included in the training stage. The most common techniques for CYP were compared: multiple linear regression, stepwise linear regression, M5′ regression trees, and perceptron multilayer neural networks. Results per technique are compared against those obtained using the optimal attribute combination derived from the test dataset. The potential attributes considered for this work were irrigation water depth (mm), accumulated rainfall (mm), solar radiation (MJ/m^2), maximum and minimum temperatures (°C), relative humidity (%), and the farm location. To build the models, historical data of eight crops were obtained from one irrigation module located in Mexico. Results show the best CYP technique, the most influential attributes for each model, and the fact that an exhaustive approach on the training dataset does not guarantee optimality on testing dataset.

This paper is organized as follows. Section 2 describes data sources, data-driven techniques, accuracy metrics, and the recursive algorithm used to build and test the models. Section 3 shows the experimental results and discussion. Finally, Section 4 presents the conclusions about realized work.

2. Materials and Methods

2.1. Data Description. This paper uses data obtained from the irrigation district 075 (Santa Rosa III-1 module) in Sinaloa, Mexico (one of the largest and most productive districts in the country). Two data sources from the year 1999 to 2007 were collected: (a) agricultural production data and (b) weather information data. The former included attributes regarding sowed areas, crop types, quantity of irrigated water, starting and ending sowing dates, and crop yield. Such data were obtained from Spriter-GIS system [11]. The second data source includes climatological variables measures such as rainfall, solar radiation, and temperatures. Weather data were collected from the National Meteorological Service (SMN) stations located in the module vicinity. The CRISP-DM [5] methodology was applied to clean, homogenize, and integrate both data sources into one single database, obtaining eight crop representative datasets. Eight potential attributes (Table 1) were selected based on previous CYP works [12] and the data availability. Such attributes are referred to as *potential* because this work uses a complete algorithm to find the best attribute subset for each regression technique. Thus, the final subset of attributes depend on the algorithm execution. Average of weather attributes (solar radiation, temperatures, and humidity) was estimated with the last three crop growing stages, the most influential in the crop development.

Table 1: Potential attributes in crop datasets.

Attribute code name	Attribute description
SP	Section (farm location where crop was sowed)
IWD	Irrigation water depth applied (mm)
SGR	Solar radiation (M-Joules/m^2)
RF	Rainfall (mm)
Max$T$	Maximal temperature (°C)
Min$T$	Minimal temperature (°C)
RH	Relative humidity in leafs (%)

The crop datasets are described in Table 2. To simplify future references of these datasets, an ID is assigned to each one (which is shown in the first column). Table 2 describes the quantity of records and periods of time used for the training and testing stages. In order to maintain realistic conditions, the last year of available data was reserved for testing.

2.2. Data-Driven Modeling Techniques. The most common data-driven techniques applied to CYP were selected for this work: multiple and stepwise linear regression [3, 7], M5′ regression trees [2, 8, 13], and artificial neural networks [1, 3, 7, 12].

2.2.1. Multiple and Stepwise Linear Regression. Multiple linear regression (MLR) is a popular technique which can be applied to predict a dependent variable Y_i, using a set of independent variables X_{ij}. MLR model is described by [14]

$$Y_i = \sum_{j=1}^{k} B_j X_{ij} + \epsilon_i, \quad (1)$$

where k is the number of independent variables, B_j is a regression coefficient, X_{ij} is the j value for the observation i, and ϵ_i is the residual error. If $X^T X$ is a nonsingular matrix, an approximation for $B(\overline{\beta})$ can be obtained by $\overline{\beta} = (X^T X)^{-1} X^T Y$. Then (1) can be rewritten as $Y = X\overline{\beta} + \epsilon$.

Stepwise linear regression (SLR) works with the same principle. However, SLR performs a semiautomated selection on independent variables to maximize the model's prediction

Table 2: Testing and training samples distribution per crop dataset.

Dataset ID	Crop species	Cultivar	Training period	Training samples	Testing period	Testing samples
PJ01	Pepper (*Capsicum annuum*)	Jalapeno	1999–2005	116	2006	18
CBP02	Common bean (*Phaseolus vulgaris*)	Peruano	1999–2006	361	2007	9
CBA03	Common bean (*Phaseolus vulgaris*)	Azufrado	1999–2006	120	2007	21
CBM04	Common bean (*Phaseolus vulgaris*)	Mayocoba	1999–2006	332	2007	27
CP05	Corn (*Zea mays*)	Pioneer 30G54	2000–2005	179	2006	19
PA06	Potato (*Solanum tuberosum*)	Alpha	1999–2006	1749	2007	116
PA07	Potato (*Solanum tuberosum*)	Atlantic	1999–2006	1062	2007	92
TS08	Tomato (*Lycopersicon esculentum* Mill.)	Saladette	1999–2005	182	2006	15

efficiency. Linear regression is performed by adding or removing independent variables on each iteration. Initially, the variable with the highest correlation (R-squared) measured with respect to the dependent variable is included. Then, the remaining independent variable with the highest correlation with respect to the dependent variable is selected. This iterative process is repeated while the addition of a remaining independent variable increases R-squared with a significant quantity. We use the SLR implementation in SPSS [15], which combines forward selection and backward elimination [16]. At each step, the best remaining variable is added according to a significance criterion α of five percent; then the entire set of variables is reviewed to decide whether a single variable is removed using an α of ten percent.

2.2.2. Regression Trees. A regression tree (RT) is based on a decision tree, a classifier expressed as a recursive partition of the samples' space [17]. A tree is formed by nodes, in which the first is named the root node (without incoming edges). All the other nodes have exactly one incoming edge. A node with outgoing edges is called a test node and a node without outgoing edges is called a leaf node. Each internal node in the tree splits the samples' space into two or more subspaces based on conditions of the input attributes values. In the case of numerical attributes the condition refers to a range of values. Each leaf is assigned to one class representing the most appropriate target value. Samples are classified by navigating them from the root of the tree down to a leaf, according to the outcome of the tests along the path. For regression trees, the class at the leaf nodes assigns a numerical value to the tested sample which corresponds to the value predicted by the regression model. The most common algorithms to build RTs are CART, M5, and M5′ [17]. This work uses M5′ algorithm implemented by Weka [17]; the standard deviation reduction (SDR) is applied as a measure of impurity on continuous attributes. The parameters to build an RT with a minimum of two samples by node, pruned and smoothed, were selected.

2.2.3. Artificial Neural Networks. From a structural point of view, an artificial neural network (ANN) is a collection of simple processing units linked via directed and weighted interconnections. Each processing unit receives a number of inputs from the outside or other processing units. Each input is calibrated based on the weights of their interconnections. Once calibrated, inputs are combined and transmitted to other processing units via the appropriate interconnections. The units are organized by layers, hiding the intermediate layers to the user. This process is represented by a nonadditive and nonlinear function that maps the set of inputs to a set of outputs [17]. The training stage is an iterative process performed to pound connections and it is guided for error measure. There are many ANN topologies and training algorithms. This work uses the most popular topologies and learning algorithm combinations: multilayer perceptron (MLP) and backpropagation algorithm [17]. MLP network has been a popular choice for CYP [1, 3]. Backpropagation algorithm minimizes the error function using the gradient descent method. The combination of weights obtained is a solution of the learning problem. Since this method requires computation of the gradient of the error function at each iteration step, the continuity and differentiability of this function should be verified. In addition, an activation function is required where the sigmoidal function $(1/(1 + e^{-cx}))$ is commonly used [17]. In this work, a topology with three layers and 10 neurons on a single hidden layer was used; this topology was applied in other works [4]. The most recommended parameters were applied such as the weight decay and numeric attribute normalization [3]. Training epochs, learning rate, and the momentum were established by experimentation, being 1000, 0.3, and 0.01, respectively. Quantity of neurons at the input layer depends on number of attributes (see Section 2.5), while the output layer has only a neuron (CYP estimation).

2.3. Accuracy Metrics. We use three of the most common metrics of regression models [5]: the root relative square error (RRSE), correlation factor (R), and the relative mean absolute error (RMAE). RRSE compares the model prediction against the mean, which is frequently used to supply the crop yield value. An RRSE less than 100% indicates a prediction that is better than the average value. Correlation factor (R) measures the linear relationship between regression model predictions and the real values. Mean absolute error (MAE) is the average of estimation differences (in physical units). This metric is expressed as a percentage relative to the mean yield, being called RMAE instead of MAE. Equation (2) shows how these metrics are calculated, where y is the real yield value, $\hat{y}$ represents the yield estimation, i is the number of sample, $\overline{y}$

```
Function to obtain the best attribute subset on training samples.
Inputs: samples (a set of training samples), potAttr (set of potential attributes), algorithm (MLR,
M5′ or ANN), minYear, maxYear (minimum and maximum year in the training dataset).
resultList is a dynamic list and a global variable. Each entry in this list has the form <testAttr,
metricMeasures>,
where testAttr is an attribute subset and metricMeasures are the metrics results obtained from an
model's evaluation made with attributescontained in testAttr
Function findBestAttrSubset(samples, potAttr, algorithm, minYear, maxYear) {
clearList(resultList)
   localSamples = extract samples from samples in the range [minYear, maxYear − 1]
   validSamples = extract samples from samples with year equals to maxYear
   for i = 0 to sizeOf(potentialAttr) begin
      testAttr = create an empty set of attributes
      testAttr = testAttr ∪ potAttr_i
      // call a recursive procedure to evaluate all attribute subsets starting from the i attribute in
potAttr
      testAttrCombination(potAttr, testAttr, localSamples, validSamples, algorithm)
   end_for
   return the results at the top of resultList
end_function

// Recursive procedure to evaluate an attribute combination
// Inputs: potAttr, testAttr (a set of attributes); trainSamples, validSamples (a set of samples),
algorithm (a regression algorithm).
procedure testAttrCombination(potAttr, testAttr, trainSamples, validSamples, algorithm)
Begin
   // make a regression model of algorithm type using trainSamples and testAttr
   model = makeRegressionModel(algorithm, trainSamples, testAttr)
   // evaluate a regression model using validSamples
   metricMeasures = evalModel(model, validSamples)
   // add the attribute subset and the metric measures in the sorted result list
   addResults(resultList, ⟨testAttr, metricMeasures⟩)
    index = obtain the highest position of one element of testAttr in potAttr
    for i = index + 1 to length(potAttr) begin
      // add the potential attribute i to testAttr
      testAttr = testAttr ∪ potAttr_i
      // recursive call
      testAttrCombination(potAttr, testAttr, trainSamples, validSamples, algorithm)
      // remove the potential attribute i from testAttr
      testAttr = testAttr − potAttr_i
   end_for
end_procedure
```

Algorithm 1: Recursive algorithm to perform the optimal attribute subset search.

is the average of the real yield values, and $\overline{\widehat{y}}$ is the average of predictions:

$$\text{RRSE}\,(\%) = \sqrt{\frac{\sum_{i=1}^{n}\left(y_i - \widehat{y}_i\right)^2}{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)^2}} \times 100,$$

$$R = \frac{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)\left(\widehat{y}_i - \overline{\widehat{y}}\right)}{\sqrt{\sum_{i=1}^{n}\left(y_i - \overline{y}\right)^2}\sqrt{\sum_{i=1}^{n}\left(\widehat{y}_i - \overline{\widehat{y}}\right)^2}}, \tag{2}$$

$$\text{RMAE}\,(\%) = \left(\frac{\sum_{i=1}^{n}\left|y_i - \widehat{y}_i\right|}{(n)\left(\overline{y}\right)}\right) \times 100.$$

Many CYP works use root mean squared error (RMSE) as accuracy metric. RMSE measures the difference between real and estimations values, exaggerating the presence of outliers [5]. We use RRSE instead of RMSE because the former applies the average value as common reference point, being easy to understand by people unaccustomed to physical crop yield dimensions.

2.4. Method to Find the Best Attribute Subset. A combinatorial procedure to perform a complete enumeration of all the subsets $\{\mathbf{x_1}, \mathbf{x_2}, \mathbf{x_3}, \ldots, \mathbf{x_m}\}$ is presented in this paper. The procedure starts with a potential set of attributes $A = \{a_1, a_2, \ldots, a_n\}$, such that each $\mathbf{x_1}$ is a subset of A. Each $\mathbf{x_k}$ subset is evaluated using the training dataset, which is

Table 3: RRSE, R, and RMAE measures using the OAS on testing dataset.

Crop dataset	RRSE (%)			R			RMAE (%)		
	MLR	M5′	ANN	MLR	M5′	ANN	MLR	M5′	ANN
PJ01	50.69	29.29	49.62	0.87	0.96	0.88	8.63	4.56	8.27
CBP02	52.14	58.85	58.05	0.67	0.68	0.67	5.67	6.40	6.41
CBA03	63.40	38.66	38.66	0.94	0.93	0.93	4.72	3.62	3.62
CBM04	70.53	71.20	75.04	0.69	0.59	0.58	1.30	1.59	1.58
CP05	87.83	83.52	87.59	0.72	0.65	0.70	8.13	6.39	8.46
PA06	95.28	74.02	86.16	−0.13	0.63	0.54	25.58	20.05	23.13
PA07	95.84	88.14	91.24	0.60	0.51	0.45	17.78	16.42	17.40
TS08	86.59	82.40	74.87	0.69	0.64	0.73	11.08	13.46	14.57
Average	75.29	65.76	70.15	0.63	0.70	0.72	10.36	9.06	10.43

divided in two datasets. The majority of samples are used to build the models, while the most recent ones are applied for performance measurement. In CYP context, if the [**a**, **b**] year range of historical data is available for training, the [**a**, **b** − **1**] range is really used for training, and data from year **b** is reserved for validation. The model's validation is made using the metrics described in Section 2.3. Each validation result and the related attribute subset are registered in a sorted list according to these metrics. Ties are solved in the following order: RRSE (lower), R (higher), and RMAE (lower). At the end of the process, the subset at the top is taken as the best. Algorithm 1 shows the algorithm of the optimal attribute search process.

The function *evalModel(model, validSamples)* of Algorithm 1 evaluates the argument *model* with samples taken from the *validSamples* dataset. This function uses the percentage-split validation scheme approach [5]. We tried other validation schemes as well, such as training and validating the models with the entire training dataset and cross-validation (CV). The former provided very poor results to predict the yield of future samples. On the other hand, CV (considered a robust validation scheme) was difficult to apply because (1), for k subsets required for CV, $k - 1$ models should be stored, and (2) the computational cost of the entire process is increased $k - 1$ times for each evaluation [3], being not computationally tractable in practical applications.

2.5. Distance to the Optimal Attribute Subset (OAS). Algorithm 1 can be applied to both the learning and testing stages. When this algorithm is applied to the former, a ranking of attribute combinations is obtained, placing the best attribute combination at the top. This subset is named the learning attribute subset (LAS). In testing stage, the algorithm is applied to the union of the training and the testing datasets, obtaining a rank of attribute subsets. In this last case, the subset at the rank's top is named the optimal attribute subset (OAS). Evidently, this last rank cannot be available in practice, because testing dataset represents unseen samples from the future. However, the rank of attribute combinations that originated the OAS can be used to define a new performance metric, which should be used only for evaluation purposes. Let x be an attribute subset and D the number of combinations that separates the OAS results from the x subset results. Then D can be used as a performance measure of x. We called measure D the "distance to the optimal attribute subset."

3. Results and Discussion

Experimental results are presented in the next three sections. Section 3.1 shows the metric measures obtained in testing dataset with the OAS. Section 3.2 shows the metric measures using the potential attributes. Section 3.3 describes the results using the LAS on testing dataset.

3.1. Metric Measures Using the OAS on Testing Dataset. The OAS for the testing dataset to each crop technique was obtained with the algorithm of Section 2.4. Table 3 shows every metric obtained per technique (RRSE, R, and RMAE). RRSE shows that all techniques achieve better predictions than the average. For the potato crop datasets (PA06 and PA07), MLR obtains only slightly better results than the average (RRSE of 95%). In general, nonlinear techniques show some improvements over MLR, introducing small RRSE measures and R values near to 0.7.

Tables 4(a), 4(b), 4(c), and 4(d) show the OAS composition found for each regression technique. The attributes in OAS are shown in shaded cells. Evidently, OAS is the same for SLR and MLR (Tables 4(a) and 4(b), resp.). Attributes selected are grouped in Table 5, which shows the quantity of times that a particular attribute is included in the OAS for each crop dataset. The average column in Table 5 indicates that the IWD, RH, SGR, and MINT attributes appear in more than half of optimal crop yield models, mostly independent of the regression technique. Besides, IWD (irrigation water depth) was the attribute most selected by all techniques.

Because attributes selected can be influenced by temporal elements, Figure 1 shows the results obtained only with the five crop testing datasets of year 2007. Attributes most frequently selected by MLR technique were MinT and IWD, with the latter always included in the OAS. Attributes most frequently selected by M5′ were MinT and RH, with the latter always included in the OAS. On the other hand, attributes selected by ANN technique were IWD and RF. Unlike other techniques, ANN did not always select a specific attribute.

Table 4: Attributes in OAS and LAS selected by each crop technique. Attributes in optimal subset (OAS) are remarked with asterisks. Attributes selected with training data (LAS) are identified with the √ symbol.

(a) SLR

Crop dataset	Attributes						
	SP	IWD	SGR	RF	Max$T$	Min$T$	RH
PJ01	√	*	√*	√			*
CBP02		*		√*	√*	*	√*
CBA03	√*	*				√*	
CBM04	√*	*	*	√			
CP05	*	√*	√*	*		√*	*
PA06	√	√*	*	√*	*	√*	*
PA07		√*	√*	*	√*	*	
TS08	√*	*	*	√*	*		*
Count (OAS)	4	8	6	5	4	5	6
Count (LAS)	5	3	3	5	2	3	1

(b) MLR

Crop dataset	Attributes						
	SP	IWD	SGR	RF	Max$T$	Min$T$	RH
PJ01	√	√*	*	√	√	√	*
CBP02		√*	√	√*	√*	√*	√*
CBA03	√*	√*	√	√	√	√*	√
CBM04	*	*	√*	√	√		
CP05	√*	√*	√*	*		√*	*
PA06		√*	√*	*	*	√*	√*
PA07		√*	√*	*	√*	*	
TS08	√*	*	√*	*	√*		*
Count (OAS)	4	8	6	5	4	5	6
Count (LAS)	4	6	7	4	6	5	3

(c) M5′

Crop dataset	Attributes						
	SP	IWD	SGR	RF	Max$T$	Min$T$	RH
PJ01	√*	√*	√		√*	√	√*
CBP02	√*		√*	√	√*	*	√*
CBA03	√		√		√	√*	*
CBM04	√	√	√	*	√		*
CP05	√*	*	√*				
PA06	√	√*		√	*	√*	*
PA07	*	√*			√	√*	*
TS08	√*	*	*	*	√	√	
Count (OAS)	5	5	3	2	3	4	6
Count	7	4	5	2	6	5	2

(d) ANN

Crop dataset	Attributes						
	SP	IWD	SGR	RF	Max$T$	Min$T$	RH
PJ01	√	*	√*	√	*	*	*
CBP02				√*	√*		√*
CBA03	√*	*				√*	
CBM04	√		*	√*			
CP05		√*	√*			√*	
PA06	√*	√*		√*		√	*
PA07		√*	√*		√		
TS08	√*		*	√			*
Count (OAS)	3	5	5	3	2	3	4
Count (LAS)	5	3	3	5	2	3	1

Table 5: Quantity of crop yield models where attributes appear as optimal.

Attribute	Regression technique			Average
	MLR	M5′	ANN	
SP	4	5	3	4.00
IWD	8	5	5	6.00
SGR	6	3	5	4.67
RF	5	2	3	3.33
Max$T$	4	3	3	3.33
Min$T$	5	5	3	4.33
RH	5	6	4	5.00

3.2. Metric Measures Using All the Potential Attributes. Table 6 shows the RRSE, R, and RMAE measures using all the potential attributes as explanatory variables. RRSE indicates that only two of the eight crop models per technique obtain better predictions than the mean yield value. MLR has three models with good predictions. However, PA06 model shows an R value of 0.07, indicating a very low linear relationship between the prediction and the real yield. The models for the PJ01 dataset are the most consistent, showing good results with every technique and a small improvement with nonlinear models. For every technique, the set of RRSEs lower than one hundred percent was averaged; in the case of Table 6, the figures obtained were 94.41, 74.34, and 75.79 for MLR, M5′, and ANN, respectively. The averages with the entire set of RRSEs were also calculated and shown in the row named *Average (all)* of Table 6. We decided to average the RMAEs with an RRSE lower than one hundred percent and an R factor close to one and greater than a threshold value, set as 0.6 in this work. As is well known, a good prediction model should have a low RRSE and an R value close to 1. Therefore, for all the potential attributes and when only RRSE and R are considered, we can observe that M5′ is the best technique. Averaging the RMAEs that accomplish this criterion (RRSE $< 100\%$ and an $R > 0.6$), the best techniques were again M5′ and ANN.

The distance D to the optimal attribute subset (described in Section 2.5) provides an idea of how far are the OAS results to those obtained with all the potential attributes. Table 7 shows D values for the evaluated techniques, which indicates that very few models are close to the optimal results using all the potential attributes. Considering all the 256 possible combinations, most of the obtained results with all the attributes are located beyond the middle of the rank of combinations.

Table 6: RRSE, R, and RMAE measures using all the potential attributes.

Crop dataset	RRSE (%)			R			RMAE (%)		
	MLR	M5′	ANN	MLR	M5′	ANN	MLR	M5′	ANN
PJ01	**85.36**	**48.83**	**65.51**	**0.89**	**0.9**	**0.92**	**14.21**	**6.99**	**9.28**
CBP02	**99.85**	**99.85**	124.23	**0.63**	**0.63**	0.64	**10.25**	**10.25**	13.21
CBA03	136.96	156.29	**86.07**	0.76	0.77	**0.59**	14.99	15.33	**7.83**
CBM04	470.62	262.08	350.32	−0.66	−0.66	−0.68	11.2	6.54	8.05
CP05	102.68	362.5	123.61	0.36	0.08	0.54	10.12	32.25	11.75
PA06	**98.02**	102.87	110.24	**0.07**	0.15	0.19	**26.02**	27.56	27.93
PA07	110.86	165.41	113.18	−0.03	−0.13	−0.18	20.67	37.07	24.23
TS08	166.86	100.56	146.6	0.45	0.28	0.09	32.83	19.95	43.57
Average (RRSE < 100)	94.41	74.34	75.79	0.53	0.77	0.76	16.83	8.62	8.55
Count (<100)	3	2	2				3	2	2
Average (all)	158.9	162.3	139.97	0.31	0.25	0.26	17.54	19.49	18.23

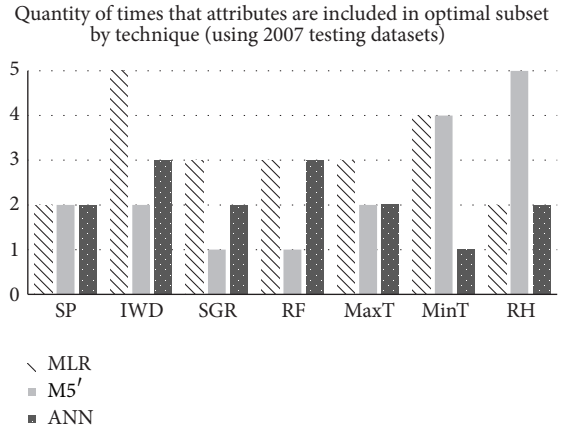


Figure 1: Quantity of occurrences of each attribute in the OAS for each technique (only crop datasets with 2007 testing data).

Table 7: Distance from OAS error measures using the potential attribute set.

Crop dataset	Distance from optimal (combinations)		
	MLR	M5′	ANN
PJ01	38	14	18
CBP02	80	189	135
CBA03	111	118	32
CBM04	231	135	216
CP05	71	206	196
PA06	23	173	186
PA07	230	194	213
TS08	229	69	185
Average	127	137	148

3.3. Metric Measures Using the LAS on Testing Dataset. The LAS for each crop technique was obtained during the learning stage using only data from each training dataset. The models built with the LAS were applied to predict the yield of samples on testing dataset. In addition, SLR has its own attribute selection mechanism and is included in this section. RRSE and R measures in Table 8 show the obtained results. Our attribute selection algorithm (Section 2.4) improves the MLR, M5′, and ANN models performance, increasing the number of CYP models with RRSE measures lower than 100% (MLR obtained five models, while M5′ and ANN obtained six models each). SLR has a poor performance, with only one model with an RRSE measure lower than 100%, and an average error even higher than MLR using all the potential attributes (Table 9). RRSE of nonlinear techniques shows small improvements with respect to MLR. In addition, R measures of MLR are higher than those obtained by M5′ and inferior to those obtained with ANN. Average R values for the models with RRSE lower than 100% are greater than 0.5. Only nonlinear techniques obtained an average RRSE value lower than 100%. Among these, ANN obtained better results, with the lowest RRSE, the highest R, and the lowest RMAE measures.

Table 9 shows that MLR, M5′, and ANN model errors built with the LAS decrease considerably when they are compared against the use of all the potential attributes. Average RMAE values decreased as follows (in %): (a) for MLR, from 17.54 to 13.59; (b) for M5′, from 19.49 to 11.93; and (c) for ANN, from 18.23 to 12.89. The average RMAE measure for SLR was scored worse than MLR using all the potential attributes, obtaining an RMAE of 18.65%. Table 10 shows the distances D of the error measure obtained from the LAS to the OAS results. ANN is the regression technique with more CYP models closer to the optimal. MLR and M5′ present similar average results. SLR was too far to optimal combination.

Tables 4(a), 4(b), 4(c), and 4(d) show the attributes contained in each LAS grouped per technique. These attributes are identified with a $\surd$ symbol. As it can be seen, LAS and OAS differ, showing that the best attribute subset can vary from one year to another. Nevertheless, such behavior is different for each technique. Let us illustrate the situation with Figures 2(a), 2(b), and 2(c), which show the frequency when an attribute appears in the LAS and OAS for the MLR, M5′, and ANN regression techniques (resp.). To avoid mixing results from different years, these charts only include attribute subsets obtained from the crop datasets with 2007 testing

Table 8: RRSE and R measures using the LAS on testing dataset.

Crop dataset	RRSE (%)				R			
	SLR	MLR	M5′	ANN	SLR	MLR	M5′	ANN
PJ01	203.86	**81.90**	**58.00**	**75.25**	0.87	**0.81**	**0.90**	**0.82**
CBP02	130.52	**55.05**	**74.67**	**58.05**	0.66	**0.52**	**0.73**	**0.67**
CBA03	98.76	136.96	112.45	**58.40**	0.64	0.76	−0.05	**0.98**
CBM04	479.43	306.29	**85.30**	**78.96**	−0.67	0.66	**0.27**	**0.61**
CP05	103.77	**91.06**	**94.50**	**87.59**	**0.50**	**0.69**	**0.52**	**0.70**
PA06	102.41	102.36	**85.96**	101.33	−0.42	−0.32	**0.55**	0.11
PA07	110.44	**97.49**	101.31	**91.24**	−0.06	**0.67**	0.09	**0.45**
TS08	112.85	**86.59**	**82.40**	137.48	0.42	**0.69**	**0.64**	0.69
Average (RRSE < 100)	98.76	82.41	80.14	74.92	0.50	0.67	0.60	0.71
Count (<100)	1	5	6	6				
Average (all)	167.76	119.71	86.82	86.04	0.24	0.56	0.46	0.63

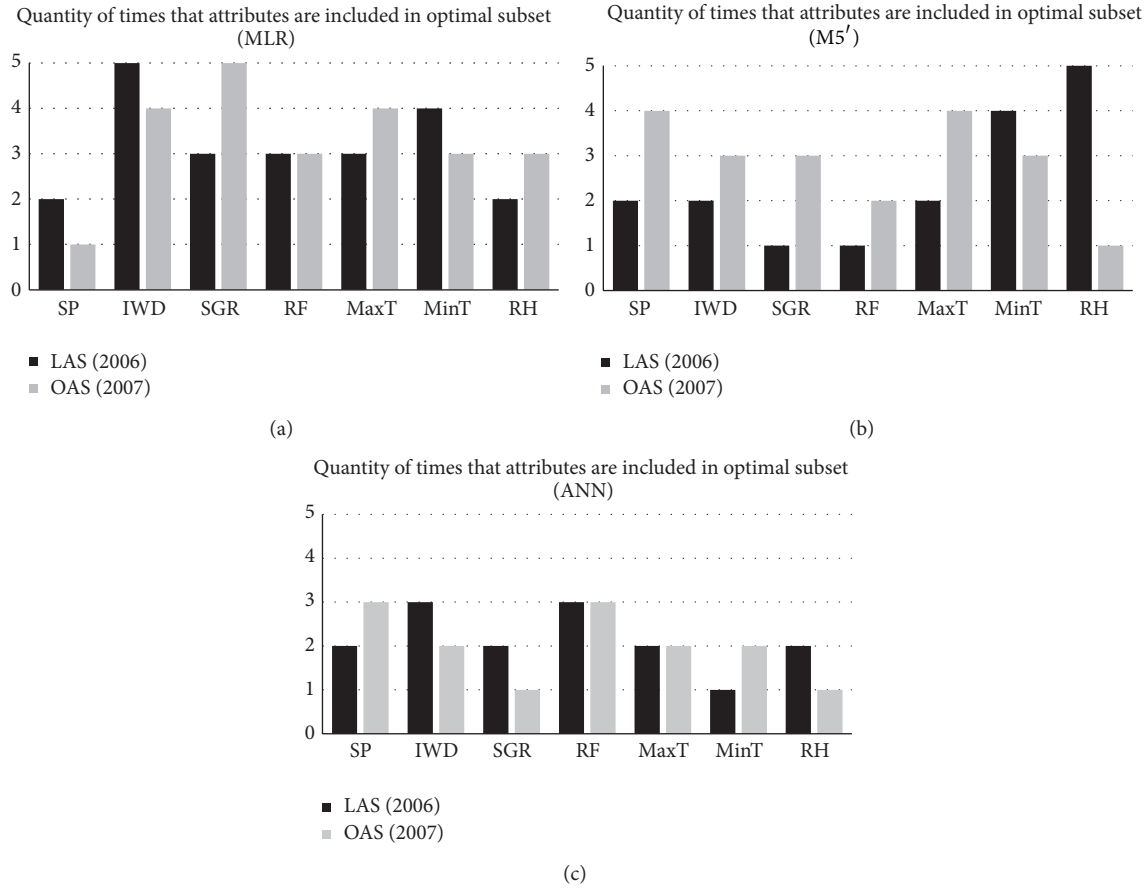


Figure 2: Quantity of occurrences of each attribute on the LAS and OAS for each technique (only crop datasets with 2007 testing data).

data (CBP02, CBA03, CBM04, PA06, and PA07). Figures 2(a) and 2(c) show that OAS and LAS are similar for MLR and ANN techniques. On the other hand, LAS and OAS obtained with M5′ technique are completely different in almost all cases (Figure 2(b)). As a result, ANN is the most consistent technique since its best attribute subsets scarcely varied from year to year (Figure 2(c)).

We applied the R measure to the attributes in LAS that intersects the OAS and the RRSE. This allows us to distinguish the error caused by including the relevant attributes in the model and the errors due to the regression technique predictive ability. This measure is shown in column two of Table 11. We also applied the R metric to the attributes excluded from LAS and those selected in OAS with the

Table 9: RMAE (%) measures using the LAS on testing dataset.

Crop	RMAE (%)			
	SLR	MLR	M5′	ANN
PJ01	40.61	**15.46**	**10.00**	**12.46**
CBP02	14.10	**5.16**	**8.41**	**6.41**
CBA03	10.11	14.99	12.35	**6.08**
CBM04	11.31	8.65	**1.72**	**1.72**
CP05	**10.11**	**8.81**	**8.04**	**8.46**
PA06	27.17	26.98	**23.10**	26.29
PA07	21.01	**17.61**	18.35	**17.40**
TS08	14.75	**11.08**	**13.46**	24.27
Average (RRSE < 100)	10.11	11.13	10.79	8.75
Count (RRSE < 100)	1	5	6	6
Average (all)	18.65	13.59	11.93	12.89

Table 10: Distance from LAS to OAS results.

Crop	Distance from optimal			
	SLR	MLR	M5′	ANN
PJ01	184	30	26	35
CBP02	145	7	113	1
CBA03	43	111	69	5
CBM04	232	170	6	6
CP05	96	9	23	1
Average	140	65.4	47.4	9.6

Table 11: Correlation coefficient between the counts of attributes in LAS-OAS intersection and RRSE.

Regression technique	*R* Correlation between RRSE and LAS-OAS intersection	*R* Correlation between RRSE and attributes left out from OAS
SLR	−0.707	0.347
MLR	−0.542	0.150
M5′	−0.641	0.034
ANN	−0.068	0.340

RRSE. This *R* correlation is shown in the third column of Table 11. These measures are included for SLR, MLR, and M5′ techniques. From column two in this table we observe that variables included in LAS and OAS have a strong effect over the error metric for SLR and M5′, while the effect of these variables scarcely affects the ANN performance. The third column of Table 11 indicates that the attributes excluded from OAS and included in LAS have a very small effect in M5′ and MLR, while a moderate impact over ANN and SLR can be observed. As a consequence, a biggest error among all these techniques can be expected from SLR.

4. Conclusions

This paper presents a comparison among several methods (linear and nonlinear) for crop yield prediction. The comparison is made using the best attribute subset found in the training dataset for each method, which was detected using a complete algorithm and the percentage-split validation scheme. The algorithm uses the oldest samples in training datasets to build the models, leaving the most recent to search the optimal attribute subset. The best attribute subset performance is measured with testing datasets composed of unseen samples from the future. The comparison covered eight crop datasets. The most common data-driven techniques for crop yield prediction were evaluated: stepwise linear regression, multiple linear regression, regression trees, and neural networks. The experimentation shows that our attribute selection using a complete method substantially improves the performance of all the evaluated techniques. ANN and M5′ obtained the best prediction, and, between them, the former achieved the lower RRSE, the higher R correlation, and the lower RMAE value. With respect to the optimal attribute subset composition, MLR and ANN techniques show small differences between the best attribute subset in learning stage and the optimal attribute subset for the testing stage, while M5′ shows the largest differences. The optimal attribute composition was different for all the evaluated techniques, which reinforces the hypothesis that using the same attributes subset for all the techniques is unfair. Nevertheless, none of the techniques was able to obtain the optimum subset with the training data for all the eight crops. The best technique was ANN, which achieved three attribute subsets equal to the optimal, and the other two subsets were very close to it. Thus, an attribute subset that can be used permanently in all the years for all the crops is difficult to select.

Results obtained from machine-learning methods cannot be directly applied to a different set of crop databases, due their high data dependency. However, the procedure presented in this paper can be extended for a larger number of techniques and crop datasets. A future research focused on finding the best minimal subset of attributes which provide a good yield of predictions on other irrigation zones should be done.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] A. A. Raorane and R. V. Kulkarni, "Data mining: an effective tool for yield estimation in the agricultural sector," *International Journal of Emerging Trends of Technology in Computer Science*, vol. 1, no. 2, pp. 75–79, 2012.

[2] B. Marinkovic, J. Crnobarac, S. Brdar, B. Antic, G. Jacimovic, and V. Crnojevic, "Data mining approach for predictive modeling of agricultural yield data," in *Proceedings of the 1st International Workshop on Sensing Technologies in Agriculture, Forestry and Environment (BioSense '09)*, pp. 1–5, Novi Sad, Serbia, October 2009.

[3] S. T. Drummond, K. A. Sudduth, A. Joshi, S. J. Birrell, and N. R. Kitchen, "Statistical and neural methods for site-specific yield prediction," *Transactions of the American Society of Agricultural Engineers*, vol. 46, no. 1, pp. 5–14, 2003.

[4] A. Irmak, J. W. Jones, W. D. Batchelor, S. Irmak, K. J. Boote, and J. O. Paz, "Artificial neural network model as a data analysis tool in precision farming," *Transactions of the ASABE*, vol. 49, no. 6, pp. 2027–2037, 2006.

[5] J. Han and M. Kamber, *Data Mining: Concepts and Techniques*, Morgan Kaufmann, 2nd edition, 2006.

[6] A. Roel and R. E. Plant, "Factors underlying yield variability in two California rice fields," *Agronomy Journal*, vol. 96, no. 5, pp. 1481–1494, 2004.

[7] J. G. Fortin, F. Anctil, L.-É. Parent, and M. A. Bolinder, "Site-specific early season potato yield forecast by neural network in Eastern Canada," *Precision Agriculture*, vol. 12, no. 6, pp. 905–923, 2011.

[8] G. Ruß and R. Kruse, "Feature selection for wheat yield prediction," in *Research and Development in Intelligent Systems XXVI*, M. Bramer, R. Ellis, and M. Petridis, Eds., pp. 465–478, Springer, London, UK, 2010.

[9] M. Karagiannopoulos, D. Anyfantis, S. Kotsiantis, and P. Pintelas, "A feature selection for regression problems," in *Proceedings of the 8th Hellenic European Conference on Computer Mathematics and Its Applications (HERCMA '07)*, Athens, Greece, September 2007.

[10] A. Arauzo-Azofra, J. M. Benitez, and J. L. Castro, "Consistency measures for feature selection," *Journal of Intelligent Information Systems*, vol. 30, no. 3, pp. 273–292, 2008.

[11] W. Ojeda-Bustamante, J. M. González-Camacho, E. Sifuentes-Ibarra, E. Isidro, and L. Rendón-Pimentel, "Using spatial information systems to improve water management in Mexico," *Agricultural Water Management*, vol. 89, no. 1-2, pp. 81–88, 2007.

[12] B. Safa, A. Khalili, M. Teshnehlab, and A. Liaghat, "Artificial neural networks application to predict wheat yield using climatic data," in *Proceedings of 20th International Conference on IIPS*, pp. 1–39, Iranian Meteorological Organization, January 2004.

[13] J. Frausto-Solis, A. Gonzalez-Sanchez, and M. Larre, "A new method for optimal cropping pattern," in *Proceedings of the 8th Mexican International Conference on Artificial Intelligence (MICAI '09)*, vol. 5845 of *Lecture Notes in Computer Science*, pp. 566–577, Springer, Guanajuato, México, 2009.

[14] L. Wasserman, *All of Statistics. A Concise Course in Statistical Inference*, Springer, New York, NY, USA, 2004.

[15] IBM Corp., *IBM SPSS Statistics for Windows, Version 20.0*, IBM Corp., Armonk, NY, 2011.

[16] R. Mundry and C. L. Nunn, "Stepwise model fitting and statistical inference: turning noise into signal pollution," *The American Naturalist*, vol. 173, no. 1, pp. 119–123, 2009.

[17] I. H. Witten and E. Frank, *Data Mining, Practical Machine Learning Tools and Techniques*, Morgan Kaufmann, 2nd edition, 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 672630, 6 pages
http://dx.doi.org/10.1155/2014/672630

Research Article

Kruskal-Wallis-Based Computationally Efficient Feature Selection for Face Recognition

Sajid Ali Khan,[1,2] Ayyaz Hussain,[3] Abdul Basit,[1] and Sheeraz Akram[1]

[1] *Department of Software Engineering, Foundation University, Rawalpindi 46000, Pakistan*
[2] *Department of Computer Science, Shaheed Zulfikar Ali Bhutto Institute of Science and Technology Islamabad, Islamabad 44000, Pakistan*
[3] *Department of Computer Science and Software Engineering, International Islamic University, Islamabad 44000, Pakistan*

Correspondence should be addressed to Sajid Ali Khan; sajidalibn@gmail.com

Received 5 December 2013; Accepted 10 February 2014; Published 21 May 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

Face recognition in today's technological world, and face recognition applications attain much more importance. Most of the existing work used frontal face images to classify face image. However these techniques fail when applied on real world face images. The proposed technique effectively extracts the prominent facial features. Most of the features are redundant and do not contribute to representing face. In order to eliminate those redundant features, computationally efficient algorithm is used to select the more discriminative face features. Extracted features are then passed to classification step. In the classification step, different classifiers are ensemble to enhance the recognition accuracy rate as single classifier is unable to achieve the high accuracy. Experiments are performed on standard face database images and results are compared with existing techniques.

1. Introduction

Face recognition is becoming more acceptable in the domain of computer vision and pattern recognition. The authentication systems based on the traditional ID card and password are nowadays replaced by the techniques which are more preferable in order to handle the security issues. The authentication systems based on biometrics are one of the substitutes which are independent of the user's memory and not subjected to loss. Among those systems, face recognition gains special attention because of the security it provides and because it is independent of the high accuracy equipment unlike iris and recognition based on the fingerprints.

Feature selection in pattern recognition is specifying the subset of significant features to decrease the data dimensions and at the same time it provides the set of selective features. Image is represented by set of features in methods used for feature extraction and each feature plays a vital role in the process of recognition. The feature selection algorithm drops all the unrelated features with the highly acceptable precision rate as compared to some other pattern classification problem in which higher precision rate cannot be obtained by greater number of feature sets [1].

The feature selected by the classifiers plays a vital role in producing the best features that are vigorous to the inconsistent environment, for example, change in expressions and other barriers. Local (texture-based) and global (holistic) approaches are the two approaches used for face recognition [2]. Local approaches characterized the face in the form of geometric measurements which matches the unfamiliar face with the closest face from database. Geometric measurements contain angles and the distance of different facial points, for example, mouth position, nose length, and eyes. Global features are extracted by the use of algebraic methods like PCA (principle component analysis) and ICA (independent component analysis) [3]. PCA shows a quick response to light and variation as it serves inner and outer classes fairly. In face recognition, LDA (linear discriminate analysis) usually performs better than PCA but separable creation is not precise in classification. Good recognition rates can be produced by transformation techniques like DCT (discrete cosine transform) and DWT (discrete wavelet transform) [4].

To analyze unstable signals, wavelet analysis is used which is fast and also provides good frequency domain quality. Image of the face is divided first into subregions [5]. Afterwards facial features are extracted using weber local descriptor. The orientation component is done by Sobel descriptor. The subregions of an image are recognized by the use of the nearest neighborhood method. Integration in decision level results in final recognition. Rates of recognition are high but costly. In [6] two famous techniques are discussed in order to extract face features. Significant features are selected by particle swarm optimization. It also decreases the dimensions of data and Euclidean distance classifier is trained and tested by using optimized features. But the problem with PSO is that it is an expensive process.

A lot of methods, for example, greedy algorithm [7], branch and bound algorithm [8], mutual information [9], and tabu search [10], have been used on the testing and training data for feature selection. Methods based on genetic algorithm [11] and ant colony optimization have gained a lot of attention [12] which are population-based optimization algorithms. These methods try to provide a good solution by obtaining knowledge from the older iteration. Feature selection algorithm usually uses heuristic in order to avoid confusion.

The main aim of the paper is to introduce a face recognition system that is computationally less expensive, and only the information related to facial features is used. For this DWT- and WLD-based techniques are used to extract face features. The significant features with high information are utilized by Kruskal-Wallis algorithm and the results are compared with the famous techniques like PSO and GA. An ensemble of three classifiers is used to improve the precision rate of recognition.

2. The proposed Methodology

The proposed techniques steps are represented by Figure 1. In the first step, facial features are extracted using discrete wavelet transform. To reduce the data dimensions, computationally efficient technique (Kruskal-Wallis) is applied to select the most prominent face features. In the last step, different well-known classifiers are trained and tested using those extracted features to recognize the face image.

2.1. Feature Extraction. Two techniques are used to extract the face features. Details of these techniques are provided below.

2.1.1. Wavelet-Based Face Feature Extraction. DWT is one of the wavelet transforms which was founded in 1976 when discrete time signals were decomposed by Polikar [13]. It is substitute for cosine transform in which the functions of sine cosine are added and the varying value of time and frequency is returned by wavelet.

Decomposition by columns and rows is the second famous method which uses low pass filter in its iterations. The input image is divided into subcomponents after passing through low and high pass filter. These subbands include the details about vertical and horizontal properties of an image. The low pass filter gives low frequency subbands which includes detailed information and the process is repeated on the subband in the second level. The high pass filter gives the high frequency subbands, the vertical subbands, and the diagonal coefficients. Figure 2 shows DWT standard method.

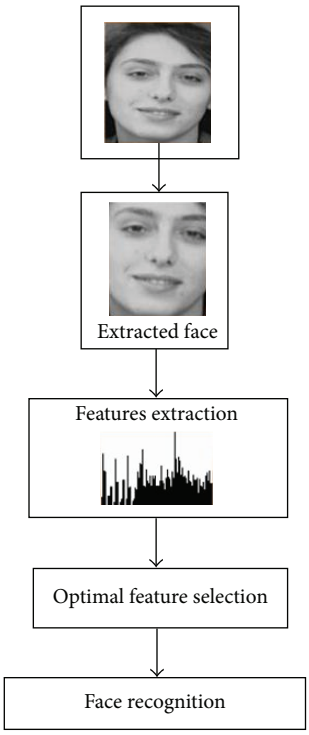


Figure 1: The proposed system architecture.

Figure 3 shows the detailed process of DWT. Four subbands of an image (LL, HL, LH, and HH) are acquired by applying DWT. LL shows the approximate coefficients. Detail coefficients are represented by HL, LH, and HH. There are different types of wavelet transform, for example, Symmlet, Haar, Daubechies, and Coiflet with various numbers of vanishing moments (features of wavelets). These are scaling functions which show complex signals precisely [14].

Daubechies wavelet is used in the proposed method to extract DWT features. It is the orthogonal wavelets which show the discrete wavelet transform with greater vanishing moments. Another scaling function is named as father wavelet which produces an orthogonal multiresolution analysis (MRA) used in the method [15]. The scaling function makes sure that the whole spectrum is covered and filters the lowest level of transform. The MRA is the sequence of nested subspaces. Vector space is the first element of the MRA and for every vector space there exists another vector space with higher resolution until a final image is obtained.

2.2. Feature Selection. The performance of classification system can be degraded by using all the features of input data as it increases the complexity. Picking up the optimized features is very important as some features play more important role in recognition. There are a lot of methods that are developed

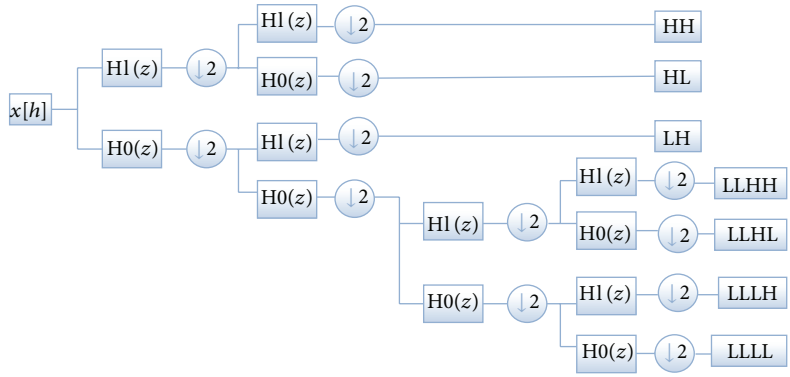


Figure 2: Standard DWT [13].

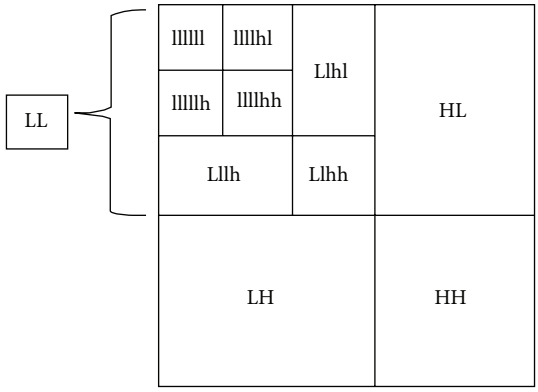


Figure 3: Standard 2D DWT decomposition.

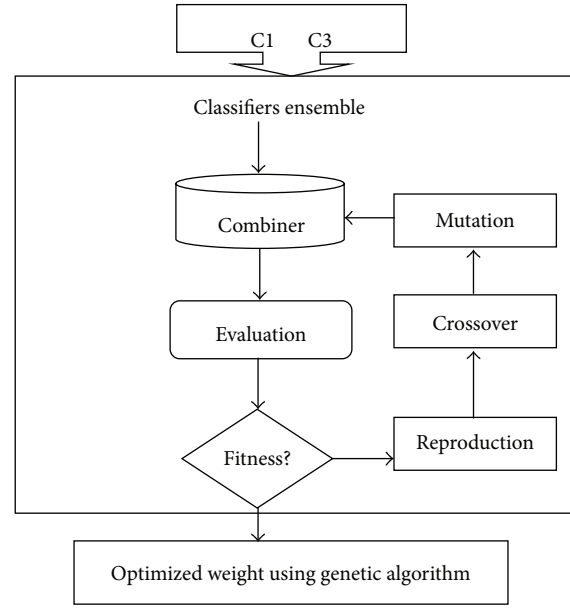


Figure 4: Classifiers ensemble and optimization process flow diagram.

and have been used for features but most of them are computationally expensive and complex in nature. Kruskal-Wallis method [16] is used in the proposed method in order to select significant features which is computationally less expensive and very simple in use. Kruskal-Wallis method tests if two or more classes have equal median and gives the value of P. Features with discriminative information are selected. If the value of P is close to "0" it means that the feature contains discriminative information; otherwise it will not be selected. DWT features are processed using the Kruskal-Wallis technique. Features which result in a value of P less than a threshold are selected to be used in the next recognition step.

2.3. Classification. Single classifier is unable to achieve high accuracy rate so two well-known classifiers are trained and tested. Figure 4 represents the classifiers ensemble and optimization process using Genetic algorithm. Description of the classifiers is given below.

2.3.1. K-Nearest Neighbour Classifier. K-nearest neighbour classifier classifies the sample data by allocating it to that class label which more commonly represents its nearest neighbours value, that is, k. If the tie situation occurs between test samples then decision is based on distance calculation. The sample will be assigned to that class which has smaller distance from the test sample. KNN performance is dependent on the optimal value of K and the distance. Different methods have been used by researchers to calculate the distance, for example, Euclidean, Minkowsky, and cambra. The Euclidean distance method is more common and famous [17]. The equation used to calculate the distance between the two points, X and Y, is as follows:

$$d(X,Y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}, \quad (1)$$

where $X = \{x_1, x_2, x_3, \ldots, x_n\}$ and $Y = \{y_1, y_2, y_3, \ldots, y_n\}$.

Learning speed of KNN classifier is fast but its classification accuracy is relatively poor. KNN has another beauty that it is the smallest classifier compared to all other machine learning algorithms [17].

Figure 5: Extended Yale face database sample images.

2.3.2. Support Vector Machine (SVM). SVM assigns the testing samples to the class whose distance is maximum to the nearest point in the training set. SVM draws a hyperplane if the data is linearly separable. Distinct kernel is utilized to check the accuracy of SVM [18].

2.3.3. Optimization through GA. The classifier behaviour changes every time, and it is possible that the accuracy of a better classifier may degrade and vice versa. Some mechanism needs to be developed to keep the classifier accuracy rate at an acceptable rate. In our approach, the weights of the classifiers are optimized using genetic algorithm to achieve this goal. GA is used in many distinct optimization concepts because it does not require any particular knowledge about problem domain. First, the GA uses problem space to randomly pick different solutions, that is, N. In each iteration, selection and reproduction operator are used to optimize these problems.

We normalize the weights between [0-1]. The chromosome "m" length is matched to the number of classifiers used, that is, l. The results produced by the classifiers are used as initial population and then error rate is generated after applying the fitness function for evaluation. Elitism policy is used in the experiments to bring the "e" number of the best chromosome in the new generation. In the next step, new weights are found using mutation "mu" and crossover operation "cr."

The condition that whether the new weight should be given to the classifiers or not is the quality of the chromosomes. The GA will terminate when the generation reached to Gmax or population convergence to fulfill solution [19].

3. Experimental Results and Discussion

We have performed experiments on extended Yale face database B [20]. Yale database contains 16,128 images of 28 different individuals in GIF format and of size 100 × 100. These variations in expressions include sleepy, sad, wink, and happy. Light conditions are also changed which include normal light, centre light, and right light. Figure 5 shows the Yale face database sample images. Leave-one strategy is used in all experiments and performance of the system was evaluated and compared after performing different steps.

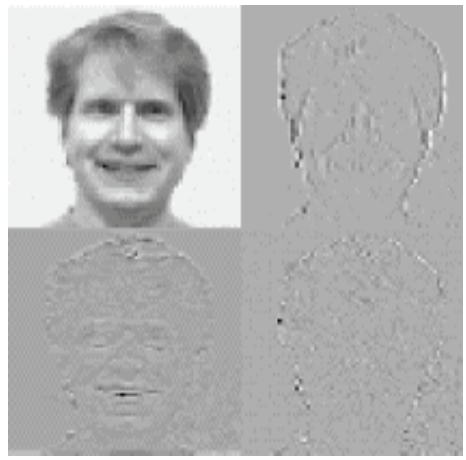


Figure 6: 1-level DWT decomposition.

Table 1: Face recognition accuracy rate on DB wavelet.

Classifier/features	11 × 14	9 × 8	6 × 8
KNN	0.884	0.914	0.8332
SVM	0.8571	0.899	0.8134

First, the DWT-based coefficients are extracted from the face. The image size is reduced to 1/4 after implementation of 2D discrete wavelet transform. The image is decomposed up to two levels and different feature vectors are formed. Figure 2 depicts the 2-level DWT decomposition strategy.

In Figure 6, the top right corner image represents the image having maximum discriminative information and more stable as compared to other subbands. We have used this subband for the next level decomposition. Then after the second level decomposition using Daubechies wavelet, the most important features are extracted by applying principal component analysis (PCA). The features having higher eigenvectors are selected. The feature vector dimensions are 11 × 14, 9 × 8, and 6 × 8, which are correlated to the levels 0, 1, 2, and 3 in decomposition of wavelet.

Table 1 presents the recognition accuracy rate of DWT-based extracted features using Daubechies family.

Table 2: Face recognition accuracy on Haar Wavelets.

Classifier/features	11 × 14	9 × 8	6 × 8
KNN	0.866	0.914	0.827
SVM	0.871	0.9233	0.844

Table 3: Proposed technique comparison with existing techniques.

Method	Recognition rate
Proposed technique	98%
DWT + PSO [6]	96%
Local ternary pattern (LTP) [21]	91%
K2DSPCA [22]	96%
Harmony search algorithm [23]	94%
SIFT [24]	91%

During the experiments, different classifiers are trained and tested using the extracted features. Performance of KNN and SVM is better than other classifiers in case of classifying face images. In case of KNN, accuracy rate of 88% is achieved using feature vector of size 11 × 14. Accuracy rate increases when the feature vector size decreases (i.e., 9 × 8). The best accuracy of 91% is obtained using KNN in case of 9 × 8 feature vector size.

Table 2 presents the results after using Haar family of DWT. We have observed that in our case the Daubechies family of wavelet performed slightly better than Haar wavelet. It has been observed that SVM is outperformed as compared to KNN using feature set of different sizes. Average accuracy rate of 92% is obtained using SVM classifier.

After performing different experiments, we noted that some of the face features are redundant and do not contribute to recognition process. Kruskal-Wallis feature selection technique is used to select discriminative face features. In Kruskal-Wallis algorithm the value of P is changed to eliminate the redundant features and find the optimum threshold. The variation [0.01~0.19] and [2.0 of 4.0] of P was followed for the experiments. After implementation of Kruskal-Wallis algorithm feature vector of size 40 is obtained. These features contained more discriminative information about face and produced high recognition rate.

Single classifier is unable to achieve the highest accuracy rate. In the next step, SVM and KNN classifier is ensemble and optimized using GA to improve the recognition rate.

Figure 7 represents the recognition rate of single classifier and accuracy after the classifiers are ensemble and optimized by GA. It has been observed that data of high dimension and irrelevant feature decrease the accuracy rate and also are time consuming. After optimization process the average accuracy rate of 98% is achieved.

In Table 3, the proposed technique is compared with other existing techniques in terms of recognition rate accuracy.

4. Conclusions and Future Work

In this work, DWT is analyzed to extract the prominent face features. The search space is reduced greatly by using Kruskal-Wallis algorithm. Kruskal-Wallis algorithm selects the more discriminative face features. It is concluded that single classifier is unable to achieve the high accuracy rate. In order to improve the accuracy rate, two well-known classifiers are ensemble and then optimized using GA. After optimization process the accuracy rate increases. Kruskal-Wallis algorithm searching strategy is simple and less time consuming as compared to other GA and particle swarm optimizations.

In the future the proposed technique will be modified and will be used for 3D face images.

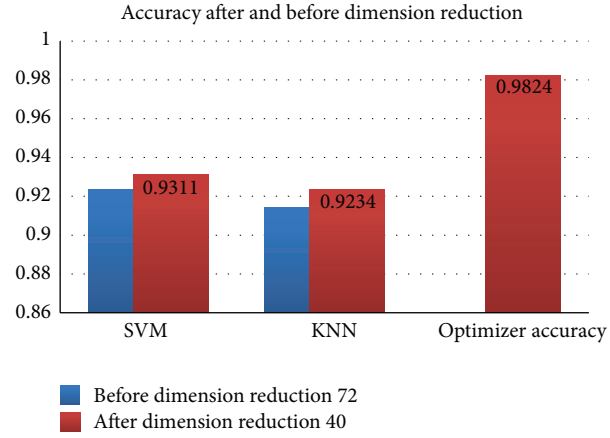


Figure 7: Different classifiers accuracy rate before and after data dimension reduction.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] T. Chung, L. Chuang, J. Chang, and C. Yang, "Feature selection using PSO-SVM," *International Journal of Computer Science*, vol. 33, pp. 31–37, 2007.

[2] Q. Villegas and J. Climent, "Holistic face recognition using multivariate approximation, genetic algorithms and adaBoost classifier," *Proceedings of World Academy of Science: Engineering and Technolog*, vol. 44, pp. 802–806, 2008.

[3] H. Rady, "Face recognition using principle component analysis with different distance classifiers, genetic algorithms and adaBoost classifier," *International Journal of Computer Science and Network Security*, vol. 11, pp. 134–144, 2011.

[4] A. Samra, S. Allah, and R. M. Ibrahim, "Face recognition using wavelet transform, fast fourier transform and discrete cosine transform," in *Proceedings of the 46th IEEE International Midwest Symposium Circuits and Systems (MWSCAS '03)*, pp. 272–275, 2003.

[5] D. Gong and S. Li, "Face recognition using the weber local descriptor," in *Proceedings of the 1st Asian Conference on Pattern Recognition*, pp. 589–592.

[6] R. Ramadan and A. Kader, "Face recognition using particle swarm optimization-based selected features," *International*

Journal of Signal Processing, Image Processing and Pattern Recognition, vol. 2, pp. 51–66, 2009.

[7] E. Kokiopoulou and P. Frossard, "Classification-specific feature sampling for face recognition," in *Proceedings of the IEEE 8th Workshop on Multimedia Signal Processing (MMSP '06)*, pp. 20–23, October 2006.

[8] P. M. Narendra and K. Fukunaga, "A branch and bound algorithm for feature subset selection," *IEEE Transactions on Computers*, vol. 6, no. 9, pp. 917–922, 1977.

[9] R. Battiti, "Using mutual information for selecting features in supervised neural net learning," *IEEE Transactions on Neural Networks*, vol. 5, no. 4, pp. 537–550, 1994.

[10] H. Zhang and G. Sun, "Feature selection using tabu search method," *Pattern Recognition Letter*, vol. 35, no. 3, pp. 701–711, 2002.

[11] X. Fan and B. Verma, "Face recognition: a new feature selection and classification technique," in *Proceedings of the 7th Asia-Pacific Conference on Complex Systems*, 2004.

[12] H. R. Kanan, K. Faez, and M. Hosseinzadeh, "Face recognition system using ant colony optimization-based selected features," in *Proceedings of the IEEE Symposium on Computational Intelligence in Security and Defense Applications (CISDA '07)*, pp. 57–62, April 2007.

[13] H. Polikar, "The wavelet tutorial," 1999.

[14] G. Graps, "An introduction to wavelets," *IEEE computational Science & Engineering*, vol. 2, no. 2, pp. 50–61, 1995.

[15] Daubechies wavelet, 1999.

[16] Y. Saeys, I. Inza, and P. Larrañaga, "WLD: review of feature selection techniques in bioinformatics," *Bioinformatics*, vol. 23, no. 19, pp. 2507–2517, 2007.

[17] S. B. Kotsiantis, "Supervised machine learning: a review of classification techniques," *Informatica*, vol. 31, no. 3, pp. 249–268, 2007.

[18] L. I. Kuncheva, *Combining Pattern Classifiers: Methods and Algorithms*, 2004.

[19] B. Gabrys and D. Ruta, "Genetic algorithms in classifier fusion," *Applied Soft Computing Journal*, vol. 6, no. 4, pp. 337–347, 2006.

[20] A. S. Georghiades, P. N. Belhumeur, and D. J. Kriegman, "From few to many: illumination cone models for face recognition under variable lighting and pose," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 23, no. 6, pp. 643–660, 2001.

[21] X. Tan and B. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE Transactions on Image Processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[22] S. Kumar and S. Banerji, "Face recognition using K2DSPCA," in *Proceedings of the International Conference on Information and Network Technology*, pp. 84–88, 2011.

[23] R. Sawalha and I. Doush, "Face recognition using harmony search-based selected features," *International Journal of Hybrid Information Technology*, vol. 5, pp. 1–16, 2012.

[24] C. Geng and X. Jiang, "Face recognition using SIFT features," in *Proceedings of the IEEE International Conference on Image Processing (ICIP '09)*, pp. 3313–3316, November 2009.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 810368, 15 pages
http://dx.doi.org/10.1155/2014/810368

Review Article

Intelligent Screening Systems for Cervical Cancer

Yessi Jusman, Siew Cheok Ng, and Noor Azuan Abu Osman

Department of Biomedical Engineering, Faculty of Engineering Building, University of Malaya, 50603 Kuala Lumpur, Malaysia

Correspondence should be addressed to Siew Cheok Ng; siewcng@um.edu.my and Noor Azuan Abu Osman; azuan@um.edu.my

Received 24 December 2013; Accepted 11 February 2014; Published 11 May 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

Advent of medical image digitalization leads to image processing and computer-aided diagnosis systems in numerous clinical applications. These technologies could be used to automatically diagnose patient or serve as second opinion to pathologists. This paper briefly reviews cervical screening techniques, advantages, and disadvantages. The digital data of the screening techniques are used as data for the computer screening system as replaced in the expert analysis. Four stages of the computer system are enhancement, features extraction, feature selection, and classification reviewed in detail. The computer system based on cytology data and electromagnetic spectra data achieved better accuracy than other data.

1. Introduction

Cervical cancer is a leading cause of mortality and morbidity, which comprises approximately 12% of all cancers in women worldwide according to World Health Organization (WHO). In fact, the annual global statistics of WHO estimated 470 600 new cases and 233 400 deaths from cervical cancer around the year 2000. As reported in National Cervical Cancer Coalition (NCCC) in 2010, cervical cancer is a cancer of the cervix which is commonly caused by a virus named Human Papillomavirus (HPV) [1]. The virus can damage cells in the cervix, namely, squamous cells and glandular cells that may develop into squamous cell carcinoma (cancer of the squamous cells) and adenocarcinoma (cancer of the glandular cells), respectively. Squamous cell carcinoma can be thought of as similar to skin cancer because it begins on the surface of the ectocervix. Adenocarcinoma begins further inside the uterus, in the mucus-producing gland cells of the endocervix [2].

Cervical cancer develops from normal to precancerous cells (dysplasia) over a period of two to three decades [3]. Even though the dysplasia cells look like cancer cells, they are not malignant cells. These cells are known as cervical intraepithelial neoplasia (CIN) which is usually of low grade, and they only affect the surface of the cervical tissue. The majority will regress back to normal spontaneously. Over time, a small proportion will continue to develop into cancer.

Based on WHO system, the level of CIN growth can be divided into grades 1, 2, and 3. It should be noted that at least two-thirds of the CIN 1 lesions, half of the CIN 2 lesions, and one-third of the CIN 3 lesions will regress back to normal [3]. The median ages of patients with these different precursor grades are 25, 29, and 34 years, respectively. Ultimately, a small proportion will develop into infiltrating cancer, usually from the age of 45 years onwards.

In 1994, the Bethesda system was introduced to simplify the WHO system. This system divided all cervical epithelial precursor lesions into two groups: the Low-grade Squamous Intraepithelial Lesion (LSIL) and High-grade Squamous Intraepithelial Lesion (HSIL). The LSIL corresponds to CIN1, while the HSIL includes CIN2 and CIN3 [4].

Since a period of two to three decades is needed for cervical cancer to reach an invasive state, the incidence and mortality related to this disease can be significantly reduced through early detection and proper treatment. Realizing this fact, a variety of screening tests have therefore been developed in attempting to be implemented as early cervical precancerous screening tools.

2. Methodology

This paper reviews 103 journal papers. The papers are obtained electronically through 2 major scientific databases:

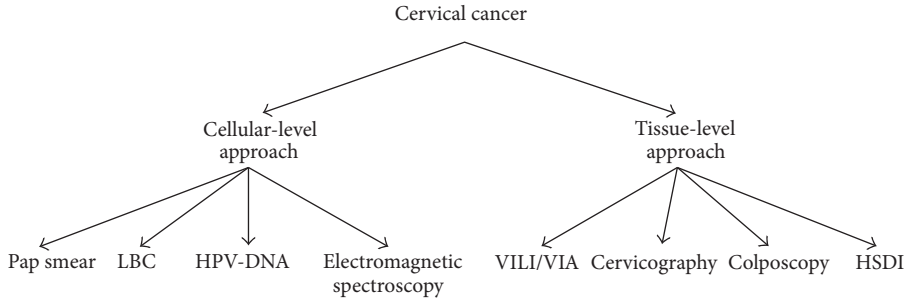


Figure 1: Taxonomy of cervical cancer screening.

Table 1: Comparison of the ability of the manual cervical screening methods.

Highlighted features	Cellular level				Tissue level			
	Pap smear	LBC	HPV-DNA	EMS	VILI/VIA	Cervicography	Colposcopy	HSDI
Low cost	V*	V*	V	V*	V	V	V*	V*
Short time	X	X	X	X	V	V	V	V
Not Subjective	X	X	V	V	X	X	X	X
Possible in real time	V	V	X	V	X	V	V	V

Google Scholar (http://scholar.google.com.my/) and Scopus (http://www.scopus.com/home.url). In the databases, the IEEE and Science Direct databases will be included already. Since there are various aspects being reviewed here, four sets of keywords have been used. The first set contains Cervical Cancer, Feature Extraction, and Intelligent System, which give an overview of an intelligent system for cervical cancer detection. The second set contains Cervical Cancer, Image Processing, and Intelligent System. The third set is made up of Cervical Cancer, Image Processing, and Classification. The final set contains Cervical Cancer, Features Extraction, and Image Processing.

In order to ensure a quality review, the academic papers reviewed here are limited to peer reviewed journal papers. Recent conference papers published in the year 2010 onwards are also considered as the work is up to date and the journal related to this work has yet to be published. However, certain conference papers that showed excellent results or used methods that are currently unpopular are also included to give a more complete perspective of the work done in this field.

3. Screening for Cervical Carcinoma

Screening programs for cervical cancer have been implemented in developing countries for decades and have shown to be effective in reducing the overall mortality from this disease. There are two main diagnostic screening approaches for cervical cancer as presented in Figure 1:

(1) diagnostic screening approach based on cellular level (i.e., Pap smear, liquid based cytology (LBC), HPV-DNA testing, and electromagnetic spectroscopies);

(2) diagnostic screening approach based on the tissue level (i.e., visual inspection after applying Lugol's iodine (VILI) or acetic acid (VIA), cervicography, colposcopy, and hyperspectral diagnostic imaging (HSDI)).

For diagnostic screening based on cellular-level, the specimen collections are required before it is analyzed for the expert analysis results. In contrast, specimen collection is not required for diagnostic screening based on tissue-level. The expert analysis is required for cervix images visually after applying certain liquid into the cervix surface. Detail of standard procedure, advantages, and disadvantages for Pap smear, LBC, HPV-DNA, VILI/VIA, cervicography, and colposcopy techniques can be found in [5].

On the other hand, current technologies have investigated the cervical cell from the specimen under the spectroscopy equipment inducing an electromagnetic light. There are several techniques utilized for cervical cancer detection:

(1) image results: fluorescent in situ hybridization (FISH) [6–11];

(2) spectra results: Raman spectroscopy [12, 13], fluorescence spectroscopy [14, 15], and Fourier transform infrared (FTIR) spectroscopy [16–24].

On the other hand, there is an alternative technique based on tissue level known as hyperspectral diagnostic imaging (HSDI). The surface of the cervix is scanned with ultraviolet and white light for detecting lesions [25–27]. The scanning is achieved one line at a time, with the scan time varying from 12 to 24 seconds. By taking a series of scan lines, a hyperspectral data cube is obtained. This hyperspectral data cube contains spatial information (pixels) in two dimensions and spectral information (bands) in the third dimension [27]. This technique produces a 3D cervix image that is easier to interpret.

Based on the references, the techniques have several features required for considerations as summarized in Table 1. Each of the technique has advantages and disadvantages

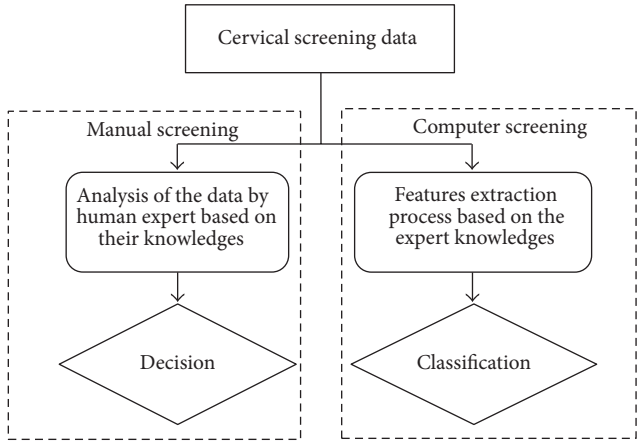


Figure 2: Comparison of analysis screening system by human expert and machine.

individually. Almost all of the techniques have on average nonexpensive or low cost features [5, 28, 29]. However, the EMS machines as well as microscope (for Pap smear and/or ThinPrep) and high resolution camera (for colposcopy and/or HSDI) are quite expensive to be bought for the beginning proses as screening technique but it is cost effective in the long run as no analysis from pathologist is required.

For the cellular-level techniques, the specimen collections require certain duration time and the results cannot be obtained spontaneously after specimen collection process due to need of the next process for the expert reading (i.e., image, spectrum, genetic material, etc.). As for the tissue-level, analysis of the experts could be obtained after reading the images captured by the camera.

Based on Table 1, the HPV-DNA is not subjective due to the genetic material for chemistry analysis on the cell. Similarly, the EMS techniques are also not subjective. They have quantitative results used for analysis. However, the HPV-DNA and the VILI/VIA techniques are not possible to interface in real time so they cannot be developed into an intelligent system. Therefore, intelligent systems for cervical precancerous is limited to the six possible techniques in real time as presented in Table 1.

4. Intelligent System Approach to Cervical Cancer

The cervical screening methods mentioned in Section 3 are highly dependent on the skill of the experts. However, their judgment may be subjective and often leads to considerable variability [5]. Aside from that, the limited number of experts and the large number of patients resulted in a long queue for the screening process. To overcome these problems, computational tools have been developed for automated cancer diagnosis as drawn in Figure 2. The automated cancer diagnosis facilitates objective judgment complementary to expert's decision.

Figure 2 shows the principle comparison of the computer screening technique and the human expert. The feature's extraction and classification by the computer replace the analysis and decision of human experts. Currently, the requirement for analysis based on computer screening increases. A number of researches were carried out specifically with the attempts to automate the classification [30, 31]. The results of several research to indicate that computer-imaging-assisted screening significantly increases the detection of cervical abnormalities compared to the manual screening [32, 33]. Consequently, automated screening devices would be a tremendous improvement for reducing the likelihood of human errors.

A typical computer screening system involves four stages, namely, data enhancement, features extraction, features selection, and classification as shown in Figure 3. Aside from visual inspection after applying Lugol's iodine (VILI) or acetic acid (VIA) and HPV-DNA Testing, the data from the other screening techniques can be digitalized and fed into the intelligent computer screening system. These data can be categorized as images or spectra.

In the enhancement stage, the image or spectra will be processed in order to eliminate the noise to increase the signal to noise ratio. For images, this stage also involves determination of the region of interest to be segmented out for further processing. For the images, features are extracted either at the cellular or at the tissue-level. Basically, the morphology, texture, shape, and/or intensity of the cell/tissue image are extracted as features. For spectra, the features are height of intensity, shift of wave number, and corrected area and area under peaks of the spectra.

The main purpose of feature's selection is to reduce the number of features used in classification while maintaining acceptable classification accuracy. Feature's selection includes methods such as sequential backward selection [34], sequential forward selection [35], sequential floating search method [36], discriminant analysis [37], and principal component analysis [17].

After features selection step, several classifiers can be employed to obtain classification performance based on the used features. Different classification results can be performed by the different features used [38]. The aim of

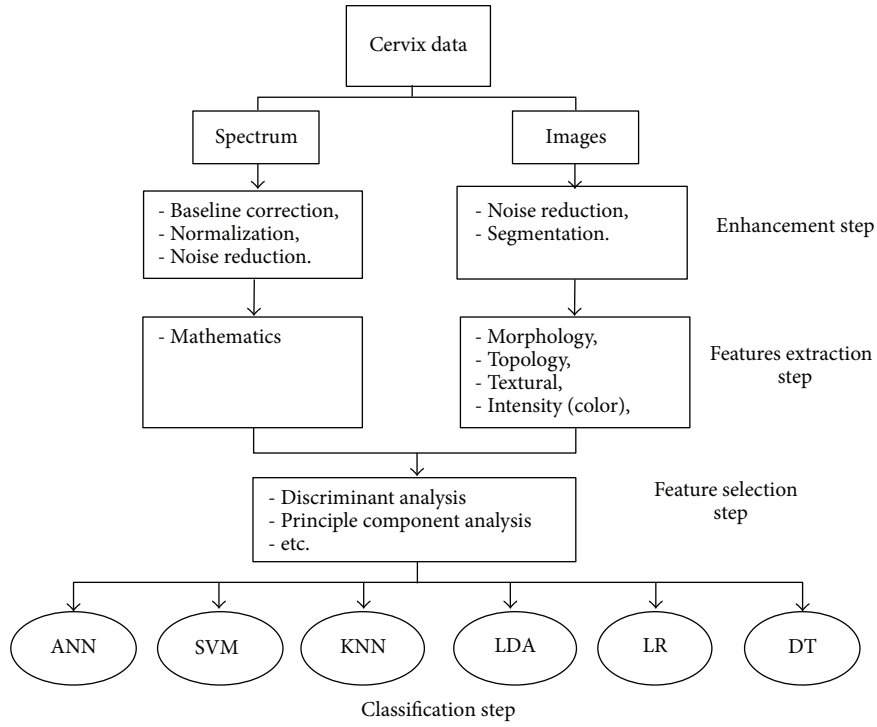


Figure 3: Intelligent cervical cancer classification systems.

Table 2: Information about cervical screening instruments.

Information	PAPNET	AutoPap 300	FocalPoint	TIS
Input data	Pap smear only	Pap smear only	Pap smear and ThinPrep	ThinPrep only
Characteristic	Semiautomatic system	Automatic system	Automatic system	Automatic system
USFDAapproval	Secondary screening	Primary screening	Primary screening	Primary screening

diagnosis step is to distinguish benignity and malignancy or to classify different malignancy levels by making use of extracted features. This step uses statistical analysis of the features and machine learning algorithms to reach a decision. An overview of these four stages is given in Figure 3. In the following sections, we will study each of these steps in detail.

Nowadays, there are several instruments which have been used to screen for abnormal cervical cells such as semi-automated or interactive system (PAPNET) and automated systems (AutoPap 300, FocalPoint, and ThinPrep Imaging System (TIS)) [30, 33, 39–41]. These instruments have been approved by United States Food and Drug Administration (USFDA) for screening system. These instruments utilize algorithmic image analysis to extract morphological features. Most of these systems help the expert to perform better diagnosis by improving cervical cell images quality so that the morphological features can be seen easily. Table 2 summarizes the instruments to view their advantages and disadvantages.

In fact, to build the current intelligent cervical screening system, two types of raw data (i.e., digital images and spectra) as presented in Section 3 can be used for the purposes. To construct the intelligent system, data enhancement (optional), features extraction, and classification steps are applied to the raw data to obtain good screening results approach of the human expert knowledge in some areas of their expertise [42, 43]. Therefore, here we review some current features extraction techniques and classification of two types of cervical data.

4.1. Data Enhancement. As stated in earlier section, there are two types of cervical cancer data, which are spectrum and image as presented in Figure 4. The main aim of the enhancement stage is to reduce noise and for the image data to determine the area of interest as well. Due to a considerable amount of noise that arises from the staining process, it is usually necessary to reduce the noise prior to the segmentation process. In some studies, noise reduction and segmentation are carried out at the same time.

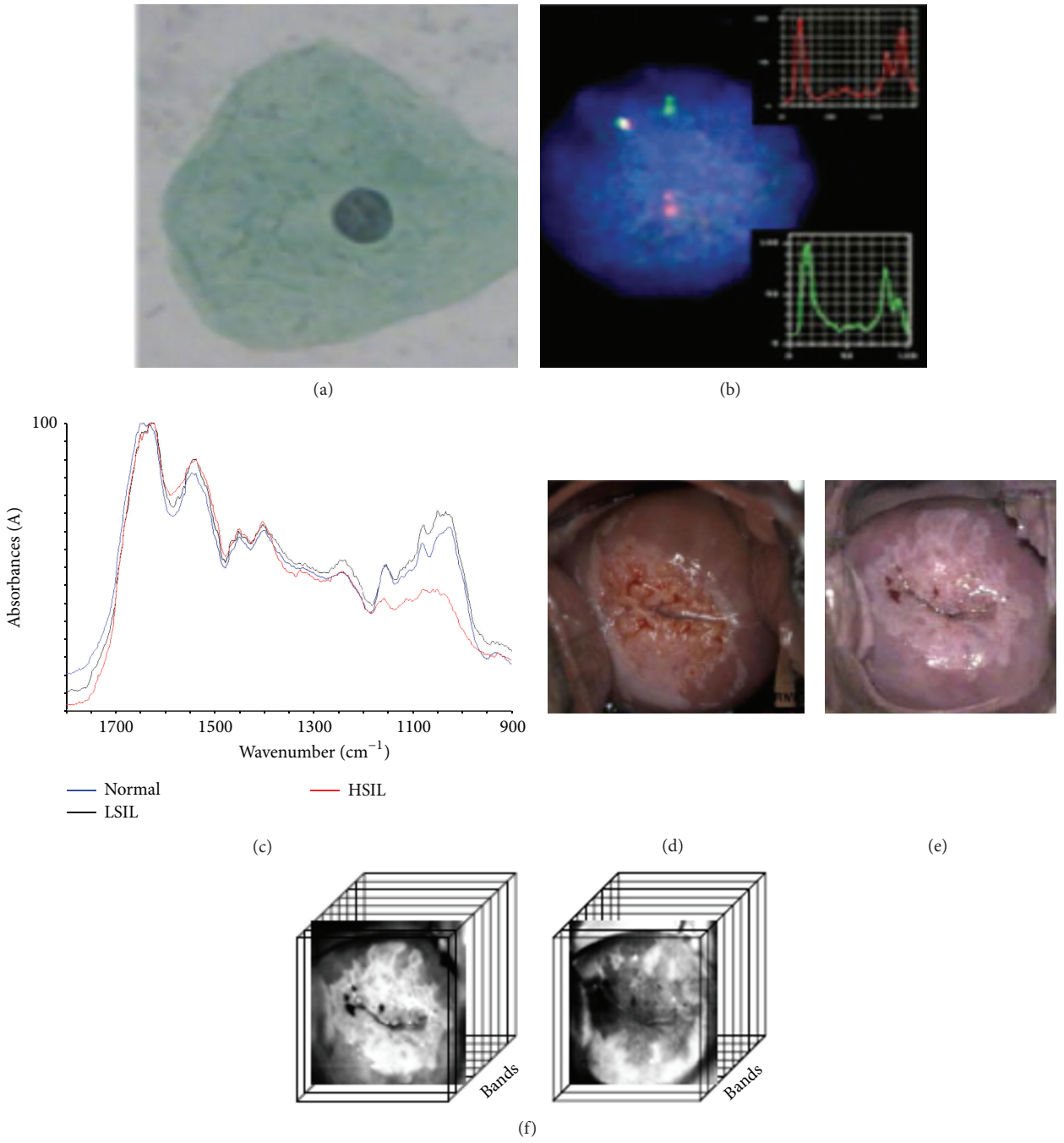


Figure 4: Cervical data used for intelligent classification. Celluler-level features; (a) cytology image, (b) FISH image, and (c) optical spectra. Tissue-level features; (d) cervicography, (e) colposcopy, and (f) optical image (HSDI).

The aim of noise reduction for the spectrum is to reduce high frequency noise contained in the spectrum that can be from either noise conducted through power lines or radiated through the hot air in the electromagnetic spectroscopy equipment [44]. Savitzky-Golay (SG) filter is currently being used widely for smoothing the spectroscopy spectra [45–52]. The SG filter has boundary problems which can be solved by using other techniques such as Binomial and Chebyshev filters [53–55].

For image data, image noise is random (not present in the object image) variation of brightness or color information in images and is usually an aspect of electronic noise. The noise is an undesirable by-product of image capture that adds spurious and extraneous information. It can compromise the level of detail in cervix image, and so reducing this noise can greatly enhance the image. There are several noise reduction techniques offered by many researchers for the automated cervical cancerous applications system as follows.

(i) Based on pixel intensity: thresholding [43, 56–60] and filtering techniques [57, 60, 61].

(ii) Based on shape: mathematical morphology [58, 60, 62].

(iii) Based on the gradient: [63, 64].

Thresholding and filtering are to reduce the noise by making use of the pixel intensities. In threshold, the intensity histogram of an image is employed to determine the threshold value where the pixels are considered to be noise. For example, the Otsu method determines an optimal threshold which minimizes the within-class variance [62]. This method yields satisfactory results when the numbers of pixels in each class are close to each other. One weakness of threshold is that all pixels under the threshold value can be noise even the pixel information which is important. Conversely, the pixels over the threshold value can be information even the pixels which are noise. In filtering, the value of a pixel is transformed to a new value which is computed as a function of the values of pixels located in a selected neighborhood around this particular pixel. This is an improvement over the threshold method.

Another method for noise reduction which reduces the noise based shape characteristics of the input image is to use mathematical morphology. The basic morphological operators are the erosion and dilation of the set with a structuring element. These two basic transformations give two other transformations known as opening and closing. Opening is the erosion of an image followed by the dilation; it breaks narrow isthmuses and eliminates small objects and sharp peaks in the image. On the other hand, closing is the dilation of an image followed by the erosion; it fuses narrow breaks and fills tiny holes and gaps in the image [58, 65]. This technique can enhance region of interest (ROI) of the images perfectly by removing and adding small shape in the focused images.

Meanwhile, the segmentation process is used to detect the region of interest in the cervical image. The process is a key procedure in automating computer-aided diagnostic systems, because accurate images segmentation could help to reduce the processing time and increase the sensitivity rates. The segmentation method should be chosen depending on the type of the features to be extracted. Several segmentation techniques have been proposed and applied in cervix images as follows.

(i) Based on shape: [57, 58, 66].

(ii) Based on color: [61, 67–70].

(iii) Based on texture: [61, 71].

(iv) Based on contour: [59, 72–74].

4.2. Features Extraction. Automated cervical cancer diagnosis relies on using the information obtained from (i) the abnormalities in the cell structures (*cellular-level*) and (ii) the abnormalities in the cell distribution across the tissue (*tissue-level*). Many researchers have applied various captured techniques for the automated classification of cervical cancer. The techniques are cytology, FISH, and electromagnetic scanner for cellular level while cervicography, colposcopy, and HSDI are used for tissue level. Features are then extracted from data of the techniques as presented in Table 3.

The features are extracted to quantify these changes in a given tissue. In order to measure the abnormalities at the cellular/tissue level, size and shape, ratio, topology, texture, and color intensity can be used as features listed in Table 3. The features are extracted and represented by a value to be used in the intelligent system.

4.2.1. Size and Shape Feature. A cell includes a nucleus surrounded by cytoplasm. As a traditional way, a pathologist evaluates the cytoplasm and the background of slide. The abnormality features are described as size (i.e., there is an increased size of the nucleus compared to the cytoplasm), shape (i.e., smooth, circular, and oval outline belongs to a normal nucleus), texture (i.e., rough textures belong to an abnormal nucleus), chromaticity (i.e., abnormal nucleuses are darker than normal ones) [62]. The quantification of these properties enables differentiating the malignant cells from those of benign and normal.

The size is expressed by the radius, area, and perimeter of the cell. Suppose that $S = \{s_1, \ldots, s_n\}$ is a set of the boundary points of a segmented cell/nucleus and C is the centroid of these boundary points, a sample of a nucleus with its boundary points. On the other hand, the shape is expressed by the length of the major and minor axes, symmetry, and circularity. The size and shape features defined on the set of the boundary points, S, are given as follows.

(i) Radius r is defined as the average length of the radial lines towards every boundary point. Mathematically,

$$r = \frac{\sum_{i=1}^{n} |s_i C|}{n}. \quad (1)$$

(ii) Area is the number of pixels within the boundary.

(iii) Perimeter P is measured as the sum of the distances between every consecutive boundary point. Mathematically,

$$P = |s_n s_1| + \sum_{i=1}^{n-1} |s_i s_{i+1}|. \quad (2)$$

(iv) Major axis is the longest chord that goes through the center and minor axis is the line that is perpendicular to the major axis and that goes through the center.

(v) Circularity is quantified by drawing chords between nonadjacent boundary points and checking whether or not the boundary points lie inside these chords.

Several researchers have identified capability of the size and shape features to classify the cervix using the cytology image [72, 73, 75], FISH image [60, 76, 77], and electromagnetic spectrum [24, 78]. Besides these features, the ratio of the same feature for different parts of a biological structure is used as another feature. For example, the nuclear area/cytoplasm area ratio [73] and the corrected area under peak A/under peak B ratio [78] are such a kind of features.

Table 3: The list of features that are extracted by different data.

	Cellular-level based features			Tissue-level based features		
	Cytology	FISH	Electromagnetic spectra	Cervicography	Colposcopy	HSDI image
Size	(i) Area of Cell [72], (ii) Area of Nucleus [72, 73, 75] (iii) Area of Cytoplasm [75].	(i) Area for each coloured spot [60, 76, 77]. (ii) Radius of each coloured spot [60],	Shift of peak frequency [24]	Perimeter of anatomical features [63, 79]	Perimeter of anatomical features [80]	Perimeter of acetowhite [27]
Shape	(i) Circularity of cytoplasm [75] (ii) Circularity of nucleus [24, 75]	Circularity of each coloured spot [60, 77].		(i) Circularity of cervix [66] (ii) Circularity or elliptical shape of Os region [66]		
Ratio	(i) Percentage of cell coverage [72] (ii) Ratio of nucleus to cytoplasm size [72, 75] (iii) Percentage of empty cells [72].		(i) Ratio of peak intensities [24, 78] (ii) Ratio of area under peaks [78]			
Topology	(i) Distribution of cell [72] (ii) Distribution of nucleus [72],	(i) Distances between the same color spots [60, 77]. (ii) Distance between the centers of the two spots [60, 77]. (iii) Center of gravity for each coloured spot [60], (iv) Number of red and green spots [60, 76, 77].				
Texture	(i) Multinucleus cells [72], (ii) Halos in cells [72].			Acetowhite region [86–88],	Acetowhite region [61, 71, 89–91]	
Color intensity	(i) Cell [72, 83] (ii) Nucleus [73] (iii) Cytoplasm [73]	Intensity of each coloured spot [60].		Anatomical features [86, 93, 94]	Anatomical features [61, 67, 91, 95–97].	

From cytology images as presented in Figure 4(a), the specific features as listed in Table 3 (i.e., size, shape, and ratio), namely, average nucleus size [72, 73, 75], average cytoplasm size [75], average cell size [72], cytoplasm circularity [75], nucleus circularity [24, 75], percentage of cell coverage [72], ratio of a nucleus to cytoplasm size [72, 75], and percentage of empty cells [72], are partially used to be an input attribute to the classification system.

For FISH image, the features from labeled biomarker spots of chromosomes 3 (red spot) and X (green spot) are the size of each colored spot [60, 76, 77], the effective radius of each red or green spot computed as the radius of a circle that had the same size as the colored spot [60, 76, 77], and the circularity of each colored spot [60, 77].

Meanwhile, from the electromagnetic spectra, the features are shift of peak frequency [24], absorbance value, and area under the spectra. For the absorbance features, the corrected absorbance value and ratio of the absorbance/corrected absorbance values for certain regions in a spectrum are derived from the features [24, 78]. Then, from area under the spectra, the features can be taken as corrected area and ratio of the area/corrected area values for certain regions in one spectrum [78].

At the case of tissue-level image, the shape feature is applied to differentiate the cervix images. The anatomical region features of the cervix (as marked by the medical experts) can be characterized by their elliptical or circular shapes; hence, the ellipse and the circle are chosen for the shape models. A vast amount of work was done to embed prior-shape information into a segmentation task. A popular approach is to use prior models based on allowable deformation of a template shape [66]. In addition, for tissue level case, the AW perimeter obtained after Lugol's solutions was assessed by examining the topography of the perimeter lines cut across the image contour with lines positioned in radial direction [27, 63, 79, 80].

Several techniques are applied to extract the size and shape features:

(i) thresholding technique [60, 62, 77, 81];

(ii) clustering technique [70, 73];

(iii) fuzzy technique [69];

(iv) wavelet technique [82, 83];

(v) statistic techniques [13, 22, 78, 84].

At cellular-level, the size and shape features in cytology images are extracted using thresholding [62], clustering [70, 73], fuzzy [69], and wavelet techniques [83]. In the FISH images, the features are extracted using thresholding [60, 77, 81]. Besides, in the electromagnetic spectra, the features are extracted using statistical techniques [13, 22, 78, 84] and wavelet technique [82]. At tissue-level, perimeters were analyzed in terms of their topology changes such as perimeter' peaks [80]; van Raad et al. [68] used landmark technique of the closed contours to extract the perimeter features which differentiate normal and abnormal cervix. In the HSDI image case, the perimeter feature is extracted using landmark technique after an enhancement process [27]. Automated landmark extraction, including the extraction of the cervix boundary, detection of the Os (one of the anatomic region), and detections (and elimination) of specular reflections are used by [63, 79, 85].

4.2.2. Topology Features. The topological features provide information on the structure of a tissue by quantifying the spatial distributionof its cells. For that, this approach encodes the spatial interdependency of the cells prior to the feature extraction. The features are applied for cellular-level case. The specific features implemented for cytology images are distribution of cell [72] and distribution of the nucleus [72], while the distances between the same color spots [60, 77], the distance between the centers of the two spots [60, 77], the gravity center of each colored spot [60], and the total number of red spots and green spots [60, 76, 77] have been implemented in the FISH images. The thresholding techniques are applied to extract the features in the cellular-level case [60, 72].

4.2.3. Textural Features. Texture is a connected set of pixels that occurs repeatedly in an image. It provides information about the variation in the intensity of a surface by quantifying properties such as smoothness, coarseness, and regularity. At the cellular level, the existence of multinuclear cells [72] and the existence halos in cells [72] are used as features in the cytology image. Meanwhile, at the tissue level, the texture features are extracted from the AW region of the cervix image [61, 71, 86–91]. The texture is formed after giving the acetic acid or Lugol's iodine to the cervix surface as a sign of the abnormality.

There are several techniques applied for extracting the textural features in cervix images as follows.

(i) Wavelet technique [89].

(ii) Mathematical morphological operations [71, 90].

(iii) Clustering technique [86].

(iv) Thresholding technique [62, 88].

At the cervix image of tissue-level, van Raad [89] demonstrated Gabor wavelet for extracting the textural features which outline the area of metaplastic changes, known as the transformation zone (TZ). The performances of the Gabor wavelet scheme achieve close to 80% accuracy in discrimination on the ROI. On the other hand, textural features (i.e., mosaic pattern) within the AW region are obtained from skeletonized vascular structures uniquely. The skeletonized vascular structures represented typical vascularity embedded in the normal and abnormal regions extracted by a series of mathematical morphological operations [71]. The series of mathematical morphological operations are gray-scale method, top hat transform, morphological opening with a rotating structuring element (ROSE), thresholding, and skeletonizing. Similarly, the textural features are extracted based on iterative morphological operations with various sizes of structural elements, in combination with adaptive thresholding [90]. Furthermore, combination of mathematical morphology and clustering based on Gaussian mixture model (GMM) is proposed to extract the textural features in the cervix image [86]. The algorithms are used to segment macro regions of the textural cervix images. Thresholding technique is used to segment tissues and nucleus as the texture for each application, respectively [62, 88, 92].

4.2.4. Color Intensity Based Features. The color intensity-based features are extracted from the gray-level or color histogram of the image. This type of features does not provide any information about the spatial distribution of the pixels. The intensity histogram in a cell is employed to define features. In the case of cellular level images, the difference of color intensity can be used as features for the cancerous cells [72, 73]. Cytology image has a relatively darker color intensity composition than normal cells. The distinguishable patterns can be analyzed using the corresponding image's color intensity histogram [83]. Meanwhile, another feature to differentiate the abnormality of cervix using FISH image is the average intensity of each colored spot [60]. At the case of tissue level images, the changes in color and intensity correlate closely with changes in tissue type, severity of cervical neoplasia, and vessel patterns [61, 67, 86, 91, 93–97].

Several techniques used for extracting the intensity features are as follows.

(i) Clustering technique [67, 86, 95, 97].

(ii) Watershed technique [93, 94].

(iii) Statistical technique [96].

van Raad [67] used a clustering technique (i.e., GMM) based MAP algorithm probability model in cervical images to extract color information features belonging to each of the tissue types in the cervix, such as the cervical canal (CC), the transformation zone (TZ), the squamous epithelium (SE), and the artifact named specular reflection (SR). Besides, mean-shift clustering is used to extract color and texture features of a tissue type [95]. Clustering based on the GMM

Table 4: The list of classifiers that are used by different studies.

	Cellular-level based features			Tissue-level based features		
	Cytology	FISH	Electromagnetic spectra	Cervicography	Colposcopy	HSDI image
Artificial Neural network	(3/1241/10/78.7) [56], (2/400/10/99) [72], (3/550/4/97.5) [73], (5/78/5/91.4) [75].		(2/361/13/74.4) [84], (2/201/3/87) [82], (3/780/22/97.4) [78].		(2/283/7/95.8) [80].	
Support vector machine			(3/63/10/72) [116]			
Logistic regression			(4/145/—/88) [12]			
K-nearest neighbors					(2/283/7/68.9) [80], (2/48/10/76.5) [97].	(7/371/5/95.96) [117]
Linear discriminant analysis	(5/230/15/60.4) [85].		(2/324/—/78) [118], (2/275/8/96.4) [22], (2/150/5/99.5) [13], (4/800/7/90) [24], (2/92/3/97.6) [119]	(2/100/—/78.5) [88].	(2/40/4/87.2) [61].	
Decision trees	(3/1241/10/77) [56], (—/61/2/96.7) [120].	(2/325/—/93.6) [60]		(2/211/—/78) [94].	(2/29/—/86) [121], (2/99/—/88.5) [96].	

The values given in bracket are number of classes/number of data/number of features used/accuracy.

is used in a joint color and geometric feature space to segment macro regions [86]. Similarly, [97] used a clustering technique (i.e., K-means clustering (KMC)) to generate an anatomical feature map for each cervical tissue type. The tissue regions defined by the anatomical feature map are further clustered into subregions. Watershed technique is used for a specific focus on the detection of lesion regions in uterine cervix images [93, 94]. Meanwhile, the spatial change of the AW lesion is extracted using color and texture information based on an opacity index that indicates the grades of temporal change [96].

As presented in Figure 4, possibility of the ratio and texture features can be extracted from FISH image for future works. As listed in Table 3, the features of the FISH image are area and radius for each colored spot. The ratio of the area for one colored spot and other colored spot can be possibly extracted. The ratio of the radius of one colored spot and other colored spots can be also possibly extracted. The texture of one FISH image integrally can be also extracted to differentiate the abnormality of the images.

4.3. Features Selection. After all the possible features for classification had been extracted, the selection of significant or dominant features can be conducted. Besides feature's extraction systems, the classification performance also depends on the selected features and the classification technique used. Feature selection is an important stage in classification, especially if it involves a large dimension of input features. By applying this feature selection stage, the original high dimensional inputs could be transformed and reduced into new lower dimensional features [98].

Generally, all possible extracted features can be used as the inputs for a classification system. However, irrelevant or noisy features could deteriorate between classes and increase the overlap in a non-linear manner. The noisy features can mix up the boundaries for the generalization performance of the classification system [99]. A clasifier with fewer inputs needs fewer weights to be adjusted, leading to better generalization and faster training [100]. Adding newer features can significantly lead to a reduction in the performance of the classification system [100].

Many researchers in computer vision based spectroscopy data applied the features selection techniques for cervical cells and other cell features [13, 51, 58, 84, 98, 101–110]. Generally, good performances in classification are achieved after applying the features selection techniques. Since the spectral data is heavily redundant, the selection of the significant wavelength as features in this case is vital.

For the image processing application, discriminant analysis (DA) and principle component analysis (PCA) are methods commonly used to find a linear combination of features which characterizes or separates two or more classes of objects or events. The resulting combination may be used as a linear classifier or, more commonly, for dimensionality reduction before later classification [111].

The DA works by creating a new variable called the discriminant function score which is used to predict to which group a case belongs. The discriminant function scores are

computed similarly to factor scores (i.e., using eigenvalues). The computations find the coefficients for the independent variables (features) that maximize the measure of distance between the groups defined by the dependent variable. The disadvantages of the DA are the distribution of distance matrices in the same class to be singular if the dimension of the data is much higher than the number of training samples [112].

Besides, the PCA is mathematically defined as an orthogonal linear transformation that transforms the data to a new coordinate system. In the PCA, the greatest variance by any projection of the data comes to lie on the first coordinate (called the first principal component), the second greatest variance on the second coordinates, and so on. The first principal component corresponds to a line that passes through the multidimensional mean and minimizes the sum of squares of the point's distances from the line. The second principal component corresponds to the same concept after all correlations with the first principal component have been subtracted out from the points.

There are several researchers in cervical cancer application who use the PCA [13, 17, 113]. In their researches, the PCA is used as dimensionality reduction to improve the classification performance and decrease the training time of classifier. However, the disadvantages of the PCA consist in the fact that the directions maximizing variance do not always maximize information. In case, a great disadvantage of PCA is that it does not consider any class information [114]. This can lead to a loss of important discriminating information. In fact, the analysis showed that it was practically impossible to improve the classification error by this method [114]. Another disadvantage of the PCA is that it has high memory and computational requirements [115].

4.4. Classification. The effectiveness of the automatic cervical precancerous screening system is evaluated in this section. The classifiers mostly used for cervical cancer study in detail are artificial neural networks or neural network (NN) [56, 72, 73, 75, 78, 80, 82, 84], support vector machine (SVM) [116], logistic regression [12], K-nearest neighborhood (KNN) [80, 97, 117], linear discriminant analysis (LDA) [13, 22, 24, 61, 85, 88, 118, 119], and decision trees [56, 60, 94, 96, 120, 121], as listed in Table 4. The performances of the classifiers generally showed good results as presented in Figure 5.

Generally, each type of classifiers can be employed for all types of data. For example, the FISH or cervicography data can be classified using NN, SVM, logistic regression, KNN, LDA, and decision tree. However, it is important to know the advantages and disadvantages of the classifiers that might be considered as alternatives. Logistic regression is attractive for probability prediction, because it is mathematically constrained to produce probabilities in the range [0, 1] and generally converges on parameter estimates relatively easily [122]. The disadvantages of the logistic regression are not designed to deal with high-dimensional data and cannot approximate any smooth polynomial function, regardless of the order of the polynomial or the number of interaction terms [122].

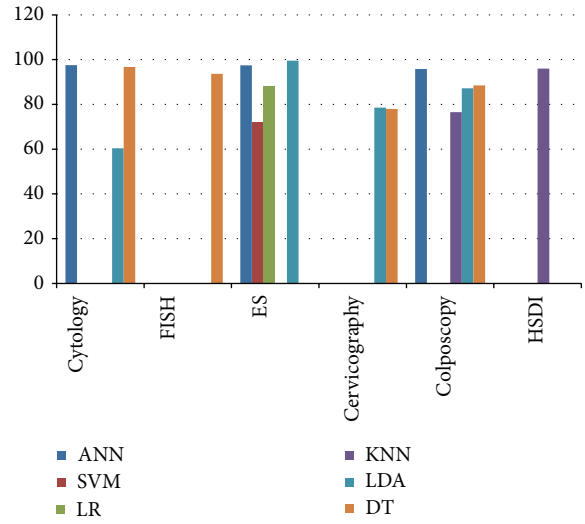


Figure 5: Performances of six classifiers generally for cervical precancerous data.

SVM's execution speed is very fast and there are no parameters to tune except the constant C. It is remarkably intolerant of the relative sizes of the number of training examples of the two classes. Since the technique is not directly trying to minimize the error rate, but trying to separate the patterns in high dimensional space, the result is that SVM is relatively insensitive to the relative numbers of each class. The possible disadvantages are large memory requirement [123] and the training time can be very large if there are large numbers of training examples [124].

Meanwhile, the NN architecture is initially not structured and the learning algorithm is responsible for the extraction of the regularities present in the data by finding a suitable set of synapses during the process of observation of the examples. Thus, NNs solve problems by self-learning and self-organization [125]. However, the neural network required long training time, and the results depend on the initialization parameters. It consisted of an arbitrary number of layers, and parameters [122]. Different combinations of number of hidden neurons, learning rate, momentum rate, activation function, epoch size, and initial weights have to be tried in order to produce better results [125].

Decision tree is relatively easy to interpret and to implement. Like SVMs and NNs, many methods for decision trees do not provide a probability of class membership, although some variants, in particular, classification and regression trees, do provide such probabilities. However, performance of all decision trees is dependent on both their method of construction and the amount of pruning (removal of highly specific nodes) performed [122].

KNN and LDA are methods implemented in numerous programs and easy to be implemented as classification tools. Both techniques have direct analytical solution and very good at detecting global phenomena (whereas decision tree detects local phenomena). However, they are simply defined and implemented, especially if there is insufficient data to

adequately define sample means and covariance matrices. Both techniques only detect linear phenomena and are sensitive to individuals outside the norm.

From Table 4 and Figure 5, the NN results showed constantly higher performance results in terms of accuracy than the other classifiers. The result ranges are 78.7–99% of accuracy. Mostly, the accuracy results are higher than 90%. In detail, [56] achieved 78.7% of accuracy in their preliminary study in classifying more than 1000 data to be two classes. [72, 73, 75, 78, 80] successfully achieved more than 90% of accuracies (i.e., 99%, 97.5%, 91.4%, 95.8%, and 97.4%) to classify 400 data to be 2 classes, 550 data to be 3 classes, 78 data to be 5 classes, 780 data to be 3 classes, and 283 data to be 2 classes.

As presented in Table 4, six types of data are used for classification purpose. All data have good capability to be used as intelligent classification data. The classification performances of the data are spread from range 60 to 99% of accuracies. Overall performance shows that cytology features and the electromagnetic spectra features give the higher accuracy than the other data. Many of the researchers that use the data gain accuracy values more than 90% such as performances using cytology data: 96.7% [120], 99% [72], 97.5% [73], and 91.4% [75] and performances using electromagnetic spectra data: 96.4% [22], 99.5% [13], 90% [24], 97.6% [119], and 97.4% [78]. Only few have performance less than 90% of accuracy.

The cytology combined with neural network gives the accuracy of up to 99% of accuracy to classify 400 data to be 2 classes, followed by neural network using the electromagnetic spectra features at 97.4% for classifying 780 data to be 3 classes. Greatly, the electromagnetic spectra features could achieve the higher accuracy only using discriminant analysis at 99.5% of accuracy. Therefore, based on Table 4, the better cervix data used for the automated diagnosis are the cytology and the electromagnetic spectra features and the best classifier used for the automated diagnosis system is neural network.

As reviewed, the intelligent classification system for cervical precancerous cells has been attempted and developed using two types of input attributes; cervical cell/tissue images and cervical cell spectra. Therefore, the systems have employed image and signal processing techniques for extracting features as the input attributes, respectively. Both systems could classify the cervical precancerous cells with high performances. The applications of image and spectra processing and classifier for cervical precancerous classification have been developed by many researchers in the world. The screening techniques have been proven to have better performance than the manual techniques. Thus, the intelligent classification system for cervical precancerous using the image and/or optical spectra as input is believed to have better classification performance and could be used as a second opinion to pathologists.

5. Summary

Six types of cervical precancerous data (i.e., cytology, FISH, electromagnetic spectra, cervicography, colposcopy, and HSDI) generally can be used for the intelligent screening of cervical cancer. Computer screening system for cervical cancer based on cellular level data, namely, cytology, FISH, and electromagnetic spectroscopy, achieved better results as compared to tissue level data such as cervicography and colposcopy.

Classification tools (i.e., ANN, SVM, logistic regression, KNN, LDA, and decision tree) generally can achieve good performances to classify the cervical precancerous data. The screening systems based on neural network technique are frequently applied due to the better results and potential of the technique to build a real time system.

The long training time of the neural network can be reduced by using the features selection stage in the computer screening system. The dimensionality reduction popularly done by using discriminant analysis and principal component analysis can be developed using new techniques that can be proposed as future work in this research field. The developed techniques will reduce the training time and improve the classification result of the neural network.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This research is supported by UM High Impact Research Grant UM-MOHE UM.C/625/1/HIR/MOHE/14 from the Ministry of Higher Education Malaysia.

References

[1] NCCC, "Cervical cancer," http://www.nccc-online.org/index.php/cervicalcancer.

[2] ACS, "What is cervical cancer?" 2011, American Cancer Society, http://www.cancer.org/Cancer/CervicalCancer/DetailedGuide/cervical-cancer-what-is-cervical-cancer.

[3] H. S. Cronjé, "Screening for cervical cancer in the developing world," *Best Practice and Research: Clinical Obstetrics and Gynaecology*, vol. 19, no. 4, pp. 517–529, 2005.

[4] K. Frankel and M. K. Sidawy, "Formal proposal to combine the papanicolaou numerical system with Bethesda terminology for reporting cervical/vaginal cytologic diagnoses," *Diagnostic Cytopathology*, vol. 10, no. 4, pp. 395–396, 1994.

[5] R. A. Kerkar and Y. V. Kulkarni, "Screening for cervical cancer: an overview," *Obstetrics and Gynecology of India*, vol. 56, no. 2, pp. 115–122, 2006.

[6] P. Segers, S. Haesen, P. Castelain et al., "Study of numerical aberrations of chromosome 1 by fluorescent in situ hybridization and DNA content by densitometric analysis on (pre)-malignant cervical lesions," *Histochemical Journal*, vol. 27, no. 1, pp. 24–34, 1995.

[7] C. Mian, D. Bancher, P. Kohlberger et al., "Fluorescence in situ hybridization in cervical smears: detection of numerical aberrations of chromosomes 7, 3, and X and relationship to HPV infection," *Gynecologic Oncology*, vol. 75, no. 1, pp. 41–46, 1999.

[8] H. Vrolijk, W. C. R. Sloos, F. M. van de Rijke et al., "Automation of spot counting in interphase cytogenetics using brightfield microscopy," *Cytometry*, vol. 24, no. 2, pp. 158–166, 1996.

[9] G. Méhes, N. Speich, M. Bollmann, and R. Bollmann, "Chromosomal aberrations accumulate in polyploid cells of High-grade Squamous Intraepithelial Lesions (HSIL)," *Pathology and Oncology Research*, vol. 10, no. 3, pp. 142–148, 2004.

[10] J. T. Bryan, F. Taddeo, D. Skulsky et al., "Detection of specific human papillomavirus types in paraffin-embedded sections of cervical carcinomas," *Journal of Medical Virology*, vol. 78, no. 1, pp. 117–124, 2006.

[11] A. H. N. Hopman, W. Theelen, P. P. H. Hommelberg et al., "Genomic integration of oncogenic HPV and gain of the human telomerase gene TERC at 3q26 are strongly associated events in the progression of uterine cervical dysplasia to invasive cancer," *Journal of Pathology*, vol. 210, no. 4, pp. 412–419, 2006.

[12] E. M. Kanter, E. Vargis, S. Majumder et al., "Application of raman spectroscopy for cervical dysplasia diagnosis," *Journal of Biophotonics*, vol. 2, no. 1-2, pp. 81–90, 2009.

[13] C. M. Krishna, N. B. Prathima, R. Malini et al., "Raman spectroscopy studies for diagnosis of cancers in human uterine cervix," *Vibrational Spectroscopy*, vol. 41, no. 1, pp. 136–141, 2006.

[14] S. K. Chang, Y. N. Mirabal, E. N. Atkinson, A. Malpica, M. Follen, and R. Richards-Kortum, "Combination of fluorescence and reflectance spectroscopy for in vivo detection of cervical pre-cancers," in *Proceedings of the IEEE Engineering in Medicine and Biology 24th Annual Conference and the Fall Meeting of the Biomedical Engineering Society (BMES/EMBS '02)*, pp. 2265–2266, Houston, Tex, USA, October 2002.

[15] Z. Huang, J. Mo, W. Zheng, J. Low, J. Ng, and A. Ilancheran, "Combining near-infrared autofluorescence and raman spectroscopy improves the in vivo detection of cervical precancer," in *Proceedings of the Conference on Quantum Electronics and Laser Science Conference on Lasers and Electro-Optics (CLEO/QELS '08)*, San Jose, Calif, USA, May 2008.

[16] P. T. T. Wong, R. K. Wong, T. A. Caputo, T. A. Godwin, and B. Rigas, "Infrared spectroscopy of exfoliated human cervical cells: evidence of extensive structural changes during carcinogenesis," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 88, no. 24, pp. 10988–10992, 1991.

[17] M. A. Cohenford, T. A. Godwin, F. Cahn, P. Bhandare, T. A. Caputo, and B. Rigas, "Infrared spectroscopy of normal and abnormal cervical smears: evaluation by principal component analysis," *Gynecologic Oncology*, vol. 66, no. 1, pp. 59–65, 1997.

[18] M. F. K. Fung, M. Senterman, P. Eid, W. Faught, N. Z. Mikhael, and P. T. T. Wong, "Comparison of fourier-transform infrared spectroscopic screening of exfoliated cervical cells with standard papanicolaou screening," *Gynecologic Oncology*, vol. 66, no. 1, pp. 10–15, 1997.

[19] L. Chiriboga, P. Xie, H. Yee, D. Zarou, D. Zakim, and M. Diem, "Infrared spectroscopy of human tissue. IV. Detection of dysplastic and neoplastic changes of human cervical tissue via infrared microscopy," *Cellular and Molecular Biology*, vol. 44, no. 1, pp. 219–229, 1998.

[20] M. Diem, L. Chiriboga, P. Lasch, and A. Pacifico, "IR spectra and IR spectral maps of individual normal and cancerous cells," *Biopolymers—Biospectroscopy Section*, vol. 67, no. 4-5, pp. 349–353, 2002.

[21] M. J. Romeo, B. R. Wood, M. A. Quinn, and D. McNaughton, "Removal of blood components from cervical smears: implications for cancer diagnosis using FTIR spectroscopy," *Biopolymers—Biospectroscopy Section*, vol. 72, no. 1, pp. 69–76, 2003.

[22] R. Sindhuphak, S. Issaravanich, V. Udomprasertgul et al., "A new approach for the detection of cervical cancer in Thai women," *Gynecologic Oncology*, vol. 90, no. 1, pp. 10–14, 2003.

[23] B. R. Wood, L. Chiriboga, H. Yee, M. A. Quinn, D. McNaughton, and M. Diem, "Fourier transform infrared (FTIR) spectral mapping of the cervical transformation zone, and dysplastic squamous epithelium," *Gynecologic Oncology*, vol. 93, no. 1, pp. 59–68, 2004.

[24] S. G. El-Tawil, R. Adnan, Z. N. Muhamed, and N. H. Othman, "Comparative study between Pap smear cytology and FTIR spectroscopy: a new tool for screening for cervical cancer," *Pathology*, vol. 40, no. 6, pp. 600–603, 2008.

[25] M. F. Parker, J. P. Karins, and D. M. O'Connor, "Hyperspectral diagnostic imaging of the cervix: initial observations," in *Proceedings of the IEEE Pacific Medical Technology Symposium*, pp. 144–148, Honolulu, Hawaii, USA, August 1998.

[26] C. Balas, "A novel optical imaging method for the early detection, quantitative grading, and mapping of cancerous and precancerous lesions of cervix," *IEEE Transactions on Biomedical Engineering*, vol. 48, no. 1, pp. 96–104, 2001.

[27] H. Lange, *Reflectance and Fluorescence Hyperspectral Elastic Image Registration*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2007.

[28] A. Stafl, "Cervicography: a new method for cervical cancer detection," *The American Journal of Obstetrics and Gynecology*, vol. 139, no. 7, pp. 815–825, 1981.

[29] K. E. Hartmann, K. Nanda, S. Hall, and E. Myers, "Technologic advances for evaluation of cervical cytology: is newer better?" *Obstetrical and Gynecological Survey*, vol. 56, no. 12, pp. 765–774, 2001.

[30] D. V. Coleman, "Evaluation of automated systems for the primary screening of cervical smears," *Current Diagnostic Pathology*, vol. 5, no. 2, pp. 57–64, 1998.

[31] J. Karnon, J. Peters, J. Platt, J. Chilcott, E. McGoogan, and N. Brewer, "Liquid-based cytology in cervical screening: an updated rapid and systematic review and economic anlysis," *Health Technology Assessment*, vol. 8, no. 20, 2004.

[32] R. Lozano, "Comparison of computer-assisted and manual screening of cervical cytology," *Gynecologic Oncology*, vol. 104, no. 1, pp. 134–138, 2007.

[33] D. Schledermann, T. Hyldebrandt, D. Ejersbo, and B. Hoelund, "Automated screening versus manual screening: a comparison of the ThinPrep imaging system and manual screening in a time study," *Diagnostic Cytopathology*, vol. 35, no. 6, pp. 348–352, 2007.

[34] P. Pudil, F. Ferri, J. Novovicova, and J. Kittler, "Floating search methods for feature selection with nonmonotonic criterion functions," in *Proceedings of the IEEE 12th IAPR International Conference on Pattern Recognition: Conference A: Computer Vision and Image Processing*, pp. 279–283, Jerusalem, Israel, 1994.

[35] P. Pudil, K. Fuka, K. Beranek, and P. Dvorak, "Potential of artificial intelligence based feature selection methods in regression models," in *Proceedings of the IEEE 3rd International Conference on Computational Intelligence and Multimedia Applications*, New Delhi, India, 1999.

[36] S. J. Reeves and Z. Zhe, "Sequential algorithms for observation selection," *IEEE Transactions on Signal Processing*, vol. 47, no. 1, pp. 123–132, 1999.

[37] R. A. Johnson and D. W. Wichern, *Applied Multivariate Statistical Analysis*, Prentice Hall, San Francisco, Calif, USA, 6th edition, 2007.

[38] K.-L. Lin, C.-Y. Lin, C.-D. Huang et al., "Feature selection and combination criteria for improving accuracy in protein structure prediction," *IEEE Transactions on Nanobioscience*, vol. 6, no. 2, pp. 186–196, 2007.

[39] R. Ashfaq, B. Solares, and M. H. Saboorian, "Detection of endocervical component by PAPNET(TM) system on negative cervical smears," *Diagnostic Cytopathology*, vol. 15, no. 2, pp. 121–123, 1996.

[40] K. Losell and A. Dejmek, "Comparison of papnet-assisted and manual screening of cervical smears," *Diagnostic Cytopathology*, vol. 21, no. 4, pp. 296–299, 1999.

[41] D. C. Wilbur, T. A. Bonfiglio, M. A. Rutkowski et al., "Sensitivity of the autoPap 300 QC system for cervical cytologic abnormalities: biopsy data confirmation," *Acta Cytologica*, vol. 40, no. 1, pp. 127–132, 1996.

[42] M. Negnevitsky, *Artificial Intelligence: A Guide to Intelligent Systems*, Addison Wesley, Harlow, UK, 2nd edition, 2005.

[43] P. Sobrevilla, E. Lerma, and E. Montseny, "An approach to a fuzzy-based automatic pap screening system-FAPSS-addressed to cytology cells detection," in *Proceedings of the IEEE 22nd International Conference of the North American Fuzzy Information Processing Society*, pp. 138–142, Chicago, Ill, USA, July 2003.

[44] J. Kauppinen and J. Partanen, *Fourier Transforms in Spectroscopy*, Wiley-VCH, Weinheim, Germany, 2001.

[45] H. Fabian, N. A. N. Thi, M. Eiden, P. Lasch, J. Schmitt, and D. Naumann, "Diagnosing benign and malignant lesions in breast tissue sections by using IR-microspectroscopy," *Biochimica et Biophysica Acta—Biomembranes*, vol. 1758, no. 7, pp. 874–882, 2006.

[46] F. Bonnier, S. Rubin, L. Ventéo et al., "In-vitro analysis of normal and aneurismal human ascending aortic tissues using FT-IR microspectroscopy," *Biochimica et Biophysica Acta—Biomembranes*, vol. 1758, no. 7, pp. 968–973, 2006.

[47] K. Das, N. Stone, C. Kendall, C. Fowler, and J. Christie-Brown, "Role of Fourier transform infrared spectroscopy (FTIR) in the diagnosis of parathyroid pathology," *Photodiagnosis and Photodynamic Therapy*, vol. 4, no. 2, pp. 124–129, 2007.

[48] S. K. Majumder, M. D. Keller, and A. Mahadevan-Jansen, "Optical detection of breast tumors—a comparison of diagnostic performance of autofluorescence, diffuse reflectance, and raman spectroscopy," in *Progress in Biomedical Optics and Imaging*, T. Vo-Dinh, R. Raghavachari, W. S. Grundfest et al., Eds., Proceedings of SPIE, no. 6430, pp. 1–11, Munich, Germany, 2007.

[49] S. K. Majumder, E. Kanter, A. R. Viehoever, H. Jones, and A. Mahadevan-Jansen, "Near-infrared raman spectroscopy for in-vivo diagnosis of cervical dysplasia—a probability-based multiclass diagnostic algorithm," in *Progress in Biomedical Optics and Imaging*, T. Vo-Dinh, R. Raghavachari, W. S. Grundfest et al., Eds., Proceedings of SPIE, no. 6430, pp. 1–11, Munich, Germany, 2007.

[50] S. K. Majumder, M. D. Keller, F. I. Boulos, M. C. Kelley, and A. Mahadevan-Jansen, "Comparison of autofluorescence, diffuse reflectance, and raman spectroscopy for breast tissue discrimination," *Journal of Biomedical Optics*, vol. 13, no. 5, Article ID 054009, 2008.

[51] N. Baheri, M. Miranbaygi, and R. Malekfar, "Improved skin xerosis detection by combining extracted features from raman spectra," in *Proceedings of the 2nd International Symposium on Applied Sciences in Biomedical and Communication Technologies (ISABEL '09)*, Bratislava, Slovakia, November 2009.

[52] K. Banas, A. Banas, H. O. Moser et al., "Multivariate analysis techniques in the forensics investigation of the postblast residues by means of fourier transform-infrared spectroscopy," *Analytical Chemistry*, vol. 82, no. 7, pp. 3038–3044, 2010.

[53] C. Palacio, C. Pascual, F. Suarez, and I. Lloret, "Smoothing of digital spectroscopic data by using a Chebyshev filter," *Vacuum*, vol. 64, no. 3-4, pp. 481–485, 2002.

[54] C. Battistoni, S. Kačiulis, G. Mattogno, and G. Righini, "Noise removal from Auger images by using adaptive binomial filter," *Journal of Electron Spectroscopy and Related Phenomena*, vol. 76, no. C, pp. 399–404, 1995.

[55] C. Battistoni, G. Mattogno, and G. Righini, "Spectral noise removal by new digital smoothing routine," *Journal of Electron Spectroscopy and Related Phenomena*, vol. 74, no. 2, pp. 159–166, 1995.

[56] J. S.-J. Lee, J.-N. Hwang, D. T. Davis, and A. C. Nelson, "Integration of neural networks and decision tree classifiers for automated cytology screening," in *Proceedings of the IEEE International Joint Conference on Neural Networks*, pp. 257–262, Singapore, July 1991.

[57] Part, P. Malm, and A. Brun, "Closing curves with Riemannian dilation: application to segmentation in automated cervical cancer screening," in *Advances in Visual Computing*, vol. 5875 of *Lecture Notes in Computer Science*, pp. 337–346, Springer, Berlin, Germany, 2009.

[58] M. E. Plissiti, C. Nikou, and A. Charchanti, "Combining shape, texture and intensity features for cell nuclei extraction in Pap smear images," *Pattern Recognition Letters*, vol. 32, no. 6, pp. 838–853, 2011.

[59] H. Lange, *Automatic Glare Removal in Reflectance Imagery of the Uterine Cervix*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2005.

[60] X. Wang, B. Zheng, S. Li et al., "Automated detection and analysis of fluorescent in situ hybridization spots depicted in digital microscopic images of Pap-smear specimens," *Journal of Biomedical Optics*, vol. 14, no. 2, Article ID 021002, 2009.

[61] W. Li, J. Gu, D. Ferris, and A. Poirson, *Automated Image Analysis of Uterine Cervical Images*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2007.

[62] B. Sokouti, S. Haghipour, and A. D. Tabrizi, "A pilot study on image analysis techniques for extracting early uterine cervix cancer cell features," *Journal of Medical Systems*, pp. 1–7, 2011.

[63] H. Greenspan, S. Gordon, G. Zimmerman et al., "Automatic detection of anatomical landmarks in uterine cervix images," *IEEE Transactions on Medical Imaging*, vol. 28, no. 3, pp. 454–468, 2009.

[64] C.-H. Lin, Y.-K. Chan, and C.-C. Chen, "Detection and segmentation of cervical cell cytoplast and nucleus," *International Journal of Imaging Systems and Technology*, vol. 19, no. 3, pp. 260–270, 2009.

[65] C. Demir and B. Yener, *Automated Cancer Diagnosis Based on Histopathological Images: A Systematic Survey*, Rensselaer Polytechnic Institute, New York, NY, USA, 2005.

[66] S. Lotenberg, S. Gordon, and H. Greenspan, "Shape priors for segmentation of the cervix region within uterine cervix images," *Journal of Digital Imaging*, vol. 22, no. 3, pp. 286–296, 2009.

[67] V. van Raad, *Image Analysis and Segmentation of Anatomical Features of Cervix Uteri in Color Space*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2005.

[68] V. van Raad, Z. Xue, and H. Lange, *Lesion Margin Analysis for Automated Classification of Cervical Cancer Lesions*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2006.

[69] P. Sobrevilla, E. Montseny, F. Vaschetto, and E. Lerma, "Fuzzy-based analysis of microscopic color cervical pap smear images: nuclei detection," *International Journal of Computational Intelligence and Applications*, vol. 9, no. 3, pp. 187–206, 2010.

[70] S. N. Sulaiman, N. A. Mat Isa, and N. H. Othman, "Semi-automated pseudo colour features extraction technique for cervical cancer's Pap smear images," *International Journal of Knowledge-Based and Intelligent Engineering Systems*, vol. 15, no. 3, pp. 131–143, 2011.

[71] B. Tulpule, S. Yang, Y. Srinivasan, S. Mitra, and B. Nutter, "Segmentation and classification of cervix lesions by pattern and texture analysis," in *Proceedings of the 14th IEEE International Conference on Fuzzy Systems (FUZZ '05)*, pp. 173–176, Reno, Nev, USA, May 2005.

[72] Z. Li and K. Najarian, "Automated classification of Pap smear tests using neural networks," in *Proceedings of the International Joint Conference on Neural Networks (IJCNN '01)*, pp. 2899–2901, Washington, DC, USA, July 2001.

[73] N. A. Mat-Isa, M. Y. Mashor, and N. H. Othman, "An automated cervical pre-cancerous diagnostic system," *Artificial Intelligence in Medicine*, vol. 42, no. 1, pp. 1–11, 2008.

[74] K. Zhang, L. Zhang, H. Song, and W. Zhou, "Active contours with selective local or global segmentation: a new formulation and level set method," *Image and Vision Computing*, vol. 28, no. 4, pp. 668–676, 2010.

[75] M. E. Gómez-Mayorga, F. J. Gallegos-Funes, J. M. de-la-Rosa-Vázquez, R. Cruz-Santiago, and V. Ponomaryov, "Diagnosis of cervical cancer using the median M-type radial basis function (MMRBF) neural network," in *Proceedings of the 8th Mexican International Conference on Artificial Intelligence*, pp. 258–267, Guanajuato, México, 2009.

[76] H. Netten, L. J. van Vliet, H. Vrolijk, W. C. R. Sloos, H. J. Tanke, and I. T. Young, "Fluorescent dot counting in interphase cell nuclei," *Bioimaging*, vol. 4, no. 2, pp. 93–106, 1996.

[77] M. Gué, C. Messaoudi, J. S. Sun, and T. Boudier, "Smart 3D-FISH: automation of distance analysis in nuclei of interphase cells by image processing," *Cytometry A*, vol. 67, no. 1, pp. 18–26, 2005.

[78] Y. Jusman, N. A. Mat Isa, R. Adnan, and N. H. Othman, "Intelligent classification of cervical pre-cancerous cells based on the FTIR spectra," *Ain Shams Engineering Journal*, vol. 3, no. 1, pp. 61–70, 2012.

[79] Z. Xue, S. Antani, L. R. Long, and G. R. Thoma, "An online segmentation tool for cervicographic image analysis," in *Proceedings of the 1st ACM International Health Informatics Symposium (IHI '10)*, pp. 425–429, Arlington, Va, USA, November 2010.

[80] I. Claude, R. Winzenrieth, P. Pouletaut, and J. C. Boulanger, "Contour features for colposcopic image classification by artificial neural networks," in *Proceedings of IEEE 16th International Conference on Pattern Recognition*, pp. 771–774, Quebec, Canada, August 2002.

[81] B. Kajtár, G. Méhes, T. Lörch et al., "Automated fluorescent in situ hybridization (FISH) analysis of t(9;22)(q34;q11) in interphase nuclei," *Cytometry A*, vol. 69, no. 6, pp. 506–514, 2006.

[82] S. Mark, R. K. Sahu, K. Kantarovich et al., "Fourier transform infrared microspectroscopy as a quantitative diagnostic tool for assignment of premalignancy grading in cervical neoplasia," *Journal of Biomedical Optics*, vol. 9, no. 3, pp. 558–567, 2004.

[83] J. Suryatenggara, B. K. Ane, M. Pandjaitan, and W. Steinberg, "Pattern recognition on 2D cervical cytological digital images for early detection of cervix cancer," in *Proceedings of the World Congress on Nature and Biologically Inspired Computing (NABIC '09)*, pp. 257–262, Coimbatore, India, December 2009.

[84] K. Turner, N. Ramanujam, J. Ghosh, and R. Richards-Kortum, "Ensembles of radial basis function networks for spectroscopic detection of cervical precancer," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 8, pp. 953–961, 1998.

[85] S. J. Keenan, J. Diamond, W. Glenn McCluggage et al., "An automated machine vision system for the histological grading of cervical intraepithelial neoplasia (CIN)," *Journal of Pathology*, vol. 192, no. 3, pp. 351–362, 2000.

[86] Y. Srinivasan, E. Corona, B. Nutter, S. Mitra, and S. Bhattacharya, "A unified model-based image analysis framework for automated detection of precancerous lesions in digitized uterine cervix images," *IEEE Journal on Selected Topics in Signal Processing*, vol. 3, no. 1, pp. 101–111, 2009.

[87] Z. Xue, L. R. Long, S. Antani, and G. R. Thoma, "Automatic extraction of mosaic patterns in uterine cervix images," in *Proceedings of the 23rd IEEE International Symposium on Computer-Based Medical Systems (CBMS '10)*, pp. 273–278, Perth, Australia, October 2010.

[88] S. Zhang, J. Huang, D. Metaxas, W. Wang, and X. Huang, "Discriminative sparse representations for cervigram image segmentation," in *Proceedings of the 7th IEEE International Symposium on Biomedical Imaging: from Nano to Macro (ISBI '10)*, pp. 133–136, Rotterdam, The Netherlands, April 2010.

[89] V. van Raad, "Design of Gabor wavelets for analysis of texture features in cervical images," in *Proceedings of the 25th Annual International Conference of the IEEE*, vol. 801, pp. 806–809, Engineering in Medicine and Biology Society, 2003.

[90] W. Li and A. Poirson, *Detection and Characterization of Abnormal Vascular Patterns in Automated Cervical Image Analysis*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2006.

[91] J. D. García-Arteaga and J. Kybic, *Geometric and Information Constraints for Automatic Landmark Selection in Colposcopy Sequences*, STI Medical Systems, S.M. Systems, Honolulu, Hawaii, USA, 2007.

[92] K. Krishnaveni, S. Allwin, S. P. K. Kenny, and G. Mariappan, "Analysis for textural features in nuclei of cervical cyto images," in *Proceedings of the IEEE International Conference on Computational Intelligence and Computing Research (ICCIC '10)*, pp. 943–947, Coimbatore, India, December 2010.

[93] A. Alush, H. Greenspan, and J. Goldberger, "Lesion detection and segmentation in uterine cervix images using an ARC-level mRF," in *Proceedings of the IEEE International Symposium on Biomedical Imaging: from Nano to Macro (ISBI '09)*, pp. 474–477, Boston, Mass, USA, July 2009.

[94] A. Alush, H. Greenspan, and J. Goldberger, "Automated and interactive lesion detection and segmentation in uterine cervix images," *IEEE Transactions on Medical Imaging*, vol. 29, no. 2, pp. 488–501, 2010.

[95] X. Huang, W. Wang, Z. Xue, S. Antani, L. R. Long, and J. Jeronimo, "Tissue classification using cluster features for lesion detection in digital cervigrams," in *Medical Imaging 2008: Image Processing*, Proceedings of SPIE, no. 6914, pp. 69141Z–69141Z, San Diego, Calif, USA, 2008.

[96] W. Li, S. Venkataraman, U. Gustafsson, J. C. Oyama, D. G. Ferris, and R. W. Lieberman, "Using acetowhite opacity

index for detecting cervical intraepithelial neoplasia," *Journal of Biomedical Optics*, vol. 14, no. 1, Article ID 014020, 2009.

[97] S. Y. Park, D. Sargent, R. Lieberman, and U. Gustafsson, "Domain-specific image analysis for cervical neoplasia detection based on conditional random fields," *IEEE Transactions on Medical Imaging*, vol. 30, no. 3, pp. 867–878, 2011.

[98] A. Malhi and R. X. Gao, "PCA-based feature selection scheme for machine defect classification," *IEEE Transactions on Instrumentation and Measurement*, vol. 53, no. 6, pp. 1517–1525, 2004.

[99] A. Yildiz, M. Akin, and M. Poyraz, "An expert system for automated recognition of patients with obstructive sleep apnea using electrocardiogram recordings," *Expert Systems with Applications*, vol. 38, no. 10, pp. 12880–12890, 2011.

[100] J. C. B. Melo, G. D. C. Cavalcanti, and K. S. Guimarães, "PCA feature extraction for protein structure prediction," in *Proceedings of the International Joint Conference on Neural Networks*, pp. 2952–2957, Portland, Ore, USA, July 2003.

[101] K. Polat, S. Güneş, and A. Arslan, "A cascade learning system for classification of diabetes disease: generalized discriminant analysis and least square support vector machine," *Expert Systems with Applications*, vol. 34, no. 1, pp. 482–487, 2008.

[102] E. Dogantekin, A. Dogantekin, and D. Avci, "Automatic hepatitis diagnosis system based on linear discriminant analysis and adaptive network based on fuzzy inference system," *Expert Systems with Applications*, vol. 36, no. 8, pp. 11282–11286, 2009.

[103] C.-L. Huang and C.-J. Wang, "A GA-based feature selection and parameters optimizationfor support vector machines," *Expert Systems with Applications*, vol. 31, no. 2, pp. 231–240, 2006.

[104] C.-L. Huang and J.-F. Dun, "A distributed PSO-SVM hybrid system with feature selection and parameter optimization," *Applied Soft Computing Journal*, vol. 8, no. 4, pp. 1381–1391, 2008.

[105] L. A. Reisner, A. Cao, and A. K. Pandya, "An integrated software system for processing, analyzing, and classifying raman spectra," *Chemometrics and Intelligent Laboratory Systems*, vol. 105, no. 1, pp. 83–90, 2011.

[106] G. M. Palmer, C. Zhu, T. M. Breslin, F. Xu, K. W. Gilchrist, and N. Ramanujam, "Comparison of multiexcitation fluorescence and diffuse eeflectance spectroscopy for the diagnosis of breast cancer," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 11, pp. 1233–1242, 2003.

[107] S. J. Baek, A. Park, J. Y. Kim, S. Y. Na, Y. Won, and J. Choo, "Detection of basal cell carcinoma by automatic classification of confocal raman spectra," in *Computational Intelligence and Bioinformatics*, vol. 4115 of *Lecture Notes in Computer Science*, pp. 402–411, Springer, Berlin, Germany, 2006.

[108] F. M. Lyng, E. Ó. Faoláin, J. Conroy et al., "Vibrational spectroscopy for cervical cancer pathology, from biochemical analysis to diagnostic tool," *Experimental and Molecular Pathology*, vol. 82, no. 2, pp. 121–129, 2007.

[109] C.-D. Huang, C.-T. Lin, and N. R. Pal, "Hierarchical learning architecture with automatic feature selection for multiclass protein fold classification," *IEEE Transactions on Nanobioscience*, vol. 2, no. 4, pp. 221–232, 2003.

[110] W. Lin, X. Yuan, P. Yuen et al., "Classification of in vivo autofluorescence spectra using support vector machines," *Journal of Biomedical Optics*, vol. 9, no. 1, pp. 180–186, 2004.

[111] V.-E. Neagoe, A.-C. Mugioiu, and I.-A. Stanculescu, "Face recognition using PCA versus ICA versus LDA cascaded with the neural classifier of concurrent self-organizing maps," in *Proceedings of the 8th International Conference on Communications (COMM '10)*, pp. 225–228, Bucharest, Romania, June 2010.

[112] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces versus fisherfaces: recognition using class specific linear projection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711–720, 1997.

[113] P. R. T. Jess, D. D. W. Smith, M. Mazilu, K. Dholakia, A. C. Riches, and C. S. Herrington, "Early detection of cervical neoplasia by raman spectroscopy," *International Journal of Cancer*, vol. 121, no. 12, pp. 2723–2728, 2007.

[114] J. Wagner, J. Kim, and E. André, "From physiological signals to emotions: implementing and comparing selected methods for feature extraction and classification," in *Proceedings of the IEEE International Conference on Multimedia and Expo, ICME 2005*, pp. 940–943, Amsterdam, The Netherlands, July 2005.

[115] G. Karypis and E. Han, "Concept indexing: a fast dimensionality reduction algorithm with applications to document retrieval and categorization," DTIC Document, U.o. Minnesota, Minneapolis, Minn, USA, 2000.

[116] E. Njoroge, S. R. Alty, M. R. Gani, and M. Alkatib, "Classification of cervical cancer cells using FTIR data," in *Proceedings of the 28th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '06)*, pp. 5338–5341, New York, NY, USA, September 2006.

[117] V. Margariti, M. Zervakis, and C. Balas, "Wavelet and physical parametric analysis of the acetowhitening optical effect: comparative evaluation of performances in non-invasive diagnosis of cervical neoplasia," in *Proceedings of the 10th International Conference on Information Technology and Applications in Biomedicine: Emerging Technologies for Patient Specific Healthcare (ITAB '10)*, Corfu, Greece, November 2010.

[118] Y. N. Mirabal, S. K. Chang, E. N. Atkinson, A. Malpica, M. Follen, and R. Richards-Kortum, "Reflectance spectroscopy for in vivo detection of cervical precancer," *Journal of Biomedical Optics*, vol. 7, no. 4, pp. 587–594, 2002.

[119] J. Mo, W. Zheng, J. J. H. Low, J. Ng, A. Ilancheran, and Z. Huang, "High wavenumber raman spectroscopy for in vivo detection of cervical dysplasia," *Analytical Chemistry*, vol. 81, no. 21, pp. 8908–8915, 2009.

[120] R. F. Walker, P. Jackway, B. Lovell, and I. D. Longstaff, "Classification of cervical cell nuclei using morphological segmentation and textural feature extraction," in *Proceedings of the 2nd Australian and New Zealand Conference on Intelligent Information Systems*, pp. 297–301, December 1994.

[121] S. Y. Park, M. Follen, A. Milbourne et al., "Automated image analysis of digital colposcopy for the detection of cervical neoplasia," *Journal of Biomedical Optics*, vol. 13, no. 1, Article ID 014029, 2008.

[122] D. Westreich, J. Lessler, and M. J. Funk, "Propensity score estimation: neural networks, support vector machines, decision trees (CART), and meta-classifiers as alternatives to logistic regression," *Journal of Clinical Epidemiology*, vol. 63, no. 8, pp. 826–833, 2010.

[123] J.-P. Zhang, Z.-W. Li, and J. Yang, "A parallel SVM training algorithm on large-scale classification problems," in *Proceedings of the International Conference on Machine Learning and Cybernetics (ICMLC '05)*, pp. 1637–1641, Guangzhou, China, August 2005.

[124] H. Drucker, D. Wu, and V. N. Vapnik, "Support vector machines for spam categorization," *IEEE Transactions on Neural Networks*, vol. 10, no. 5, pp. 1048–1054, 1999.

[125] C. Sivapragasam and N. Muttil, "Discharge rating curve extension—a new approach," *Water Resources Management*, vol. 19, no. 5, pp. 505–520, 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 624371, 16 pages
http://dx.doi.org/10.1155/2014/624371

Research Article

Mining 3D Patterns from Gene Expression Temporal Data: A New Tricluster Evaluation Measure

David Gutiérrez-Avilés and Cristina Rubio-Escudero

Department of Computer Science, University of Seville, Avenida Reina Mercedes s/n, 41012 Seville, Spain

Correspondence should be addressed to David Gutiérrez-Avilés; davgutavi@alum.us.es

Received 28 December 2013; Accepted 26 February 2014; Published 31 March 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

Microarrays have revolutionized biotechnological research. The analysis of new data generated represents a computational challenge due to the characteristics of these data. Clustering techniques are applied to create groups of genes that exhibit a similar behavior. Biclustering emerges as a valuable tool for microarray data analysis since it relaxes the constraints for grouping, allowing genes to be evaluated only under a subset of the conditions. However, if a third dimension appears in the data, triclustering is the appropriate tool for the analysis. This occurs in longitudinal experiments in which the genes are evaluated under conditions at several time points. All clustering, biclustering, and triclustering techniques guide their search for solutions by a measure that evaluates the quality of clusters. We present an evaluation measure for triclusters called Mean Square Residue 3D. This measure is based on the classic biclustering measure Mean Square Residue. Mean Square Residue 3D has been applied to both synthetic and real data and it has proved to be capable of extracting groups of genes with homogeneous patterns in subsets of conditions and times, and these groups have shown a high correlation level and they are also related to their functional annotations extracted from the Gene Ontology project.

1. Introduction

The use of high throughput processing techniques has revolutionized the technological research and has exponentially increased the amount of data available [1]. Particularly, microarrays have revolutionized biological research by their ability to monitor changes in RNA concentration in thousands of genes simultaneously [2].

A common practice when analyzing gene expression data is to apply clustering techniques, creating groups of genes that exhibit similar expression patterns [3]. These clusters are interesting because it is considered that genes with similar behavior patterns can be involved in similar regulatory processes [4]. Although in theory there is a big step from correlation to functional similarity of genes, several articles indicate that this relation exists [5].

Traditional clustering algorithms work on the whole space of data dimensions examining each gene in the dataset under all conditions tested. However, the activity of genes could only appear under a particular set of experimental conditions, exhibiting local patterns. Discovering these local patterns can be the key to discover gene pathways, which could be hard to discover in other ways. For this reason, the paradigm of clustering techniques must change to methods that allow local pattern discovery in gene expression data [6].

Biclustering [7] addresses this problem by relaxing the conditions and by allowing assessment only under a subset of the conditions of the experiment, and it has proved to be successful in finding gene patterns [8]. However, if the time condition is added to the dataset clustering, and biclustering result insufficient. There is a lot of interest in temporal experiments because they allow an in-depth analysis of molecular processes in which the time evolution is important, for example cell cycles, development at the molecular level or evolution of diseases [9]. In this sense, triclustering appears as a valuable tool since it allows for the assessment of genes under a subset to the conditions of the experiment and under a subset of times.

All clustering, biclustering, and triclustering techniques guide their search for solutions by a measure that evaluates

the quality of clusters [10]. In this work we propose an evaluation measure for triclusters called Mean Square Residue 3D (MSR_{3D}). This measure is based on a classic biclustering measure presented by Cheng and Church in [11] called Mean Square Residue (MSR). MSR measures the homogeneity of a bicluster in the relation of each value in the bicluster with the average value for all genes in the bicluster, average of all conditions, and average of all genes and conditions in the bicluster. A perfect score would be zero, which represents a constant bicluster of elements of a single value.

Our proposal, MSR_{3D}, is an adaptation of MSR to the three-dimensional space, so that a third factor, in this case time, can be taken into account. MSR_{3D} measures the homogeneity of a tricluster in the relation of each value of the tricluster, with the average of all genes, average of all conditions, average of all times, average of all genes and conditions, average of all genes and times, average of all conditions and times, and average of all genes, conditions, and times in the tricluster. As for MSR, a perfect score would be zero, which represents a constant tricluster of elements of a single value.

MSR_{3D} has been applied as an evaluation measure along with the TriGen (*Tri*clustering-*Gene*tic based) algorithm presented in [12]. TriGen is an algorithm based on evolutionary heuristic, genetic algorithms. Many heuristic approaches have been proposed both for biclustering and triclustering algorithms [13, 14], due to the NP hard nature of the problem [15].

We show the results obtained from applying the TriGen algorithm along with the MSR_{3D} measure to a synthetic dataset and four real experiments datasets: the yeast cell cycle regulated genes [16], mouse degeneration of retinal cells [17], mouse ectopic bHLH transcription factor expression Mesogenin1 effect on embryoid bodies [17], and human Transcription factor oncogene OTX2 silencing effect on D425 medulloblastoma cell line [17].

The results have been validated by analyzing the correlation among the genes, conditions, and times in each tricluster using two different correlation measures: Pearson and Filon [18] and Spearman [19]. Besides this, we have provided functional annotations for the genes extracted from the Gene Ontology project [20].

The rest of the paper is structured as follows. A review of the latest related works can be found in Section 2. Section 3 describes the methodology of the MSR and MSR_{3D} measures as well as a brief description of the TriGen algorithm. In Section 4 we show the results of applying TriGen to the synthetic and real datasets. Section 5 shows the conclusions.

2. State of the Art

This section is to provide a general overview of recent works in the field of gene expression temporal data. In particular, for those works related to the application of triclustering, we focus on the measures applied to evaluate the triclusters.

In 2005, Zhao and Zaki [21] introduced the triCluster algorithm to extract patterns in 3D gene expression data. They presented a measure to assess triclusters' quality based on the symmetry property. This allows a very efficient cluster mining since clusters are searched over the dimensions with the least cardinality. The triclusters have to fulfill some requirements such as being maximal; that is, no tricluster in the set of solutions is totally included in another tricluster in the set of solutions; the ratio of every pair of columns in the tricluster is delimited by a given ϵ; the maximum volume of the tricluster is determined by the relation among δ^x, δ^y, and δ^z for gene, condition, and time dimensions,respectively; and the minimum volume for the tricluster is also controlled. An extended and generalized version of this proposal, g-triCluster, was published one year later [22]. The authors claimed that the symmetry property is not suitable for all patterns present in biological data and propose the Spearman rank correlation [19] as a more appropriate tricluster evaluation measure.

An evolutionary computation proposal was made in [23]. The fitness function defined is a multiobjective measure which tries to optimize three conflicting objectives: clusters size, homogeneity, and gene-dimension variance of the 3D cluster.

LagMiner was introduced in [24] to find time-lagged 3D clusters, which allows in turn finding regulatory relationships among genes. It is based on a novel 3D cluster model called S^2D^3 Cluster. They evaluated their triclusters on homogeneity, regulation, minimum gene number, sample subspace size, and time periods length.

Wang et al. [25] proposed a new algorithm called ts-cluster basing their definition for coherent triclusters also on finding regulatory relationships among genes. For that purpose, time shifting is also considered among time points in the evaluated triclusters.

A new strategy to mine 3D clusters in real-valued data was introduced in [26]. The authors defined the Correlated 3D Subspace Clusters (CSCs) where the values in each cluster must have high cooccurrences and those cooccurrences are not by chance. They measure the clusters based on the correlation information measure, which takes into account both prerequisites. In particular, the authors were concerned about discovering subspaces with a significant number of items, one of the main problems typically found in tricluster-based approaches. At the same conference, another approach was presented focusing on the concept of Low-Variance 3-Cluster [27], which obeys the constraint of a low-variance distribution of cell values.

The work in [28] was focused on finding Temporal Dependency Association Rules, which relate patterns of behaviour among genes. The rules obtained are to represent regulated relations among genes.

Finally, a brief survey on triclustering applied to gene expression time series was published in 2011 [29].

3. Methodology

In this section we first describe what is triclustering in relation to biclustering, second we show the fundamentals of our proposal, the two dimensions MSR measure proposed by Cheng and Church [11] in order to assess the quality of

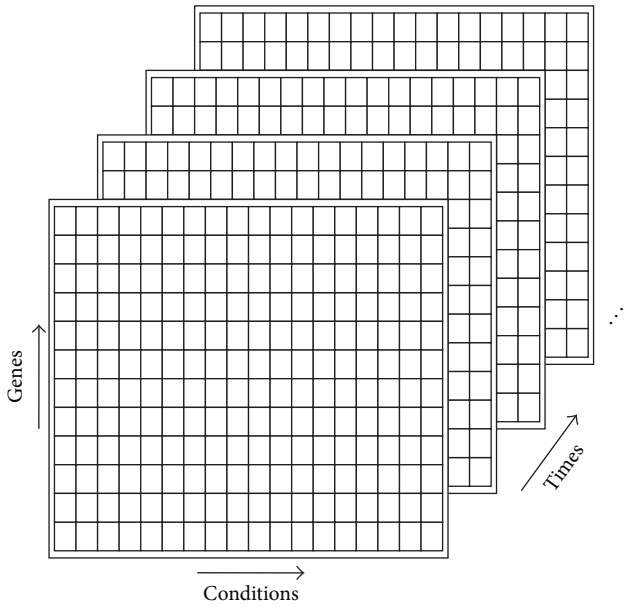


Figure 1: Tricluster representation.

biclusters grouping gene and conditions, and third we make a detailed description of our proposal, the three dimensions MSR measure (MSR_{3D}) to assess the quality of triclusters which group gene, conditions, and the time dimension. Finally, we describe TriGen and the genetic algorithm where the (MSR_{3D}) measure has been integrated to be tested.

3.1. Triclustering. Given a dataset containing information from gene expression data organized in rows/columns (genes as rows and conditions as columns), biclustering finds subgroups of genes and conditions where the genes exhibit highly correlated patterns of behavior for every condition [30].

A bicluster BC can be defined as a subset from a dataset D which contains information related to the behavior of some genes G_D under certain conditions C_D. The tricluster TC is formally defined as $TC = G \times C$ where $G \subseteq G_D$ and $C \subseteq C_D$.

Triclustering appears as an evolution of biclustering due to its capacity to mine gene expression datasets involving time as a third dimension and to find subgroups of genes, conditions, and times which exhibit highly correlated patterns of expression [12]. Figure 1 shows the structure of a tricluster, with genes as rows, conditions as columns, and time as depth.

A tricluster TC is as a subset from a dataset D which contains information related to the behavior of some genes G_D under conditions C_D at times T_D. The tricluster TC is formally defined as $TC = G \times C \times T$ where $G \subseteq G_D$, $C \subseteq C_D$, and $T \subseteq T_D$.

3.2. Two-Dimension MSR. The Mean Squared Residue (MSR) was introduced by Cheng and Church in [11]. This measure was proposed to assess the quality of biclusters extracted from gene expression data based on biclusters' homogeneity. The formal definition can be seen in

$$\mathrm{MSR}\,(\mathrm{BC}) = \frac{\sum_{g\epsilon G, c\epsilon C} r_{gc}^2}{\#G * \#C}, \tag{1}$$

where r_{gc} can be defined as

$$r_{gc} = BC_v\,(g,c) - M_G\,(c) - M_C\,(g) - M_{GC}. \tag{2}$$

Each of the terms of (1) and (2) are defined as follows:

(i) BC: bicluster being evaluated,

(ii) G: subset of genes of BC,

(iii) C: subset of conditions of BC,

(iv) $\#G$: number of genes in BC,

(v) $\#C$: number of conditions in BC,

(vi) $\mathrm{BC}_v(g,c)$: expression level of a gene g under condition c in BC,

(vii) $M_G(c)$: mean of the values of a condition c under all genes in BC,

(viii) $M_C(g)$: mean of the values of a gene g under all conditions in BC,

(ix) M_{GC}: mean value of all values in BC.

A graphical representation of the values involved in (2) can be seen in Figure 2. We can say that MSR measures the homogeneity for a given bicluster based on the difference of each individual gene expression $\mathrm{BC}_{v(i,j)}$ (see Figure 2(a)) with the average values of genes $M_{G(j)}$ (see Figure 2(b)), conditions $M_{C(i)}$ (see Figure 2(c)), and genes and conditions M_{GC} (see Figure 2(d)). The closer the value of MSR is to zero, the more homogeneous the bicluster is. This interpretation is the basis for the extension to three-dimension measure MSR_{3D} presented in the next section.

3.3. Three Dimensions MSR. Our proposal is an adaptation to three dimensions of MSR that measures the homogeneity of triclusters which contain subgroups of genes, conditions, and time points. We call this measure MSR_{3D}. The formal definition can be seen in

$$\mathrm{MSR}_{3D}\,(\mathrm{TC}) = \frac{\sum_{g\epsilon G, c\epsilon C, t\epsilon T} r_{gct}^2}{\#G * \#C * \#T}, \tag{3}$$

where r_{gct} can be defined as

$$\begin{aligned} r_{gct} = {} & \mathrm{TC}_v\,(g,c,t) + M_{CT}\,(g) + M_{GT}\,(c) + M_{GC}\,(t) \\ & - M_G\,(c,t) - M_C\,(g,t) - M_T\,(g,c) - M_{GCT}. \end{aligned} \tag{4}$$

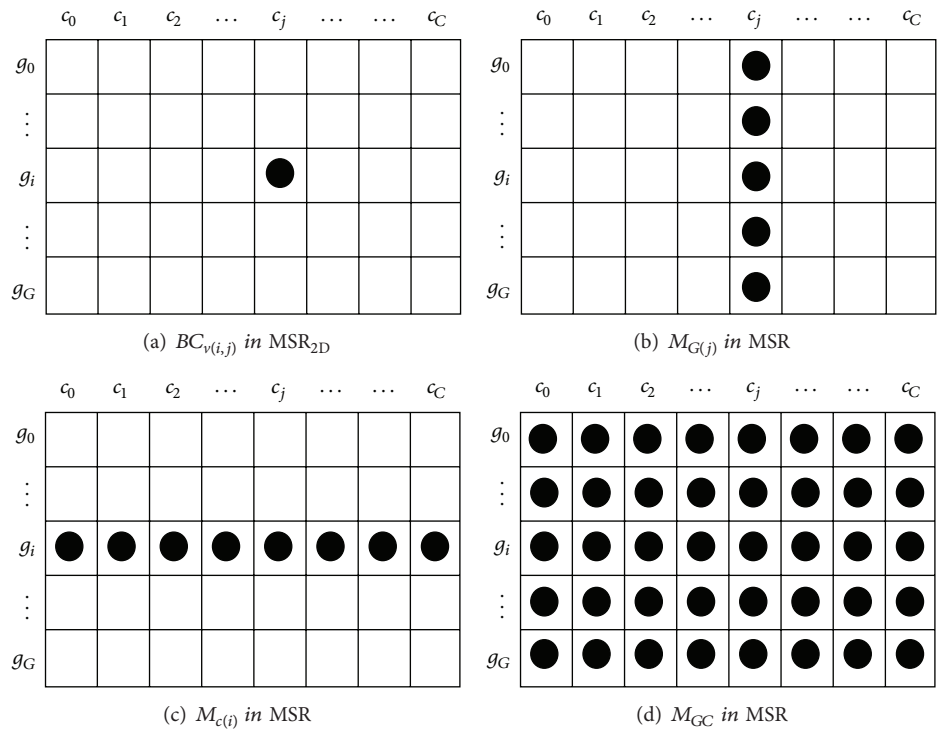
Each of the members of (3) and (4) is defined as follows:

(i) TC: tricluster being evaluated,

(ii) G: subset of genes from TC,

(iii) C: subset of conditions from TC,

(iv) T: subset of times from TC,



(a) $BC_{v(i,j)}$ *in* MSR_{2D} (b) $M_{G(j)}$ *in* MSR

(c) $M_{c(i)}$ *in* MSR (d) M_{GC} *in* MSR

Figure 2: MSR members.

(v) #G: number of genes in TC,

(vi) #C: number of conditions in TC,

(vii) #T: number of times in TC,

(viii) $\text{TC}_v(g, c, t)$: expression level of gene g under condition c at time t in TC,

(ix) $M_{CT}(g)$: mean of all conditions at all times for a gene g in TC,

(x) $M_{GT}(c)$: mean of all genes at all times for a condition c in TC,

(xi) $M_{GC}(t)$: mean of all genes under all conditions at time t in TC,

(xii) $M_G(c, t)$: mean of the values of a condition c and a time t under all genes in TC,

(xiii) $M_C(g, t)$: mean of the values of a gene g and a time t under all conditions in TC,

(xiv) $M_T(g, c)$: mean of the values of a gene g and a condition c under all times in TC,

(xv) M_{GCT}: mean value of all values in TC.

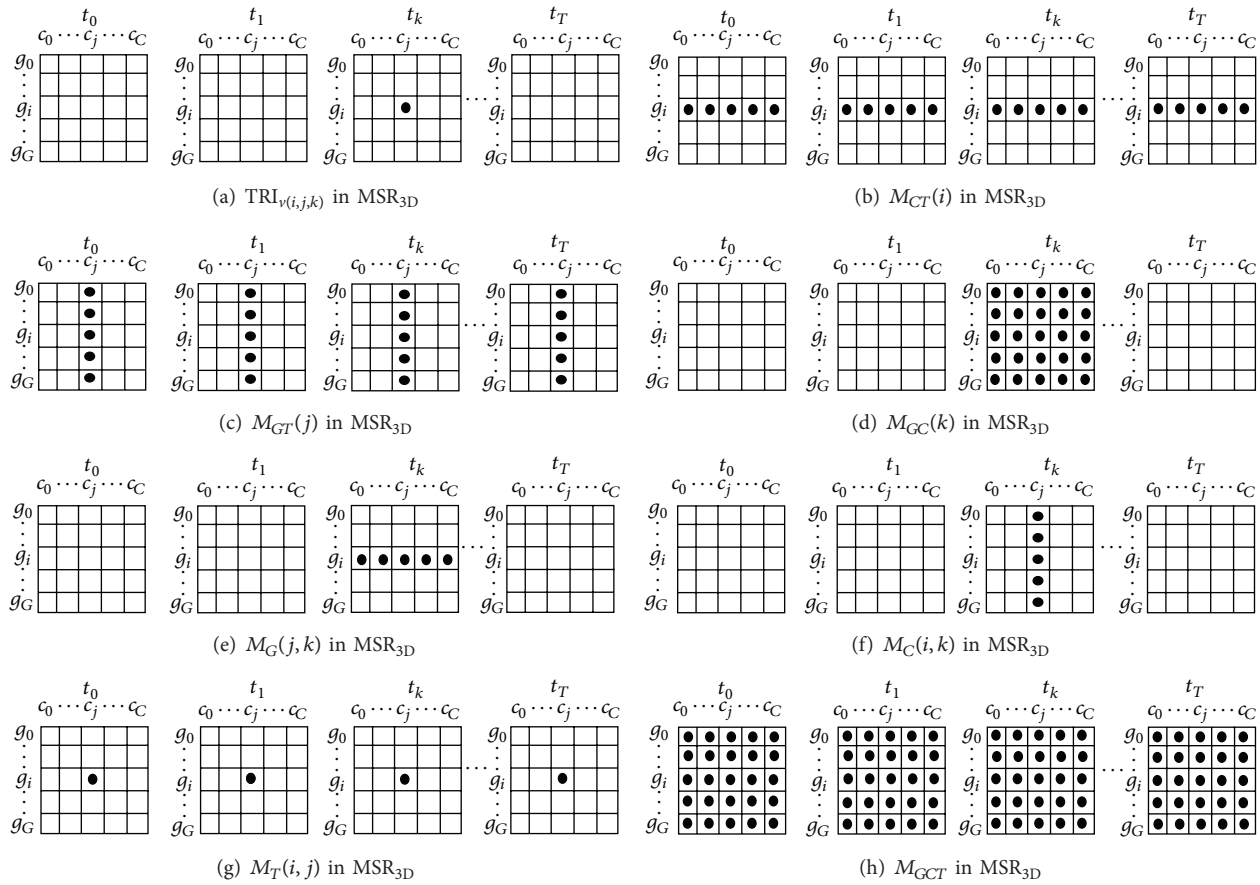
A graphical representation of the values involved in (4) can be seen in Figure 3. We can say that MSR_{3D} measures the homogeneity for a given tricluster based on the difference of each individual gene expression $\text{TC}_v(i, j, k)$ (see Figure 3(a)), the mean of all conditions at all times for a gene g $M_{CT}(g)$ (see Figure 3(b)), the mean of all genes at all times for a condition c $M_{GT}(c)$ (see Figure 3(c)), the mean of all genes under all conditions at time t $M_{GC}(t)$ (see Figure 3(d)) with the mean of a condition c and a time t under all genes $M_G(c, t)$ (see Figure 3(e)), the mean of a gene g and a time t under all conditions $M_C(g, t)$ (see Figure 3(f)), the mean of a gene g and a condition c under all times $M_T(g, c)$ (see Figure 3(g)), and the mean value of all values in TC M_{GCT} (see Figure 3(h)). The closer the value of MSR_{3D} is to zero, the more homogeneous the tricluster is. MSR_{3D} is capable of finding negatively correlated genes due to its formulation.

3.4. TriGen Algorithm. To test the effectiveness of MSR_{3D} we have included it as part of the TriGen (Triclustering-Genetic based) algorithm [12]. TriGen extracts triclusters from gene expression datasets where the time is also a component taken into account in the experiment. TriGen applies a bioinspired paradigm of an evolutionary heuristic, genetic algorithms, which mimics the process of natural selection by creating an initial population of individuals representing solutions which are crossed and mutated for a number of generations and the best individuals in the populations are finally selected. MSR_{3D} has been applied along with TriGen as a fitness function to assess the quality of the triclusters or solutions in the population.

The flowchart of the TriGen algorithm can be seen in Figure 4. In these subsections we are going to present the principal aspects of the algorithm including inputs, outputs, representation of individuals, and genetic operators.

3.4.1. TriGen's Input. The *TriGen* algorithm takes two inputs:

(i) D: a dataset containing the gene expression values from an experiment containing genes G, experimental conditions C, and times T. Therefore, each cell



(a) $\mathrm{TRI}_{v(i,j,k)}$ in MSR_{3D} (b) $M_{CT}(i)$ in MSR_{3D}

(c) $M_{GT}(j)$ in MSR_{3D} (d) $M_{GC}(k)$ in MSR_{3D}

(e) $M_G(j,k)$ in MSR_{3D} (f) $M_C(i,k)$ in MSR_{3D}

(g) $M_T(i,j)$ in MSR_{3D} (h) M_{GCT} in MSR_{3D}

Figure 3: MSR_{3D} structural members.

$[i, j, k]$ from D where $i \in G$, $j \in C$, and $k \in T$ represents the expression level of the gene i under the experimental condition j at time k;

(ii) P: set of parameters to execute the algorithm as described in Table 1. These parameters control the number of solutions or triclusters to find (N), the number of generations to execute (G), the number of individuals in the population (I), and the randomness factor which are generated within the initial population (*Ale*) as well as weights for the selection and mutation operators (sel y mut), weights to control the size of the triclusters (w_g, w_c, w_t), and weights to control the overlap among solutions (wo_g, wo_c, wo_t).

3.4.2. TriGen's Output. The TriGen algorithm's output will be a set of N triclusters. Each tricluster is composed of a subset of genes G_g, conditions C_c, and times T_t from the input dataset D, with the best scores when evaluated under the MSR_{3D} measure.

3.4.3. Codification of Individuals. Each individual in the evolutionary process of the *TriGen* algorithm represents a tricluster, that is, a subset of genes, experimental conditions, and time points. All genetic operators are applied to each individual in the population, in each of these three subsets.

Table 1: TriGen algorithm parameters.

Parameter	Description
N	Number of triclusters extracted
G	Number of generations
I	Number of individuals in the population
Ale	Randomness rate
Sel	Selection rate
Mut	Mutation probability
w_g	Weight for the number of genes
w_c	Weight for the number of conditions
w_t	Weight for the number of times
wo_g	Weight for the overlap among genes
wo_c	Weight for the overlap among conditions
wo_t	Weight for the overlap among times

The genetic material is structured as follows. An individual, as mentioned above, is composed of three sequences of structures: one for the sequence of genes G from the input dataset D, one for the sequence of conditions C, and one sequence of time points T. These sequences are set up based on the input dataset; that is,

$$G = \left\langle g_{i_1}, g_{i_2}, \ldots, g_{i_B} \right\rangle, \tag{5}$$

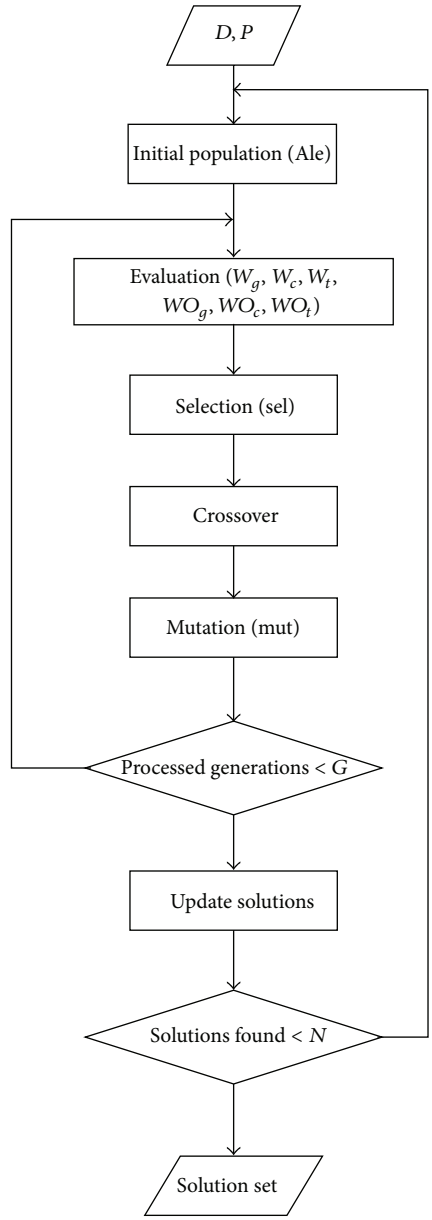


Figure 4: Flowchart for the TriGen algorithm.

where B is the number of genes listed in the input dataset, $i_j < i_{j+1}$ for all genes, and $1 < i_j < B$.

Analogously

$$C = \left\langle c_{i_1}, c_{i_2}, \ldots, c_{i_L} \right\rangle, \quad (6)$$

where L is the number of conditions listed in the input dataset, $i_j < i_{j+1}$ for all conditions, and $1 < i_j < L$.

Finally, T represents different time stamps or values of pairs gene condition at different times:

$$T = \left\langle t_{i_1}, t_{i_2}, \ldots, t_{i_M} \right\rangle, \quad (7)$$

where M is the number of samples measured over time and $t_{i_1} < t_{i_2} < \cdots < t_{i_M}$.

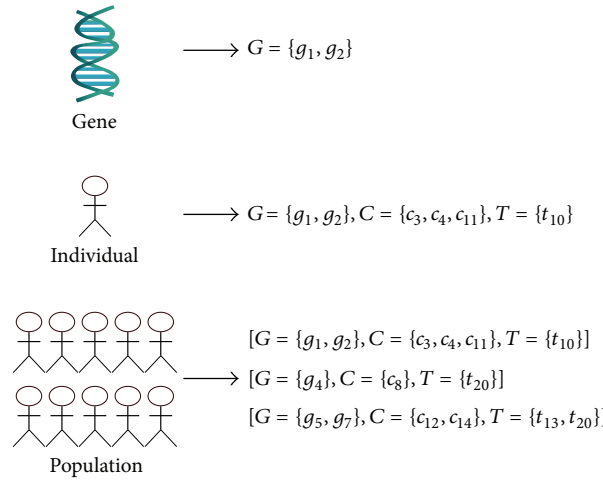


Figure 5: Genetic algorithm codification.

The algorithm's population is made up of several individuals, as depicted in Figure 5, where the individual codification has been represented.

3.4.4. Initial Population. The initial population is generated attending to the *Ale* randomness parameter. An *Ale* percent of individuals are created at random by two methods: half of the individuals are purely randomly generated; this is a random subset of genes G_g, conditions C_c, and times T_t chosen from D and the other half is also randomly created but controlling that the values for the genes G_g are contiguous; the values for the conditions C_c are contiguous and the times T_t are contiguous as well. The rest of the individuals are created at random but taking into account the previously created individuals to control overlapping of solutions.

3.4.5. Fitness Function. The proposed measure MSR_{3D} has been applied as part of the fitness function to evaluate the homogeneity of the triclusters in the population. MSR_{3D} has been combined with two other factors which measure the size of the triclusters and their overlap with previously found solutions.

Controlling the size of each of the dimensions of the triclusters might be a very important task since gene expression datasets are unbalanced on the three dimensions, with the number of genes counting in thousands and the number of conditions and times counting in tens. Therefore, the weights for the number of genes w_g, of conditions w_c, and times w_t control that the dimensions of the triclusters are balanced (e.g., if we increase w_g, the algorithm considers that solutions with a high number of genes are better than those with low number of genes).

We also control the overlap among found solutions with the weights wo_g, wo_c, and wo_t for the overlap among genes, conditions, and times, respectively, (e.g., if we increase wo_g, the algorithm considers that solutions with low level of overlap with the genes in previously found solutions are better than those with a high level of overlap).

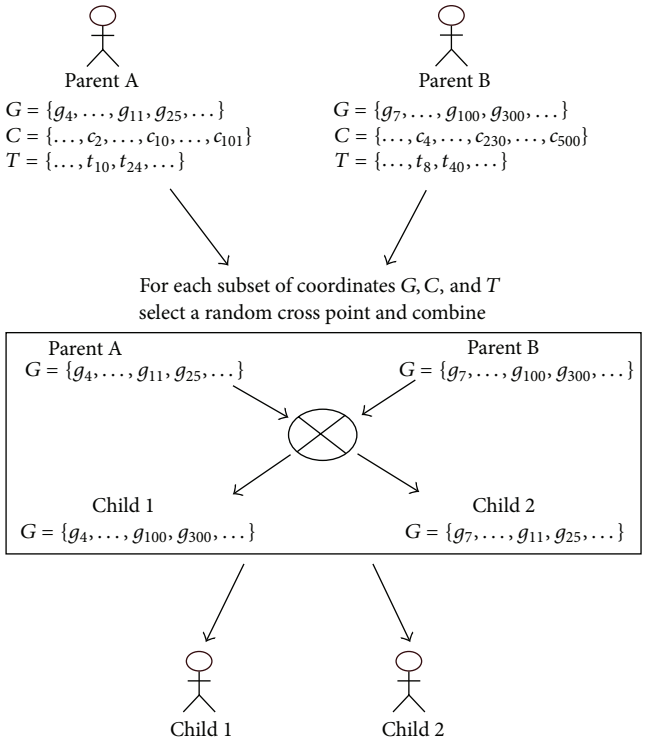


Figure 6: Representation of the crossover operator.

Therefore, the fitness function can be formulated as seen in

$$\text{FF (TC)} = \text{MSR}_{3\text{D}} - \text{size control} - \text{overlap control}. \quad (8)$$

3.4.6. Selection Operator. This operator is implemented following the roulette wheel selection method [31]. The fitness level is used to associate a probability of selection with each individual of the population. This emulates the behavior of a roulette wheel in a casino. Usually a proportion of the wheel is assigned to each of the possible selections based on their fitness value. Then a random selection is made similar to how the roulette wheel is rotated. While candidates with a higher fitness will be less likely to be eliminated, there is still a chance that they are eliminated. There is a chance that some weaker solutions may survive the selection process, which is an advantage, as though a solution may be weak, it may include some component which could prove useful following the recombination process. The *Sel* parameter indicates how many individuals will pass to the next generation undergoing this method. The rest of the individuals up to complete the next population (I – *#Selected individuals*) will be created based on the crossover operator.

3.4.7. Crossover Operator. To complete the next generation, we create new individuals with this operator as follows: two individuals (parents, A and B) are combined to create two new individuals (offspring, *child*1 and *child*2). The parents are randomly chosen. Their genetic material is combined by a random one-point cross in the genes G_g, conditions C_c, and times T_t and mixing the coordinates in both children. We can see this process in Figure 6.

3.4.8. Mutation. An individual can be mutated according to a probability of mutation, Mut. The mutation probability is verified for every individual and if it is satisfied, one out of six possible actions is taken. These actions are as follows: add a new random gene to G_g in TC, add a new condition to C_c in TC, or add a new time to T_t in TC or by removing a random gene, condition, or time. The election of these actions is also random. For the case of addition of a new gene, condition, or time, the operator checks whether the new member is already in the individual or not.

4. Results

We have applied the proposed measure $\text{MSR}_{3\text{D}}$ as part of the TriGen algorithm to analyse several datasets: synthetically generated data, data from experiments with the yeast cell cycle (*Saccharomyces cerevisiae*) obtained from the Stanford University [16], three datasets retrieved from Gene Expression Omnibus [17], and a database repository of high throughput gene expression data. Two datasets are experiments for mouse (*Mus musculus*) [32, 33] and the third one is an experiment for humans (*Homo sapiens*) [34]. All experiments examine the behaviour of genes under conditions at certain times.

To examine the quality of the results in experiments with real datasets, we show for each experiment two types of validity measures: analysis of correlation among the genes, conditions, and times in each tricluster and analysis of genes and gene product annotations for the genes in each tricluster based on the Gene Ontology project [20].

Regarding the correlation analysis, we show a table for each tricluster (in rows) in which we calculate the Pearson and Filon [18] and Spearman [19] correlation coefficient between each combination of condition time and the values series are the expression levels of all genes in the corresponding condition-time combination. For example, for a tricluster with ten genes $\{1, \ldots, 10\}$, three conditions $\{1, 3$ and $5\}$, and two times $\{2$ and $7\}$, we provide Pearson's and Spearman's correlation coefficient for values at the six possible combinations $V_{c=1,t=2}$, $V_{c=1,t=7}$, $V_{c=3,t=2}$, $V_{c=3,t=7}$, $V_{c=5,t=2}$, and $V_{c=5,t=7}$ for each of the ten genes.

In the biological analysis we provide a validation of the triclusters obtained based on the Gene Ontology project (GO) [20]. GO is a major bioinformatic initiative with the aim of standardizing the representation of gene and gene product attributes across species and databases. The project provides an ontology of terms for describing gene product characteristics and gene product annotation data. The ontology covers three domains: cellular component, the parts of a cell or its extracellular environment; molecular function, the elemental activities of a gene product at the molecular level such as binding or catalysis; and biological process, operations, or sets of molecular events with a defined beginning and end, pertinent to the functioning of integrated living units: cells, tissues, organs, and organisms. For legibility reasons, we have

presented for one solution of the experiment a GO analysis table in which we include the most representative terms extracted by the Ontologizer software [35].

We have also provided a graphical representation of the triclusters found. For legibility reasons we show graphs for one tricluster for each of the experiments. Each tricluster is represented through three graphical views in which we can see the pattern of behavior. In the first (sample curves), we show one graph for each time, genes on the x-axis, the expression levels on the y-axis, and the lines of condition as the outline. In the second (time curves), we show for each experimental condition (one graph for each condition) genes on the x-axis, the expression levels in the y-axis, and the time lines as the outline. In the third representation (gene curves), for each experimental condition (one graph for each condition) we show times in the x-axis, the expression levels in the y-axis, and the genes as the outline.

All experiments were executed on a multiprocessor machine with 64 processors Intel Xeon E7-4820 2.00 GHz with 8 GB RAM memory. We have used Java for the TriGen algorithm implementation (and other ad hoc developments) and an R framework to create graphics and get datasets resources from GEO [17].

We now analyse the results obtained in each of the five experiments.

4.1. Synthetic Datasets. Synthetic data has the advantage that the process that generated the data is well known and so one is able to judge the success or failure of the algorithm [36]. Synthetic datasets generation has been widely applied both in microarray related publications [37, 38] and in other general data mining applications [39].

We have used an application designed by ourselves to generate the synthetic data applied in this work. The data generated is a three-dimensional dataset Dsynt_{3D} with 4000 genes (rows), 30 conditions (columns), and 20 times (depth) of random numbers generated by a cryptographic secure standard library Math3 provided by Apache Commons [40] where we insert 10 triclusters $T\text{Cale}_i$, $i \in 1, \ldots, 10$ with 3D patterns of 150 genes (rows), 6 conditions (columns), and 4 times (depth) at random positions within Dsynt_{3D}.

To see the behavior of the MSR_{3D} measure applied along with TriGen and also with the aim of analyzing the effect of the value of the parameters in the solutions, we have made executions varying the number of solutions N in $\{100, 200\}$ and other control parameters as follows.

(i) Number of generations G in $\{50, 100\}$: greater number of generations gives us an increase in genetic recombination of individuals; an excessive increase in G may favour exploitation versus exploration in excess and the algorithm may return solutions which fall into a local minimum.

(ii) Number of individuals I in $\{300, 500\}$: an increase in the number of individuals creates a larger search space for the solutions; an excessive increase can create a scatter search effect and therefore not return good quality solutions.

Table 2: TriGen algorithm synthetic match ratios.

Triduster	Match ratio
TCale_1	91%
TCale_2	91%
TCale_3	90%
TCale_4	96%
TCale_5	95%
TCale_6	95%
TCale_7	95%
TCale_8	95%
TCale_9	95%
TCale_{10}	95%

(iii) Rate of selection Sel in $\{0.5, 0.7, 0.9\}$: a high selection rate creates individuals with low level of genetic recombination, favouring exploitation versus exploration and if the parameter is increased in excess, the algorithm may fall into a local minimum.

(iv) Probability of mutation Mut in $\{0.1, 0.5\}$: the opposite to the rate of selection. A high probability of mutation favours exploration versus exploitation, and if increased in excess you will end up with solutions in many areas of the search space but with low quality levels.

(v) Randomness in the initial population Ale in $\{0.5, 0.9\}$: increasing this parameter involves increasing the level of randomness in the initial population. This has to be combined with the overlap control to make sure that a wide area of the space of solutions is initially covered.

(vi) Weight for the number of genes in the solution w_g in $\{0.0, 0.4\}$, weight for the number of conditions w_c in $\{0.0, 0.1\}$, and weight for the number of times w_t in $\{0.0, 0.1\}$ control the number of items in the solutions; increasing these weights involves favouring solutions with more volume.

(vii) Overlap control weights for genes, wo_g in $\{0.4, 0.7\}$, conditions wo_c in $\{0.0, 0.1\}$, and times wo_t in $\{0.0, 0.1\}$: the increase in these weights leads to little or nonoverlapped solutions; an excessive increase can lead us to lose interesting solutions.

The results obtained are shown in Table 2. We can see the high rate of coverage (90–96%) of the 10 different triclusters TCale_i inserted at random positions in the dataset Dsynt_{3D}.

We can conclude that the MSR_{3D} measure applied along with TriGen algorithm was successful in finding the solution triclusters.

4.2. Yeast Cell Cycle Dataset. We have applied the TriGen algorithm to the yeast (*Saccharomyces cerevisiae*) cell cycle problem [16]. The yeast cell cycle analysis project's goal is to identify all genes whose mRNA levels are regulated by the cell cycle. The resources used are public and available in http://genome-www.stanford.edu/cellcycle/. Here we can find information relative to gene expression values obtained

Table 3: TriGen algorithm control parameters for Yeast Cell Cycle Dataset.

Parameter	Values
N	20
G	200
I	50
Ale	0.3
Sel	0.5
Mut	0.3
w_g	0.7
w_c	0.5
w_t	0.5
wo_g	0.8
wo_c	0.5
wo_t	0.5

from different experiments using microarrays. In particular, we have created a dataset Delu_{3D} from the elutriation experiment with 7744 genes, 13 experimental conditions, and 14 time points. Experimental conditions correspond to different statistical measures of the Cy3 and Cy5 channels while time points represent different moments of taking measures from 0 to 390 minutes.

The parameter configuration used for this experiment is shown in Table 3.

With this configuration we wanted to find solutions with a considerable number of genes ($w_g = 0.7$) because it is the largest dimension on Delu_{3D}. With the overlap control values we seek a compromise between slightly overlapped solutions and not losing interesting triclusters. The rest of the parameters have been set to a default configuration.

To analyse the results, we can see the correlation in Table 4. We see how the correlation levels vary from very low up to almost perfect correlation. This is due to the fact that MSR_{3D} is capable of finding negatively correlated values, and some genes involved in the yeast cell cycle behave in an inversely correlated manner [41, 42] as can be seen in Figure 7(a). Therefore, when calculating the averages of correlations close to one and correlations close to minus one, we get values close to zero. Triclusters TC_8, TC_{15}, and TC_{19} stand out for having Pearson and Spearman correlation values close to one indicating an almost perfect correlation.

We also show a graphical representation of the genes, conditions, and times selected by tricluster TC_9 with 30 genes, 3 conditions, and 9 time points in Figure 7. In Figure 7(a) we see a representation of genes at each condition with a graph for each time. The negative correlation among genes is clearly shown. Figure 7(b) shows the genes at each time with one graph for each condition, and finally in Figure 7(c) we see the times at each gene with a graph for each condition.

In Table 5 we show an analysis of the biological annotations related to the genes selected in our tricluster TC_9.

In this type of studies, P values are relevant below 0.05. We show the ten most significant terms with values ranking in the [0.001970,0.01039] interval. Furthermore, these terms are quiet specific increasing the quality of the tricluster obtained.

Table 4: Correlation results for tricluster Yeast Cell Cycle Dataset.

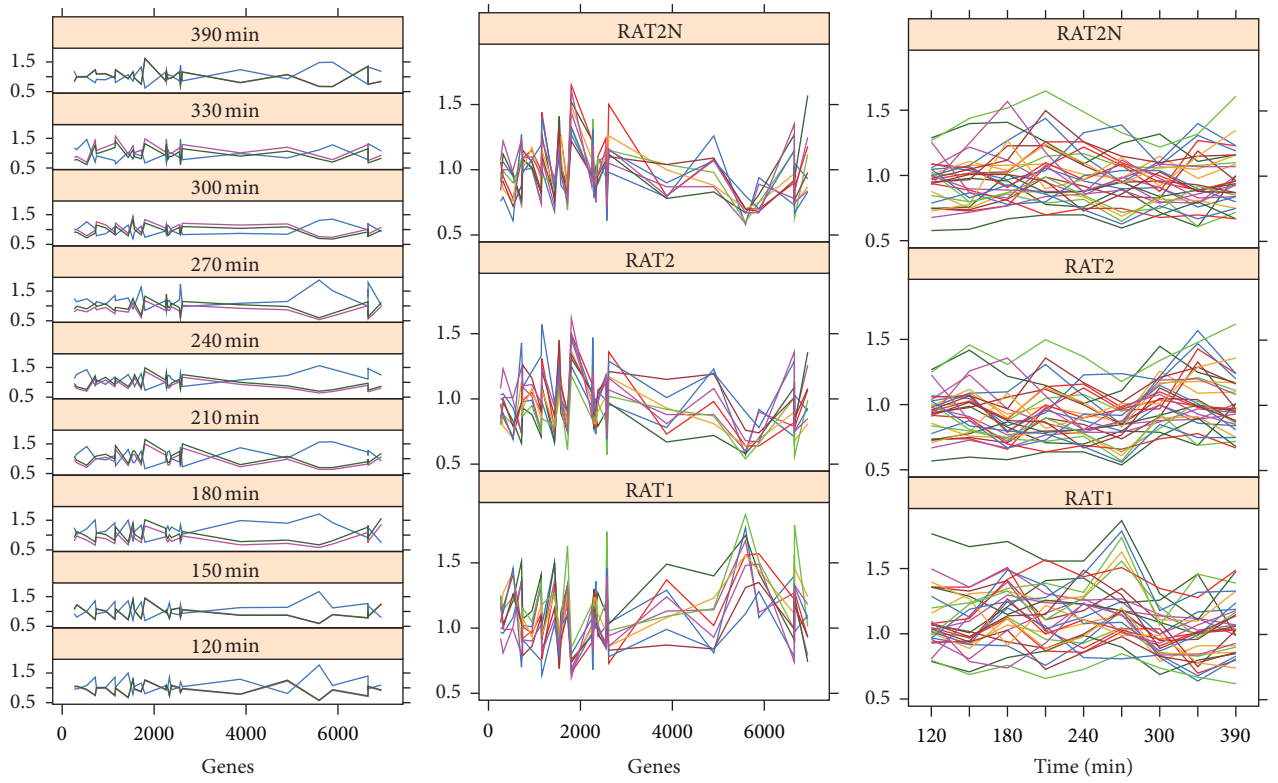
TC_{sol}	Pearson	Spearman
1	−0.02	−0.02
2	0.32	0.28
3	0.17	0.24
4	0.37	0.37
5	0.02	0.02
6	0.05	0.04
7	0.03	0.03
8	0.99	0.98
9	0.04	0.03
10	0.03	0.01
11	0.03	0.03
12	0.02	0.02
13	0.06	0.03
14	0.04	0.04
15	0.99	0.98
16	0.03	0.01
17	0.01	0
18	0.03	0.03
19	0.92	0.9
20	0.03	0.02

Table 5: GO analysis for tricluster TC_9 found in the Yeast Cell Cycle Dataset.

ID	Name	P-value
GO:0071012	Catalytic Step 1 spliceosome	0.001970
GO:0071006	U2-type catalytic Step 1 spliceosome	0.001970
GO:0072521	Purine-containing compound metabolic process	0.004610
GO:0051266	Sirohydrochlorin ferrochelatase activity	0.005208
GO:0004385	Guanylate kinase activity	0.005208
GO:0004747	Ribokinase activity	0.005208
GO:0006014	D-ribose metabolic process	0.005208
GO:0006986	Response to unfolded protein	0. 007288
GO:0046148	Pigment biosynthetic process	0.009738
GO:0070899	Mitochondrial tRNA wobble uridine modification	0.01039

4.3. Mouse GDS4510 Dataset. This dataset was obtained from the GEO [17] with accession code GDS4510 and under the title *rd1 model of retinal degeneration: time course* [32]. In this experiment the degeneration of retinal cells in different individuals of home mouse (*Mus musculus*) is analyzed over 4 days just after birth, specifically on days 2, 4, 6, and 8. Our input dataset $DGDS4510_{3D}$ is composed of 22690 genes, 8 experimental conditions (one for each individual involved in the biological experiment), and 4 time points.

We have executed the TriGen algorithm with the parameters shown in Table 6. We have increased the number of generations and individuals to create a larger search space as the input dataset has a considerable large volume. For the



(a) Sample curves (b) Time curves (c) Gene curves

Figure 7: Graphical representation for tricluster TC_9 found in the Yeast Cell Cycle Dataset.

Table 6: TriGen algorithm control parameters for Mouse GDS4510 Dataset.

Parameter	Values
N	20
G	500
I	300
Ale	0.4
Sel	0.4
Mut	0.2
w_g	0.8
w_c	0.3
w_t	0.2
wo_g	0.8
wo_c	0.5
wo_t	0.5

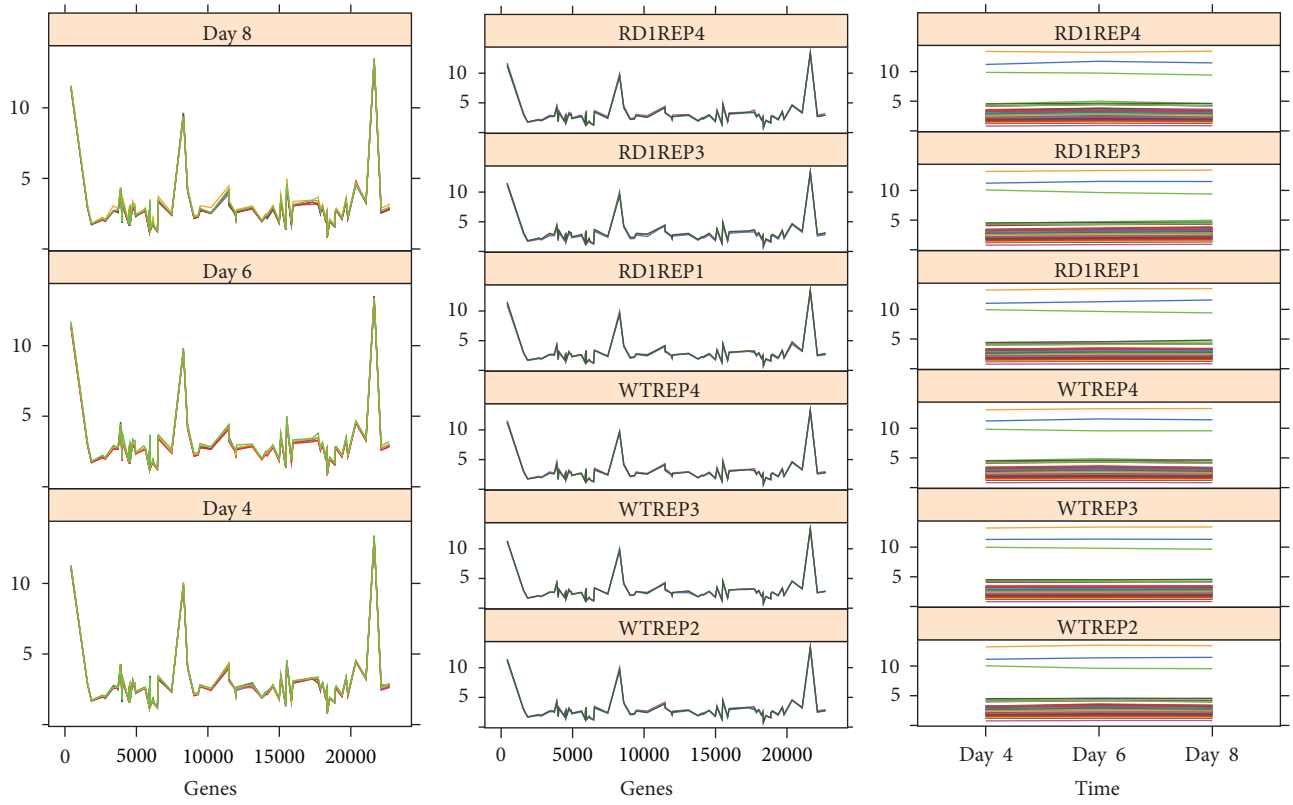
Table 7: Correlation results for tricluster Mouse GDS4510 Dataset.

TRI_{sol}	Pearson	Spearman
1	1	0.99
2	1	1
3	1	1
4	1	1
5	1	1
6	1	1
7	1	1
8	1	0.99
9	1	0.99
10	1	1
11	1	0.99
12	1	1
13	1	0.99
14	1	1
15	1	1
16	1	1
17	0.99	0.99
18	1	1
19	1	1
20	1	0.99

same reason we have increased w_g to favor individuals with a greater number of genes.

In Table 7 we see the correlation analysis for the 20 triclusters obtained. The correlation coefficients are very high and, in most cases, perfect with values close to one. This indicates almost perfect homogeneity between the genes, conditions, and times of the tricluster.

We show the graphs associated with solution TC_{20} with 78 genes, 6 conditions, and 3 time points in Figure 8. We see for the three views, Figures 8(a), 8(b), and 8(c), how all lines are totally aligned.



(a) Sample curves (b) Time curves (c) Gene curves

Figure 8: Graphical representation for tricluster TC_{20} found in the Mouse GSD4510 Dataset.

Table 8: GO analysis for tricluster TC_{20} in Mouse GDS4510 Dataset.

ID	Name	P-value
GO:0004953	Icosanoid receptor activity	1.525×10^{-6}
GO:0004955	Prostaglandin receptor activity	2.879×10^{-5}
GO:0004954	Prostanoid receptor activity	3.729×10^{-5}
GO:0001892	Embryonic placenta development	9.795×10^{-5}
GO:0004958	Prostaglandin F receptor activity	1.595×10^{-4}
GO:0060706	Cell differentiation involved in embryonic placenta development	2.868×10^{-4}
GO:0001890	Placenta development	5.151×10^{-4}
GO:0004982	N-formyl peptide receptor activity	7.342×10^{-4}
GO:0009265	2′-deoxyribonucleotide biosynthetic process	7.342×10^{-4}
G0:0046385	Deoxyribose phosphate biosynthetic process	7.342×10^{-4}

The biological validity of the solution shown can be found in Table 8 and yields good results regarding the terms listed and high statistical significance (P values below 0.05). The terms again are very specific and some are related to the dataset under study such as embryonic placenta development (GO:0001892) or cell differentiation involved in embryonic placenta development (GO:0060706).

4.4. Mouse GDS4442 Dataset. This time we have accessed the GEO database [17] to retrieve the dataset about the experiment under code GDS4442 titled *ectopic bHLH transcription factor expression Mesogenin1 effect on embryoid bodies: time course* [33]. This biological experiment examines the effect of doxycycline induction in mouse (*Mus musculus*) embryonic individuals at three stages of development: 12, 24, and 48 hours. Our input dataset $DGDS4442_{3D}$ is composed by 45101 genes, 6 experimental conditions (one for each individual involved in the biological experiment), and 3 time points.

Regarding the TriGen parameters, we increased G and I for the same reason as in the previous experiment, that is, to have more solutions in the evolutionary process with a larger number of generations due to size of $DGSD4442_{3D}$, see Table 9.

Regarding the correlation analysis, the results show high correlation values, highlighting the solutions TC_5, TC_6, and TC_{15} with Pearson's correlation values close to 1, see Table 10.

We show in Figure 9 the graphical representation of solution TC_{15} with 15 genes, 5 conditions, and 2 time points. We can see the great homogeneity among all genes, conditions, and times in Figures 9(a), 9(b), and 9(c).

The biological evaluation of tricluster TC_{15} shown in Table 11 shows annotated terms with high statistical significance, highlighting GO:0045127, GO:0009384, and

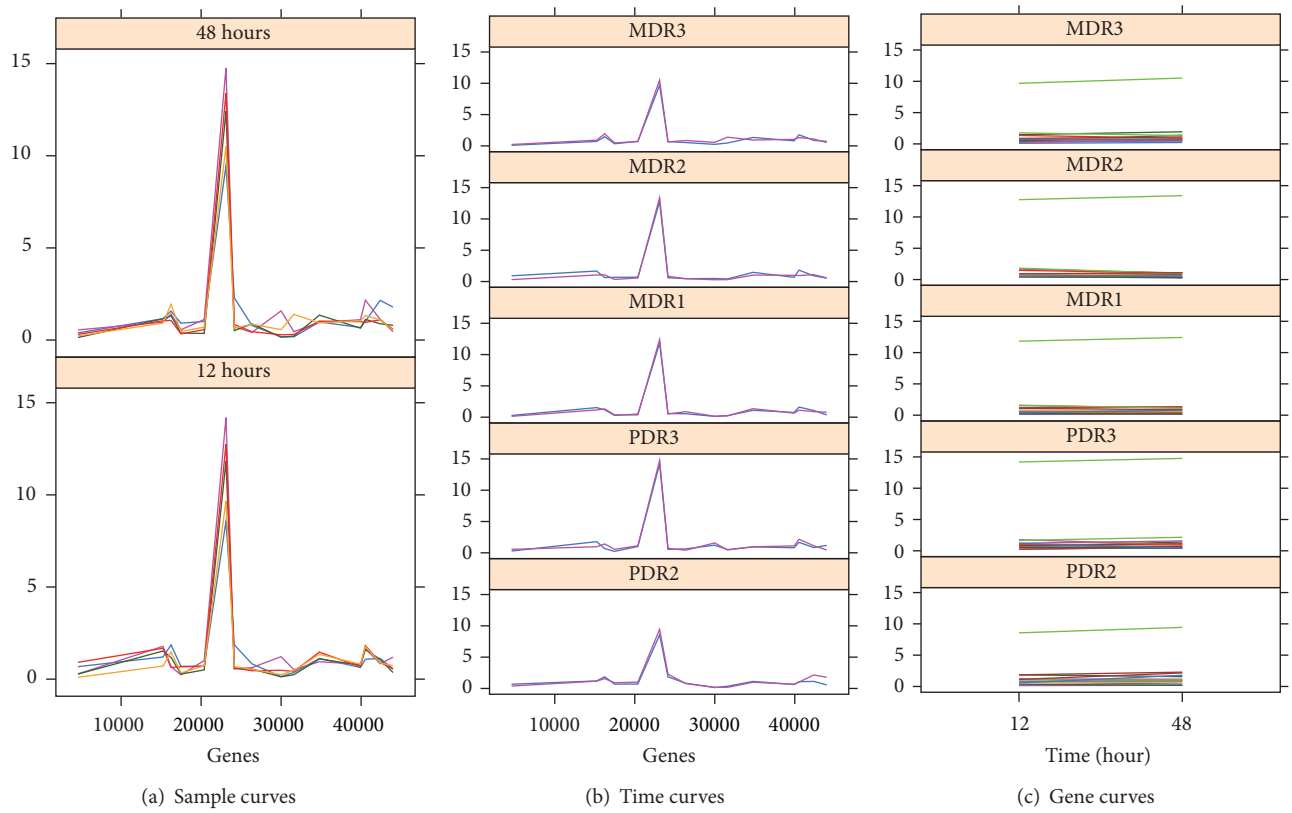


Figure 9: Graphical representation for TC_{15} found in the Mouse GSD4442 Dataset.

Table 9: TriGen algorithm control parameters for Mouse GDS4442 Dataset.

Parameter	Values
N	15
G	500
I	400
Ale	0.4
Sel	0.5
Mut	0.4
w_g	0.2
w_c	0.8
w_t	0.8
wo_g	0.5
wo_c	0.2
wo_t	0.2

Table 10: Correlation results for tricluster Mouse GDS4442 Dataset.

TRI_{sol}	Pearson	Spearman
1	0.52	0.53
2	0.34	0.36
3	0.49	0.53
4	0.52	0.45
5	0.92	0.79
6	0.86	0.83
7	0.39	0.31
8	0.35	0.44
9	0.46	0.44
10	0.6	0.58
11	0.62	0.62
12	0.59	0.58
13	0.76	0.61
14	0.61	0.64
15	0.98	0.6

GO:0019262 which are related to the cell wall synthesis which, in turn, is related to the action of doxycycline.

4.5. Human GDS4472 Dataset. This dataset has been obtained from GEO [17] under code GDS4472 titled *transcription factor oncogene OTX2 silencing effect on D425 medulloblastoma cell line: time course* [34]. In this experiment we analyze the effect of doxycycline on medulloblastoma cancerous cells at six times after induction: 0, 8, 16, 24, 48, and 96 hours. Our input dataset $DGSD4472_{3D}$ is composed by 54675 genes, 4 conditions (one for each individual involved), and 6 time points (one per hour).

Because of the volume of the dataset $D4472_{3D}$ we increase G and I to expand the space of solutions. The full set of parameters can be seen in Table 12.

Table 11: GO analysis for tricluster TC_{15} in Mouse GDS4442 Dataset.

ID	Name	P-value
GO:0045127	N-acetylglucosamine kinase activity	5.525×10^{-4}
GO:0009384	N-acylmannosamine kinase activity	0.001105
GO:0019262	N-acetylneuraminate catabolic process	0.002208
GO:0004957	Prostaglandin E receptor activity	0.002760
GO:0006054	N-acetylneuraminate metabolic process	0.003862
GO:0050901	Leukocyte tethering or rolling	0.004412
GO:0051352	Negative regulation of ligase activity	0.004963
GO:0051444	Negative regulation of ubiquitin-protein ligase activity	0.004963
GO:0090136	Epithelial cell-cell adhesion	0.006063
GO:0001921	Positive regulation of receptor recycling	0.006612

Table 12: TriGen algorithm control parameters for Human GDS4472 Dataset.

Parameter	Values
N	15
G	500
I	300
Ale	0.2
Sel	0.3
Mut	0.4
w_g	0.2
w_c	0.5
w_t	0.5
wo_g	0.5
wo_c	0.4
wo_t	0.4

Table 13: Correlation results for Human GDS4472 Dataset.

TRI_{sol}	Pearson	Spearman
1	0.94	0.78
2	1	0.8
3	0.86	0.87
4	0.98	0.72
5	0.98	0.81
6	0.95	0.93
7	0.99	0.67
8	0.99	0.62
9	0.88	0.71
10	0.88	0.73
11	0.99	0.75
12	0.85	0.69
13	0.89	0.72
14	1	0.76
15	0.98	0.67

We can see in Table 13 the high levels of correlation obtained for the 15 solutions found.

We graphically represent tricluster TC_2 with 25 genes, 2 conditions, and 2 time points in Figure 10. We can see the great homogeneity among all genes, conditions, and times in Figures 10(a), 10(b), and 10(c).

The biological validation can be seen in Table 14, where we see annotated terms with high statistical significance.

5. Conclusions

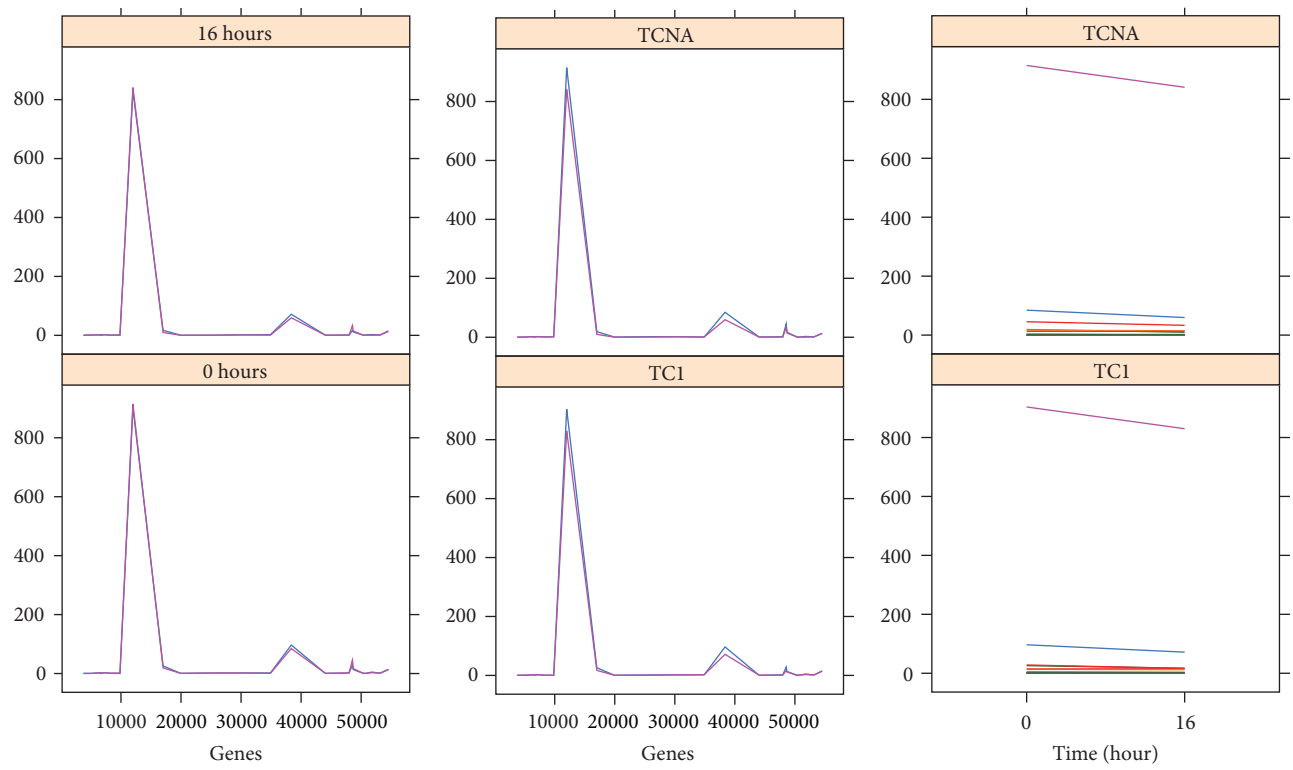
In this work we have presented a new evaluation measure for triclusters, MSR_{3D}, which measures the homogeneity among genes, conditions, and times in a tricluster. This measure has been inspired in the classic MSR measure proposed by Cheng and Church in [11]. A detailed formulation of both MSR and MSR_{3D} has been provided.

In order to assess the quality of the measure, we have applied it along with the TriGen algorithm [12], an evolutionary heuristic to mine triclusters from microarray experiments involving time, to several datasets: synthetically generated data, data from experiments with the yeast cell cycle (*Saccharomyces cerevisiae*) obtained from the Stanford University [16], and three datasets retrieved from Gene Expression Omnibus [17], two datasets are experiments for mouse (*Mus musculus*) and the third one is an experiment for humans (*Homo sapiens*). All experiments examine the behavior of genes under conditions at certain times.

The results obtained have been validated by means of analyzing the correlation among the genes, conditions, and times in each tricluster using two different correlation measures: Pearson and Filon [18] and Spearman [19]. Besides this, we have provided functional annotations for the genes extracted from the Gene Ontology project [20]. Regarding the synthetic data, we see that MSR_{3D} combined with TriGen has been capable of extracting almost all 10 triclusters artificially inserted in the dataset with a coverage of 90% to 96%. The results for the real datasets are also successful, with correlation values close to one, with the exception of the yeast dataset, where values are close to zero due to triclusters containing negatively correlated genes, found by MSR_{3D}.

The GO validation has given good results as well, with high levels of significance for the terms extracted (P values smaller than 0.05 and very specific terms). Graphical representation of the triclusters has also been provided.

MSR_{3D} is a tricluster evaluation measure created to assess the quality of triclusters extracted from temporal experiments with microarrays, but it can be used in other biologically related fields, for instance combining expression data with gene regulation information by means of substituting the time dimension by ChIP-chip data representing transcription factor-gene interactions which can provide us with regulatory network information. This proposal can also be applied to mine RNA-seq data repositories. Triclustering can also be applied to not biologically related fields, for instance, the seismic regionalization of areas at risk of undergoing an earthquake [43]. In this case, the third component does not



(a) Samples curves (b) Time curves (c) Gene curves

Figure 10: Graphical representation for tricluster TC_2 found in the Human GDS4472 Dataset.

Table 14: GO analysis for tricluster TC_2 in Human GDS4472 Dataset.

ID	Name	*P*-value
GO:0002753	Cytoplasmic pattern recognition receptor signaling pathway	4.543×10^{-4}
GO:2000299	Negative regulation of Rho-dependent protein serine/threonine kinase activity	8.415×10^{-4}
GO:2000298	Regulation of Rho-dependent protein serine/threonine kinase activity	8.415×10^{-4}
GO:2001264	Negative regulation of C–C chemokine binding	8.415×10^{-4}
GO:2001263	Regulation of C–C chemokine binding	8.415×10^{-4}
GO:0032479	Regulation of type I interferon production	0.001581
GO:0000226	Microtubule cytoskeleton organization	0.001755
GO:0032606	Type I interferon production	0.001762
GO:0070507	Regulation of microtubule cytoskeleton organization	0.001904
GO:0044092	Negative regulation of molecular function	0.001931

identify time points but features associated with every pair of geographical coordinates of the area under study.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

The authors want to thank the financial support given by the Spanish Ministry of Science and Technology with project TIN2011-28956-C02-02 and Junta de Andalucía with project TIC-7528.

References

[1] R. Durbin, S. Eddy, A. Krogh, and G. Mitchison, *Biological Sequence Analysis: Probabilistic Models of Proteins and Nucleic Acids*, Cambridge University Press, New York, NY, USA, 1998.

[2] P. O. Brown and D. Botstein, "Exploring the new world of the genome with DNA microarrays," *Nature Genetics*, vol. 21, no. 1, pp. 33–37, 1999.

[3] C. Rubio-Escudero, F. Martínez-Álvarez, R. Romero-Zaliz, and I. Zwir, "Classification of gene expression profiles: comparison

of K-means and expectation maximization algorithms," in *Proceedings of the 8th International Conference on Hybrid Intelligent Systems, HIS 2008*, pp. 831–836, Barcelona, Spain, September 2008.

[4] M. P. Tan, E. N. Smith, J. R. Broach, and C. A. Floudas, "Microarray data mining: a novel optimization-based approach to uncover biologically coherent structures," *BMC Bioinformatics*, vol. 9, article 268, 2008.

[5] P. D'Haeseleer, S. Liang, and R. Somogyi, "Genetic network inference: from co-expression clustering to reverse engineering," *Bioinformatics*, vol. 16, no. 8, pp. 707–726, 2000.

[6] A. Ben-Dor, B. Chor, R. Karp, and Z. Yakhini, "Discovering local structure in gene expression data: the order-preserving submatrix problem," in *Proceedings of the 6th Annual International Conference on Computational Biology*, pp. 49–57, April 2002.

[7] J. A. Hartigan, "Direct clustering of a data matrix," *Journal of the American Statistical Association*, vol. 67, no. 337, pp. 123–129, 1972.

[8] S. C. Madeira and A. L. Oliveira, "Biclustering algorithms for biological data analysis: a survey," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 1, no. 1, pp. 24–45, 2004.

[9] Z. Bar-Joseph, "Analyzing time series gene expression data," *Bioinformatics*, vol. 20, no. 16, pp. 2493–2503, 2004.

[10] F. Divina, B. Pontes, R. Giráldez, and J. S. Aguilar-Ruiz, "An effective measure for assessing the quality of biclusters," *Computers in Biology and Medicine*, vol. 42, no. 2, pp. 245–256, 2012.

[11] Y. Cheng and G. M. Church, "Biclustering of expression data," in *Proceedings of the International Conference on Intelligent Systems for Molecular Biology (ISMB '00)*, pp. 93–103, 2000.

[12] D. Gutiérrez-Avilés, C. Rubio-Escudero, F. Martínez-Álvarez, and J. C. Riquelme, "Trigen: a genetic algorithm to mine triclusters in temporal gene expression data," *Neurocomputing*, vol. 132, pp. 42–53, 2014.

[13] H. Banka and S. Mitra, "Evolutionary biclustering of gene expressions," *Ubiquity*, vol. 2006, article 5, 2006.

[14] S. Mitra and H. Banka, "Multi-objective evolutionary biclustering of gene expression data," *Pattern Recognition*, vol. 39, no. 12, pp. 2464–2477, 2006.

[15] A. Tanay, R. Sharan, and R. Shamir, "Discovering statistically significant biclusters in gene expression data," *Bioinformatics*, vol. 18, supplement 1, pp. S136–S144, 2002.

[16] P. T. Spellman, G. Sherlock, M. Q. Zhang et al., "Comprehensive identification of cell cycle-regulated genes of the yeast *Saccharomyces cerevisiae* by microarray hybridization," *Molecular Biology of the Cell*, vol. 9, no. 12, pp. 3273–3297, 1998.

[17] T. Barrett, S. E. Wilhite, P. Ledoux et al., "NCBI GEO: archive for functional genomics data sets—update," *Nucleic Acids Research*, vol. 41, pp. D991–D995, 2011.

[18] K. Pearson and L. N. G. Filon, "Mathematical contributions to the theory of evolution. IV. On the probable errors of frequency constants and on the influence of random selection on variation and correlation," *Philosophical Transactions of the Royal Society of London. Series A*, vol. 191, pp. 229–311, 1898.

[19] C. Spearman, "Correlation calculated from faulty data," *The British Journal of Psychology*, vol. 3, no. 3, pp. 271–295, 1910.

[20] M. Ashburner, C. A. Ball, J. A. Blake et al., "Gene ontology: tool for the unification of biology," *Nature Genetics*, vol. 25, no. 1, pp. 25–29, 2000.

[21] L. Zhao and M. J. Zaki, "TRICLUSTER: an effective algorithm for mining coherent clusters in 3D microarray data," in *Proceedings of the ACM SIGMOD International Conference on Management of Data (SIGMOD '05)*, pp. 694–705, June 2005.

[22] H. Jiang, S. Zhou, J. Guan, and Y. Zheng, "gTRICLUSTER: a more general and effective 3D clustering algorithm for gene-sample-time microarray data," in *Data Mining for Biomedical Applications*, vol. 3916 of *Lecture Notes in Computer Science*, pp. 48–59, Springer, New York, NY, USA, 2006.

[23] J. Liu, Z. Li, X. Hu, and Y. Chen, "Multi-objective evolutionary algorithm for mining 3D clusters in gene-sample-time microarray data," in *Proceedings of the IEEE International Conference on Granular Computing (GRC '08)*, pp. 442–447, Hangzhou, China, August 2008.

[24] X. Xu, Y. Lu, K. L. Tan, and A. K. H. Tung, "Finding time-lagged 3D clusters," in *Proceedings of the 25th IEEE International Conference on Data Engineering (ICDE '09)*, pp. 445–456, Shanghai, China, April 2009.

[25] G. Wang, L. Yin, Y. Zhao, and K. Mao, "Efficiently mining time-delayed gene expression patterns," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 40, no. 2, pp. 400–411, 2010.

[26] K. Sim, Z. Aung, and V. Gopalkrishnan, "Discovering correlated subspace clusters in 3D continuous-valued data," in *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM '10)*, pp. 471–480, Sydney, Australia, December 2010.

[27] Z. Hu and R. Bhatnagar, "Algorithm for discovering low-variance 3-clusters from real-valued datasets," in *Proceedings of the 10th IEEE International Conference on Data Mining (ICDM '10)*, pp. 236–245, Sydney, Australia, December 2010.

[28] Y. C. Liu, C. H. Lee, W. C. Chen, J. W. Shin, H. H. Hsu, and V. S. Tseng, "A novel method for mining temporally dependent association rules in three-dimensional microarray datasets," in *Proceedings of the International Computer Symposium (ICS '10)*, pp. 759–764, Tainan City, Taiwan, December 2010.

[29] P. Mahanta, H. A. Ahmed, D. K. Bhattacharyya, and J. K. Kalita, "Triclustering in gene expression data analysis: a selected survey," in *Proceedings of the 2nd National Conference on Emerging Trends and Applications in Computer Science (NCETACS '11)*, pp. 1–6, Shillong, India, March 2011.

[30] S. Gremalschi and G. Altun, "Mean squared residue based biclustering algorithms," in *Bioinformatics Research and Applications*, vol. 4983 of *Lecture Notes in Computer Science*, pp. 232–243, Springer, New York, NY, USA, 2008.

[31] M. Martínez-Ballesteros, F. Martínez-Álvarez, A. Troncoso, and J. C. Riquelme, "An evolutionary algorithm to discover quantitative association rules in multidimensional time series," *Soft Computing*, vol. 15, no. 10, pp. 2065–2084, 2011.

[32] V. M. Dickison, A. M. Richmond, A. Abu Irqeba et al., "A role for prenylated rab acceptor 1 in vertebrate photoreceptor development," *BMC Neuroscience*, vol. 13, article 152, 2012.

[33] R. B. Chalamalasetty, W. C. Dunty Jr., K. K. Biris et al., "The Wnt3a/β-catenin target gene Mesogenin1 controls the segmentation clock by activating a Notch signalling program," *Nature Communications*, vol. 2, no. 1, article 390, 2011.

[34] J. Bunt, N. E. Hasselt, D. A. Zwijnenburg et al., "OTX2 directly activates cell cycle genes and inhibits differentiation in medulloblastoma cells," *International Journal of Cancer*, vol. 7, no. 6, pp. E21–E32, 2011.

[35] S. Bauer, S. Grossmann, M. Vingron, and P. N. Robinson, "Ontologizer 2.0—a multifunctional tool for GO term enrichment analysis and data exploration," *Bioinformatics*, vol. 24, no. 14, pp. 1650–1651, 2008.

[36] P. Mendes, "GEPASI: a software package for modelling the dynamics, steady states and control of biochemical and other systems," *Computer Applications in the Biosciences*, vol. 9, no. 5, pp. 563–571, 1993.

[37] M. Barenco, J. Stark, D. Brewer, D. Tomescu, R. Callard, and M. Hubank, "Correction of scaling mismatches in oligonucleotide microarray data," *BMC Bioinformatics*, vol. 7, article 251, 2006.

[38] K. Hakamada, M. Okamoto, and T. Hanai, "Novel technique for preprocessing high dimensional time-course data from DNA microarray: mathematical model-based clustering," *Bioinformatics*, vol. 22, no. 7, pp. 843–848, 2006.

[39] R. P. Pargas, M. J. Harrold, and R. R. Peck, "Test-data generation using genetic algorithms," *Software Testing Verification and Reliability*, vol. 9, no. 4, pp. 263–282, 1999.

[40] Apache Commons, "Commons-math: the apache commons mathematics library," 2011.

[41] T. Zeng and J. Li, "Maximization of negative correlations in time-course gene expression data for enhancing understanding of molecular pathways," *Nucleic Acids Research*, vol. 38, no. 1, article e1, 2009.

[42] M. J. Brauer, C. Huttenhower, E. M. Airoldi et al., "Coordination of growth rate, cell cycle, stress response, and metabolic activity in yeast," *Molecular Biology of the Cell*, vol. 19, no. 1, pp. 352–367, 2008.

[43] J. Reyes and V. H. Cárdenas, "A Chilean seismic regionalization through a Kohonen neural network," *Neural Computing and Applications*, vol. 19, no. 7, pp. 1081–1087, 2010.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 350270, 11 pages
http://dx.doi.org/10.1155/2014/350270

Research Article

EEG Channel Selection Using Particle Swarm Optimization for the Classification of Auditory Event-Related Potentials

Alejandro Gonzalez, Isao Nambu, Haruhide Hokari, and Yasuhiro Wada

Department of Electrical Engineering, Nagaoka University of Technology, 1603-1 Kamitomioka, Nagaoka, Niigata 940-2188, Japan

Correspondence should be addressed to Alejandro Gonzalez; alejandrojgt@stn.nagaokaut.ac.jp

Received 6 January 2014; Accepted 26 February 2014; Published 25 March 2014

Academic Editors: V. Bhatnagar and Y. Zhang

Brain-machine interfaces (BMI) rely on the accurate classification of event-related potentials (ERPs) and their performance greatly depends on the appropriate selection of classifier parameters and features from dense-array electroencephalography (EEG) signals. Moreover, in order to achieve a portable and more compact BMI for practical applications, it is also desirable to use a system capable of accurate classification using information from as few EEG channels as possible. In the present work, we propose a method for classifying P300 ERPs using a combination of Fisher Discriminant Analysis (FDA) and a multiobjective hybrid real-binary Particle Swarm Optimization (MHPSO) algorithm. Specifically, the algorithm searches for the set of EEG channels and classifier parameters that simultaneously maximize the classification accuracy and minimize the number of used channels. The performance of the method is assessed through offline analyses on datasets of auditory ERPs from sound discrimination experiments. The proposed method achieved a higher classification accuracy than that achieved by traditional methods while also using fewer channels. It was also found that the number of channels used for classification can be significantly reduced without greatly compromising the classification accuracy.

1. Introduction

A brain-machine interface (BMI) is a system that allows a person to control or communicate with a computer or actuator using only brain signals. Since no muscle movements are needed, such systems are particularly useful to assist patients with motor disabilities such as amyotrophic lateral sclerosis or spinal cord injury.

One type of BMI system makes use of the P300 event-related potential (ERP), a neuroelectrical wave pattern that can be measured with electroencephalography (EEG). This is a pattern that carries information about the state of attention of the user and that can be robustly elicited by various types of stimuli through the oddball paradigm. In the oddball paradigm, the user pays attention to a series of incoming stimuli and must focus on the occurrences of a rare, task-relevant target stimulus hidden amongst frequent, irrelevant nontarget stimuli. Only the target stimuli elicit the P300 response when perceived by the user. The nontarget stimuli, on the other hand, are more frequent and are ignored by the user and thus do not elicit the P300 response. Therefore, by presenting to the user a stream of stimuli while measuring the user's brain the activity and then detecting the presence or absence of the P300 ERP, one can determine the intention of the user.

An example of this kind of system is the one proposed and investigated in [1–3]. This system allows a handicapped patient to communicate to a computer the direction of attention by detecting the P300 ERPs elicited with auditory stimuli from virtual-sound sources. By using the virtual sounds as the stimuli in the oddball paradigm, it is possible to estimate the intended direction of the user and thus control a transportation device like an electric wheelchair. The use of virtual sound stimuli allows the construction of more portable BMI systems and, in contrast to visual stimuli, allows the user to dedicate his or her vision to other tasks.

The detection of the P300 component is performed by means of a binary classifier fed with the signal features provided by the EEG signals, and its performance greatly depends on the chosen features. Ideally only the most

discriminative features should be used to feed the classifier but, in this case, it is usually not immediately clear which is the channel set that provides the most relevant information. Furthermore, the optimal set might not be equal for every subject. One could attempt to perform an exhaustive test of all possible combinations, but, in the case of dense array measurements, the extremely high number of combinations that exist (2^{64}, for a 64-channel array) render this approach intractable. Additionally, in practical applications, a BMI system for transportation purposes like the one described in [3] requires compact equipment and it is thus desirable to use a ERP detection system that employs as few channels as possible.

Stepwise Linear Discriminant Analysis (SWLDA) is one of the most popular classifiers used for the detection of ERPs, being used in numerous reports (see, e.g., [4–7]). This classifier is based on the feature selection performed by the Stepwise Regression algorithm in which the features that contribute the most are selected in a stepwise manner; that is, every feature is sequentially added or removed while measuring the predicting power at each step. However, a drawback of this algorithm is that the selection of the features depends on the order on which they are evaluated, especially when there are high correlations between the features [8] which is precisely the case in EEG measurements. Another algorithm worth mentioning is the one proposed by [9] that achieved the best performance in the BCI Competition III [10]. This algorithm performs a stepwise selection of channels following a classification accuracy maximization criterion, but, like SWLDA, the outcome of the procedure also depends on the order of the evaluation of channels. There are also algorithms that employ methods based on Principal Component Analysis (PCA) to extract spatial features [6, 11, 12], but these typically disregard channel selection and employ all the channels available to identify the most discriminative spatial information.

To address the abovementioned points, we propose an alternative approach in which the channel selection procedure is performed in an automated manner aimed at maximizing the classification accuracy of the system. Specifically, we propose a method for classifying P300 ERPs in which the features and the parameters of the classifier are tuned using a random optimization algorithm and evaluate it using experimental data. The proposed method performs classification using Fisher Discriminant Analysis (FDA) and uses a multiobjective hybrid real-binary Particle Swarm Optimization (MHPSO) algorithm to search for the classifier parameters and EEG channel set that simultaneously maximize the classification accuracy and minimize the number of channels used for classification. The PSO algorithm is multiobjective in the sense that it seeks to optimize a fitness function product of an aggregation of two performance metrics (i.e., the classification accuracy and the number of EEG channels) and hybrid real-binary in the sense that the search space is composed of both real and binary dimensions, necessary to tune the FDA regularization parameter (a real variable) and the addition or removal of EEG channels (a binary variable per channel).

PSO [13] is a relatively new stochastic algorithm for function optimization that has become increasingly popular and has also been applied in various fields [14]. This algorithm, inspired in the motion of fish schools that search for food, moves a swarm of particles around a parameter space searching for a solution that maximizes a certain fitness function. Like other stochastic search algorithms, this is a method that does not rely on the gradient of the function to optimize and instead looks for the best solution in a quasirandom way, being particularly useful in problems where an analytical expression for the optimization function is not available. It is important to remark that the PSO algorithm is not guaranteed to converge towards the global maximum. However, this algorithm has been applied with success in a wide variety of applications [14] and in practice tends to find a suitable solution if not the optimal one.

This text is organized as follows. First, the main components of the proposed algorithms, namely, the FDA and PSO algorithms, and their integration to classify ERP signals are described in Section 2. Then, the data and simulations used to evaluate the algorithm are described in Section 3. Lastly, in Section 4, the simulations results are shown and discussed and conclusions are given in Section 5.

2. Methods

Note on Mathematical Notation. Throughout this work, scalars are denoted by lowercase italic letters (e.g., x), vectors are written in lowercase, bold letters (e.g., $\mathbf{x}$), and matrices are denoted by uppercase bold letters (e.g., $\mathbf{X}$). All vectors are column vectors unless stated otherwise.

2.1. Particle Swarm Optimization. The objective of the PSO algorithm is to find the best parameters that maximize a given fitness function, and it does so by iteratively moving a swarm of particles around the parameter space according to especial equations. Each particle of the swarm is a particular position in the parameter space and represents a possible solution to the problem. Mathematically, a particle in PSO is a vector in an N-dimensional parameter space, and its position $\mathbf{x}$ and velocity $\mathbf{v}$ change according to the following equations:

$$x_{i,j}^{t} = x_{i,j}^{t-1} + v_{i,j}^{t}, \tag{1}$$

$$v_{i,j}^{t} = w v_{i,j}^{t-1} + c_1 \eta_1 \left(p_{i,j} - x_{i,j}^{t-1}\right) + c_2 \eta_2 \left(g_j - x_{i,j}^{t-1}\right), \tag{2}$$

where $x_{i,j}^{t}$ and $v_{i,j}^{t}$ are the jth components of the ith particle's position and velocity, respectively, at iteration t. $p_{i,j}$ is the jth component of $\mathbf{p}$, the best position that the ith particle has found so far, and g_j is the jth component of $\mathbf{g}$, the best position found by the swarm. The effect of $\mathbf{p}$ and $\mathbf{g}$ on the particle's motion is controlled by two constant parameters c_1 and c_2, and two independent random variables η_1 and η_2 uniformly distributed in $[0, 1]$. The particle's motion is also influenced by the velocity at the previous iteration and this effect is controlled by the inertia parameter w.

c_1 and c_2 are constants set by the experimenter that determine the balance between the exploitation of a potential

solution (movement towards **g**) and the exploration for new solutions (movement towards **p**). At each iteration, **p** and **g** are updated if positions with better fitness were found. This is the main feature of the PSO algorithm: each particle uses the information of its own history and the swarm's history, together with random perturbations, to search for the global maximum. The PSO algorithm, in an iterative manner, updates the velocity and positions of each particle, moving them around the parameter space, until the best global solution **g** reaches the desired fitness.

In practical applications, the search space is typically constrained to $[X_j^{\min}, X_j^{\max}]$ along each dimension j in order to limit the search space to feasible values. In this work, we enforced an invisible wall condition [15] in which the fitness of the particles that fly out of this region is neither calculated nor updated. Instead, the particles that stray out are expected (but not guaranteed) to eventually fly back into the admissible region. The invisible wall condition has the advantage of being simple to implement and avoiding the unnecessary computations required by the evaluation of unfeasible solutions. Lastly, to avoid the particles flying out of the feasible space too often, the velocity components are limited to a maximum value $V_j^{\max}$ such that

$$\left|v_{i,j}^t\right| < V_j^{\max}. \tag{3}$$

Hybrid Binary-Real PSO. In order to perform channel selection, in this work, we adopted a hybrid binary-real PSO algorithm that, in addition to real-valued variables, allows the optimization of variables that can take only two discrete values. In this case, the additional binary components are updated as shown below [15–17]:

$$x_{i,j}^t = \begin{cases} 1 & \text{if } r < S\left(v_{i,j}^t\right) \\ 0 & \text{if } r \geq S\left(v_{i,j}^t\right) \end{cases}, \quad S(x) = \frac{1}{1+e^{-x}}, \tag{4}$$

where r is a random number with a uniform distribution in $[0, 1]$ and the velocity components $v_{i,j}^t$ are updated in a manner similar to the real case (see (2)). This equation means that, at each iteration, the binary component $x_{i,j}^t$ will be 1 (i.e., the associated channel will be selected) with a probability of $S(v_{i,j}^t)$ (and 0 with a probability of $1 - S(v_{i,j}^t)$).

2.2. Fisher Discriminant Analysis. In this work, a linear binary classifier based on Fisher Discriminant Analysis (FDA) was applied to discriminate between target and nontarget signals. FDA is a machine learning technique proposed by [18] used for data classification. Strictly speaking, FDA is a dimensionality reduction used as a preprocessing step prior to classification and its goal is to find a linear combination of the features that maximizes the separation between the classes' distributions in the reduced space. Classification is then performed in this 1-dimensional space by applying some threshold criteria or by using any classifier trained on the transformed samples. Given a training dataset of n-dimensional samples $\{(\mathbf{z}_i, y_i),\ i = 1, \ldots, \ell\}$ (i.e., each sample has n features) were each of the samples belonging to either one of two classes K_1 (positive, e.g., target examples) or K_{-1} (negative, e.g., nontarget examples) as indicated by the categorical variable $y_i = \{+1, -1\}$, the transformation vector **w** is obtained by maximizing the following function:

$$J(\mathbf{w}) = \frac{\left\langle \mathbf{w}, \mathbf{m}_1 - \mathbf{m}_{-1} \right\rangle^2}{\mathbf{w}^T \mathbf{S}_W \mathbf{w}}. \tag{5}$$

($\langle \cdot, \cdot \rangle$ denotes the inner product of vectors,) $\mathbf{m}_1$ and $\mathbf{m}_{-1}$ are the means of the feature vectors of the positive and negative classes, respectively, and $\mathbf{S}_W$ is the within-class scatter matrix given by

$$\mathbf{S}_W = \sum_{j \in \{1,-1\}} \sum_{\mathbf{z}_i \in K_j} \left(\mathbf{z}_i - \mathbf{m}_j\right)\left(\mathbf{z}_i - \mathbf{m}_j\right)^T, \tag{6}$$

with

$$\mathbf{m}_1 = \frac{1}{\ell_1} \sum_{\mathbf{z}_i \in K_1} \mathbf{z}_i, \qquad \mathbf{m}_{-1} = \frac{1}{\ell_{-1}} \sum_{\mathbf{z}_i \in K_{-1}} \mathbf{z}_i. \tag{7}$$

The maximization of $J(\mathbf{w})$ therefore yields

$$\mathbf{w} = \mathbf{S}_W{}^{-1} \left(\mathbf{m}_1 - \mathbf{m}_{-1}\right), \tag{8}$$

and the classification of a new, unknown sample, **z**, is performed upon the score

$$s(\mathbf{z}) = \langle \mathbf{w}, \mathbf{z} \rangle, \tag{9}$$

which is the distance of the sample to the hyperplane parameterized by the normal vector **w** and represents a measure of the certainty about the class prediction. Regarding the class prediction, in this work, the class of the sample was determined by

$$\widehat{y} = \arg\min_{j \in \{1,-1\}} \sqrt{\frac{\left(s(\mathbf{z}) - \mu_j\right)^2}{\sigma_j^2}}, \tag{10}$$

where μ_j and σ_j are the mean and variance of the scores of the training samples belonging to class K_j ($j \in \{1, -1\}$) in the transformed space. Equation (10) means that the predicted class corresponds to the class that yields the shortest Mahalanobis distance to the sample.

The key element in the computation of **w** is the scatter matrix $\mathbf{S}_W$. $\mathbf{S}_W$ is an unbiased estimator of the true, unknown scatter matrix, and it may become imprecise when the number of features is high in comparison to the number of training samples. This is because the number of unknown parameters (the elements of the matrix) is quadratic in the number of features. An imprecise estimation of the within-class scatter matrix results in a degradation of classification performance [5]. To mitigate this effect, regularization is typically applied to the scatter matrix estimation and this is achieved by maximizing a modified target function $J(\mathbf{w})$ [19]:

$$J(\mathbf{w}) = \frac{\left\langle \mathbf{w}, \mathbf{m}_1 - \mathbf{m}_{-1} \right\rangle^2}{\mathbf{w}^T \mathbf{S}_W \mathbf{w} + \lambda \|\mathbf{w}\|^2}, \tag{11}$$

which results in

$$\mathbf{w} = \left(\mathbf{S}_W + \lambda \mathbf{I}\right)^{-1} \left(\mathbf{m}_1 - \mathbf{m}_{-1}\right), \tag{12}$$

where λ is the regularization parameter. It can be seen that for $\lambda = 0$ the canonical, unregularized FDA is obtained. Thus, it is necessary to appropriately choose the regularization parameter in order to achieve the best performance.

2.3. The Proposed Algorithm

2.3.1. Particle Encoding. As was mentioned before, the algorithm uses FDA for classification and a multiobjective hybrid PSO to tune, for each particular user, the channel set and classifier parameters that maximize the classification accuracy using as few channels as possible.

The proposed method employs a hybrid PSO algorithm to search in a space that contains both real and discrete (binary) dimensions in a similar fashion to [15, 17]. These dimensions correspond to the variables that are to be tuned. Thus, each particle is a position in the search space that represents a particular combination of FDA parameters and EEG channels and is a candidate solution for the problem. The channel set and classifier parameters are encoded in a particle $\mathbf{x}$ as follows:

$$\mathbf{x} = \left[a \;\; b_1 \;\; \cdots \;\; b_{64}\right], \tag{13}$$

where $a \in [-1, 1]$ and $b_j = \{0, 1\} \; \forall j$ denote the real and binary components, respectively. Each binary component (64 in total) b_j encodes whether the temporal features of the corresponding jth channel are used for classification ($b_j = 1$) or not ($b_j = 0$). Therefore, this encoding results in a search space of 65 dimensions. The real component a encodes the FDA regularization parameter λ, which is decoded as

$$\lambda = 10^{5a}. \tag{14}$$

This decoding was chosen because the feasible values for the regularization parameter usually range over several orders of magnitude, and by adopting an exponential representation, the particle can search with small steps for possible solutions over a broader interval.

2.3.2. Fitness Function. As mentioned above, each particle is a candidate solution and represents a possible combination of set of channels and FDA parameters. In this work, we search for the solution that yields the best performance measured in terms of the fitness function F:

$$\begin{aligned} F(\mathbf{x}) &= w_1 f_1(\mathbf{x}) + w_2 f_2(\mathbf{x}) \\ &= w_1 \sqrt{\frac{TP}{P_s} \times \frac{TN}{N_s}} + w_2 \left(\frac{N_{\text{Ch}} - n + 1}{N_{\text{Ch}}}\right), \end{aligned} \tag{15}$$

where TP stands for true positives, TN stands for true negatives, P_s and N_s are the total number of positive (target) and negative (nontarget) samples, respectively, N_{Ch} is the maximum number of channels that can be selected (64 in this work), and n is the number of channels in the set specified by the particle. In other words, F is a weighted aggregation of the optimization objectives f_1 and f_2: the maximization of the geometric mean between the true positive rate (target accuracy) and the true negative rate (nontarget accuracy) and the minimization of the number of channels. The trade-off between accuracy and number of channels is controlled by the fitness weights w_1 and w_2. Since the ratio between the weights is what actually determines the trade-off, in this work, we chose weights such that $w_1 + w_2 = 1$. The geometric mean was chosen over the arithmetic mean because it yields an aggressive evaluation in which solutions with low and/or highly unbalanced accuracies are heavily penalized. It is important to notice that the target and nontarget accuracies are not expressed as percentages and thus f_1 varies within $[0, 1]$, being 1 the perfect accuracy. f_2, on the other hand, yields higher values for fewer channels and becomes 1 when only one channel is selected. It is important to remark that, although the algorithm attempts to find a solution that yields a perfect classification using only one channel (fitness value equal to 1), such a solution might not exist because the information provided by one channel might not be enough to accurately discriminate the ERP signals.

In the offline analyses, for each particle, a classifier is built using the features and parameters encoded in the particle. Then, the true positive and true negative rates achieved by the classifier are estimated using 10-fold stratified cross-validation on the training data (all folds share the same target to nontarget sample ratio), and the particle fitness is finally calculated using (15). The proposed algorithm's flowchart is shown in Figure 1. A final classifier can then be trained using the best setup found by the algorithm and be used to classify signals in a real-time setting.

3. Simulations

3.1. Data Set Used in the Study. The data used in this work corresponds to the data gathered by [1, 2] of sound discrimination experiments that consisted of random presentations of virtual auditory stimuli from 6 directions. During these experiments, the subject had to focus his attention on one of these directions (the target direction) and count every time the stimulus source corresponded to the target direction. The subject had to ignore the stimuli from other directions (the nontarget directions). Each session consisted of 150 trials and each trial consisted of a 300 ms stimulus interval and an 800 ms silence interval. The stimulus was 300 ms of pure white noise. One of the 6 directions was fixed as the target direction throughout a single session and the corresponding stimuli were presented with a probability of 20%. Each direction was measured twice, thus yielding 12 recording sessions and a total of 1800 trials. Thus, about 20% of the samples in the data set correspond to target signals. This protocol is summarized in Figure 2(a) and the directions of the virtual sound sources are shown in Figure 2(b). The neural activity was recorded using a digital electroencephalograph (Active Two, Biosemi, Amsterdam, The Netherlands) with 64 electrodes attached to the subject's scalp using a cap. The electrodes were placed in accordance

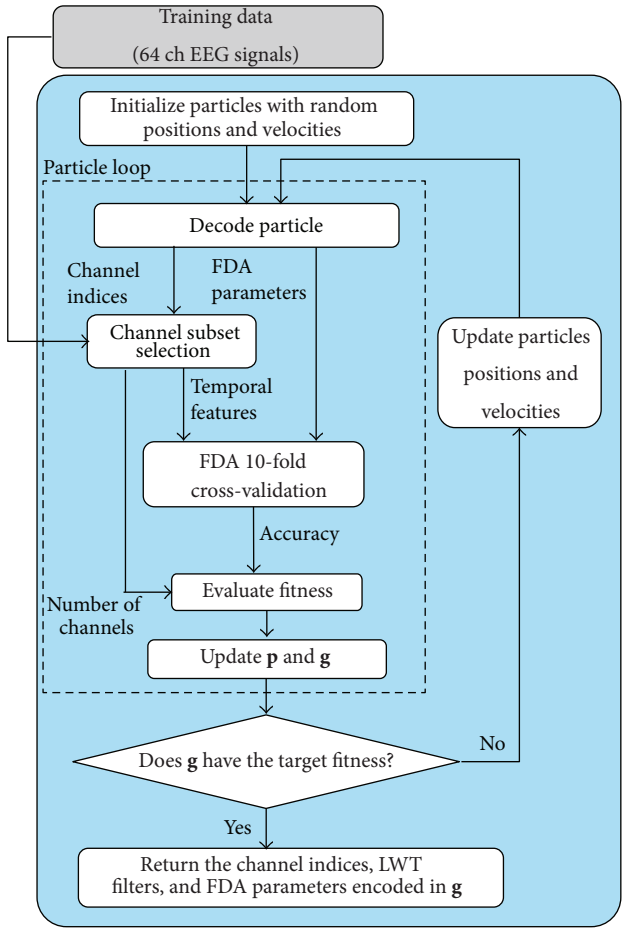


Figure 1: Flowchart of the proposed algorithm.

with the 10–20 system shown in Figure 2(c) and the reference was attached to the earlobes. Twelve healthy men (aged 22–24 years) participated in the experiment.

3.2. PSO Algorithm Setup. Preliminary tests were carried out to choose the best PSO parameters among the values suggested in [20, 21]. The chosen values are summarized as follows. All the PSO simulations were performed using 30 particles until a fitness value equal to 1 was achieved or until 100 iterations were exceeded. For the real part, the inertia parameter w was varied from 0.9 to 0.4 linearly across 100 iterations and had a constant value equal to 1 for the binary part. Both exploration and exploitation constants were set to $c_1 = c_2 = 2$. The search space of the only real component was constrained to $[-1, 1]$. The algorithm employed an invisible wall boundary condition and particles that encoded an empty channel set (i.e., all binary components equal to zero) were deemed as out-of-range. The maximum and minimum velocities were set to −0.1 and 0.1, respectively, for the real part, and −6 and 6, respectively, for the binary part. Regarding the particle's initial conditions at the start of the PSO algorithm, the real component was initialized to a random value in $[-1, 1]$ and each binary component was randomly initialized to either 0 or 1 with equal probability; thus, on average, each particle began the search process with half of the total channels selected.

3.3. Data Preprocessing and Features Used for Classification. Prior to classification, the data was divided in trials, with each trial being the 1100 ms segment of signal starting from −100 ms before the stimulus onset. Then, a zero-phase 3rd order Butterworth band-pass filter with cutoff frequencies of 0.1 Hz and 8 Hz was applied to all the signals across all channels. A baseline correction was performed by subtracting to each trial, at every channel, the mean of the signal in the prestimulus interval $[-100, 0]$ ms. Lastly, in order to reduce the number of temporal samples, each trial was downsampled by taking the average of every 10 samples.

The FDA classifier was fed with 25 temporal samples per EEG channel corresponding to the [0, 1000] ms trial segment. Thus the number of total features ranges from 25 to 1600 depending on the number of channels selected by the PSO algorithm.

3.4. Validation. The performance of the proposed algorithm was assessed by training and testing the algorithm on two disjoint training and test subsets of the original dataset (1800 single-trial samples in total). First, using the training dataset, the algorithm searches for the combination of channels and FDA parameters that maximize the classification accuracy and, when it finishes, it outputs the best configuration that could be found. A final classifier is then trained using the training dataset and the channel and classifier configuration specified by the output of the algorithm. Lastly, the performance of the final classifier is assessed by evaluating its classification accuracy on the test dataset. For the sake of consistency, all the classification accuracies reported in this work are derived from the same criteria used in the fitness function of the MHPSO algorithm, that is, the geometric mean of the target and nontarget accuracies. The subsets were made in such a way that trials of each type (i.e., target and nontarget), each direction (1 to 6), and each session (1 and 2) were present in the subsets with the same proportion as in the original set. Both the training and test sets had approximately 900 samples.

In addition to the single-trial accuracy, in all cases, the averaged-trial accuracy was also assessed. Trial-averaging is a technique usually employed to counteract the high level of noise typically found in EEG signals, and thus improve the classification accuracy. The improved accuracy comes at the cost of a reduced communication speed since several trials must be measured to produce a single estimation. In this work, a M averaged-trial was fabricated from the average classification score (using the score defined in (9)) of M single-trials of the same class randomly chosen with resampling. A set of averaged-trials is built by repeating this process until a given number of trials are fabricated. The number of samples in the averaged-trial dataset was made equal to the number of samples in the single-trial data set (around 900 samples) with the same ratio of target to nontarget samples. In reality, we made a master list containing

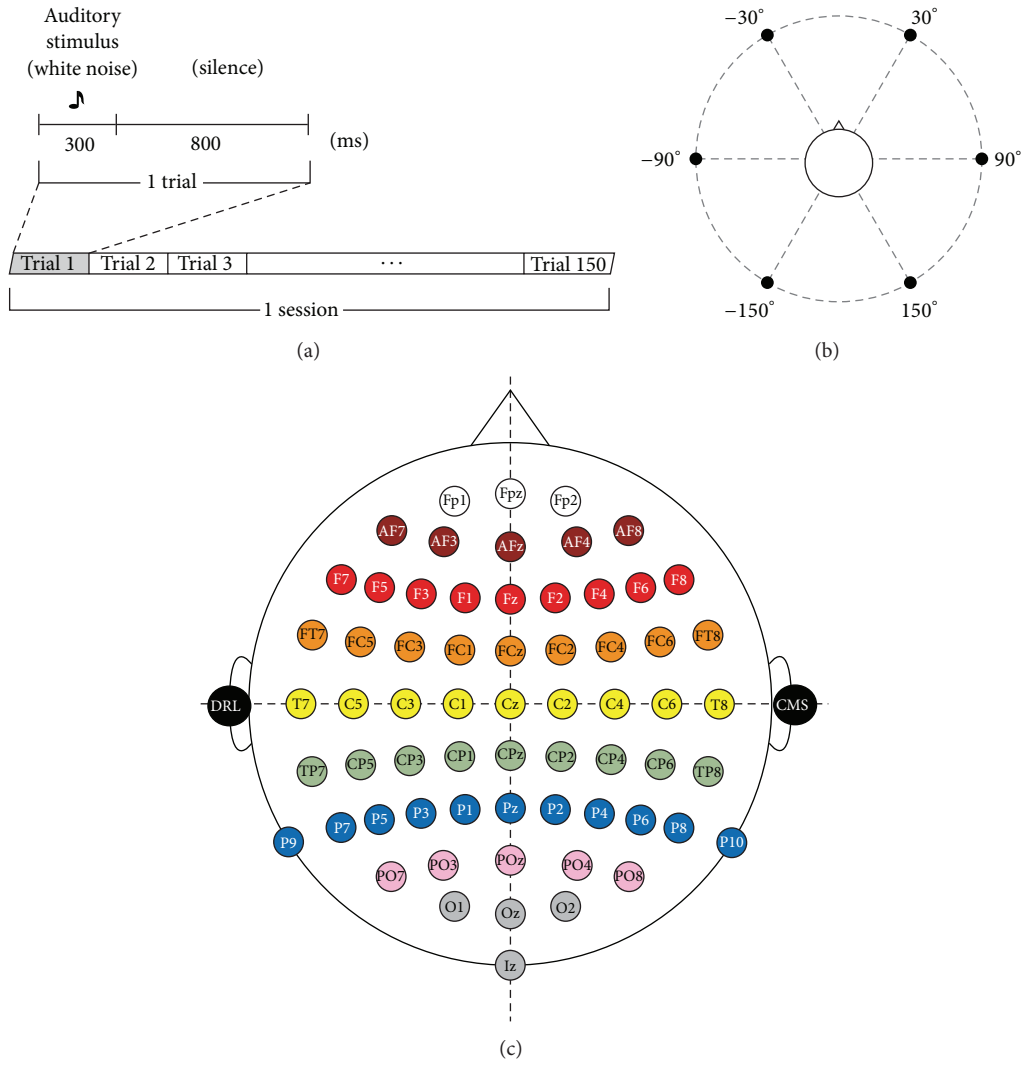


FIGURE 2: (a) Experimental protocol used to measure the auditory P300 signals. (b) Direction of the virtual sound sources. (c) 64 EEG channel layout used in the experiments. DRL and CMS are the references attached to the earlobes.

the indexes of the single-trials used to make each averaged-trial and used this list in all cases and all subjects to enforce that the averaged-trial datasets were fabricated using exactly the same information and thus ensure a fair comparison between cases. The averaged-trial accuracy was assessed for $M = 2 \ldots 10$.

Lastly, it is important to remark that score averaging was chosen over signal averaging because, if signals are averaged prior to classification, then (1) the number of samples available to train the classifier is reduced, and (2) the P300 component may cancel out if inter-trial jitter is present.

3.5. Simulation Cases. For every subject, eight cases were simulated to study the effect of the fitness function weights on the classification accuracy and number of selected channels that are ultimately achieved by the MHPSO algorithm. These cases are listed below.

Case 1: $w_1 = 1.00, w_1 = 0.00$.

Case 2: $w_1 = 0.95, w_1 = 0.05$.

Case 3: $w_1 = 0.90, w_1 = 0.10$.

Case 4: $w_1 = 0.85, w_1 = 0.15$.

Case 5: $w_1 = 0.75, w_1 = 0.25$.

Case 6: $w_1 = 0.65, w_1 = 0.35$.

Case 7: $w_1 = 0.50, w_1 = 0.50$.

Case 8: $w_1 = 0.35, w_1 = 0.65$.

Additionally, the algorithm was compared to a version that uses all 64 channels without channel selection and only tunes the FDA parameter (Fixed 64), SWLDA, and a combination of spatial PCA and SWLDA (PCA-SWLDA) similar to the one used in [6]. In the latter cases, the SWLDA algorithm was configured with a feature insertion P value of 0.1 and a feature removal P value of 0.15 as recommended by [7]. In the PCA-SWLDA case, the spatial principal components that accounted for 90% of the variability (the specific number of spatial factors varied between subjects) were used to transform the data prior to classification with SWLDA. In

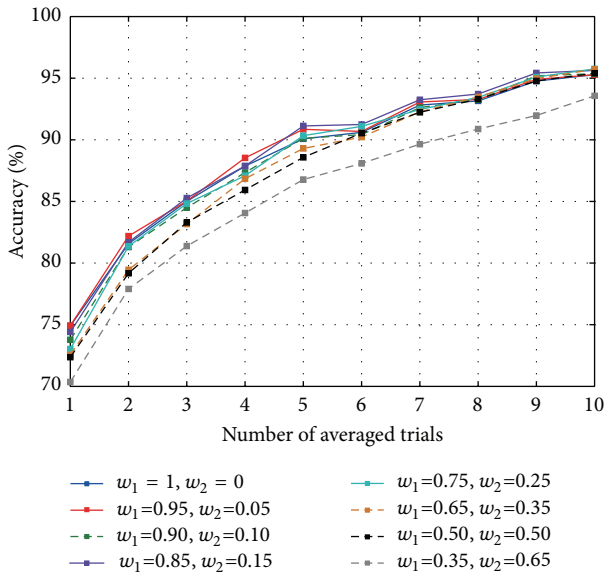


Figure 3: Average classification accuracy of 12 subjects as a function of the number of averaged samples for each case.

the SWLDA case, there were 1600 features to choose from (25 temporal features per channel) and, in the PCA-SWLDA case, there were between 75 and 200 features available for selection (25 transformed temporal features per principal component).

4. Results and Discussion

The average classification accuracy of 12 subjects as a function of the number of averaged samples for each MHPSO case is shown in Figure 3. From here, first, an improvement of classification with increasing trial averaging can be seen, as is usually expected. A two-way repeated measures analysis of variance (ANOVA) on the classification accuracy (factors: case and number of averaged samples) revealed a statistically significant difference between cases ($F(7, 77) = 3.440$, $P < 0.005$) and between the number of averaged samples ($F(9, 99) = 266.93$, $P < 0.001$). Since a significant interaction between the case and the number of averaged samples was also found ($F(63, 693) = 2.733$, $P < 0.001$), we performed analysis separately for the single-trial and the 10 averaged-trial cases.

The number of channels selected and the single-trial and 10 averaged-trial accuracies achieved in each MHPSO case are shown in Figure 4(a). The corresponding results for the Fixed 64, SWLDA, and PCA-SWLDA cases are shown in Figure 4(b). This figure illustrates the effect that fitness weights have on the number of selected channels. A one-way repeated measures ANOVA with Greenhouse-Geisser (GG) correction was conducted on the number of channels (factor: case), revealing a significant difference between the number of channels selected by each case ($\varepsilon = 0.593$, $F(4.15, 45.62) = 262.470$, $P < 0.001$). A *post hoc* multiple comparison test based on Holm's method revealed a statistically significant difference between all pairs of cases. Higher w_2/w_1 ratios indeed produced a more aggressive channel selection and selected sets with fewer channels.

While greater values of w_2/w_1 yielded fewer selected channels on average, the single-trial and 10 averaged-trial accuracies did not significantly change. A one-way repeated measures ANOVA with GG correction conducted on the single-trial accuracies (factor: case) found a significant difference between cases ($\varepsilon = 0.434$, $F(3.035, 33.383) = 8.085$, $P < 0.001$), but a *post hoc* Holm test found that these differences were only significant between pairs 1–8 ($P = 0.027$), 2–8 ($P = 0.027$) and 4–8 ($P = 0.027$). A similar analysis performed on the 10 averaged-trial accuracies did not find any significant difference between cases. These results suggest that the number of channels can be reduced without significantly hindering classification accuracy. For example, in Case 7, it can be seen that, with as few as 3 channels, the proposed algorithm could attain a single-trial accuracy slightly slower (around 3%) and an averaged-trial accuracy similar to what would be obtained using the full channel set.

The number of channels selected by the proposed and SWLDA algorithms were also compared. The frequency with which each channel was selected in both cases is shown in Figure 5. Two observations can be made from this figure. The first observation is that the SWLDA algorithm tended to choose most of the channels in most of the subjects, with the least frequently selected channels being chosen at least half of the time. A one-way repeated measures ANOVA conducted on the number of channels with GG correction (factor: case) showed a significant difference between the channels selected by the proposed method and SWLDA ($\varepsilon = 0.344$, $F(2.755, 30.305) = 366.49$, $P < 0.001$). All cases were significantly different to SWLDA ($P < 0.001$ for all pairs), confirming that indeed the proposed algorithm adapts sets with fewer channels. We hypothesized that the random initialization of the particles' binary bits gave the proposed algorithm an unfair advantage over other methods, but simulations where all the particle's binary part were initialized to all ones (i.e., all the channels selected) yielded similar results (results omitted for brevity). The second observation is that the proposed algorithm had a tendency to choose channels over the parietal, frontal, and frontal polar regions, as evidenced by the warm-colored spots over channels Pz, FC1, FCz, FC2, and FPz. While the parietal and frontal channels are in line with the spatial behavior of the P300 [22], the FPz channel is presumably being selected by the proposed algorithm to provide a mean of indirect noise reduction in the FDA classifier as suggested by [5, 23].

Another way to assess the trade-off relationship between the number of channels and the classification accuracy is to study the Pareto front encountered by the particles throughout the course of the optimization process. The Pareto front in this case represents the boundary at which an improvement of classification accuracy necessarily induces an increase of the number of channels or, conversely, an attempt to reduce the number of channels produces a loss of accuracy. An example of the Pareto front found in each case for a representative subject is shown in Figure 6 to illustrate this phenomenon. By looking at Figure 6, it becomes clear that, for increasing values of w_2/w_1, the positions visited by the swarm gradually

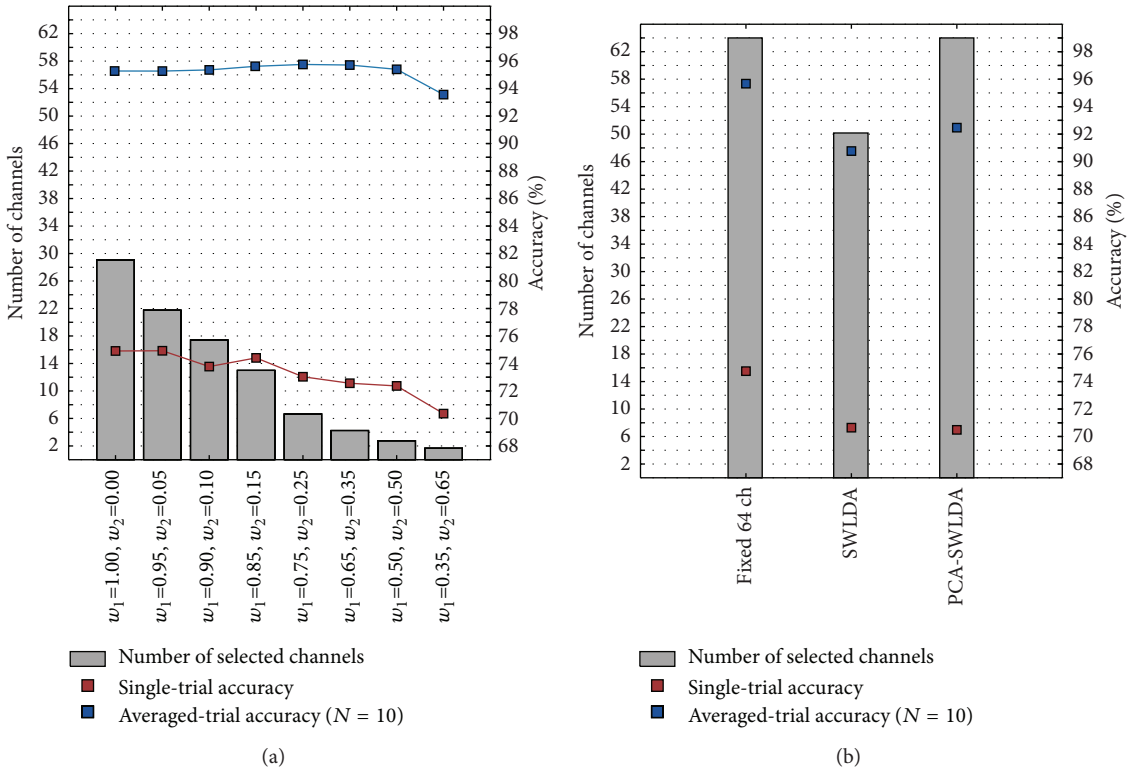


FIGURE 4: Number of selected channels (gray bars), single-trial (red line), and 10 averaged-trial (blue line) classification accuracy for (a) each MHPSO case and (b) the Fixed 64, SWLDA, and PCA-SWLDA cases.

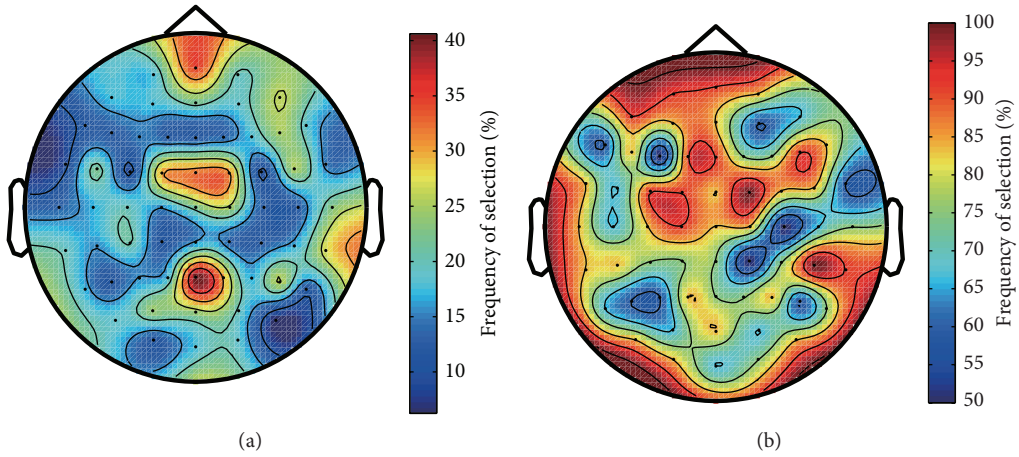


FIGURE 5: Frequency of selection of channels in (a) the proposed algorithm (across all MHPSO cases and subjects) and (b) SWLDA (across all subjects). Note that the color gradations denote different ranges in each figure.

move towards the bottom and that once an accuracy of about 80% is reached, the Pareto front is encountered and a trade-off between number of channels and accuracy begins. Nevertheless, the existence of positions with equal accuracy but different number of channels that can be seen in any of the cases is in line with the abovementioned results.

Lastly, following the results shown in Figure 5(a), it is worth asking if one could use MHPSO to find the most important channels for classification, make a fixed a channel set with a few of those channels, and use it as a general purpose set for every subject. To assess this, we chose the 5 most frequently selected channels across all subjects and cases and trained an algorithm on this reduced set of channels (like Fixed 64, this case only tuned the FDA regularization parameter). We call this case Fixed 5 and compared it to the Fixed 64, SWLDA, PCA-SWLDA, and MHPSO(4) cases. The channel set is shown in Figure 7 and the results are shown in Figure 8. A one-way repeated measures ANOVA

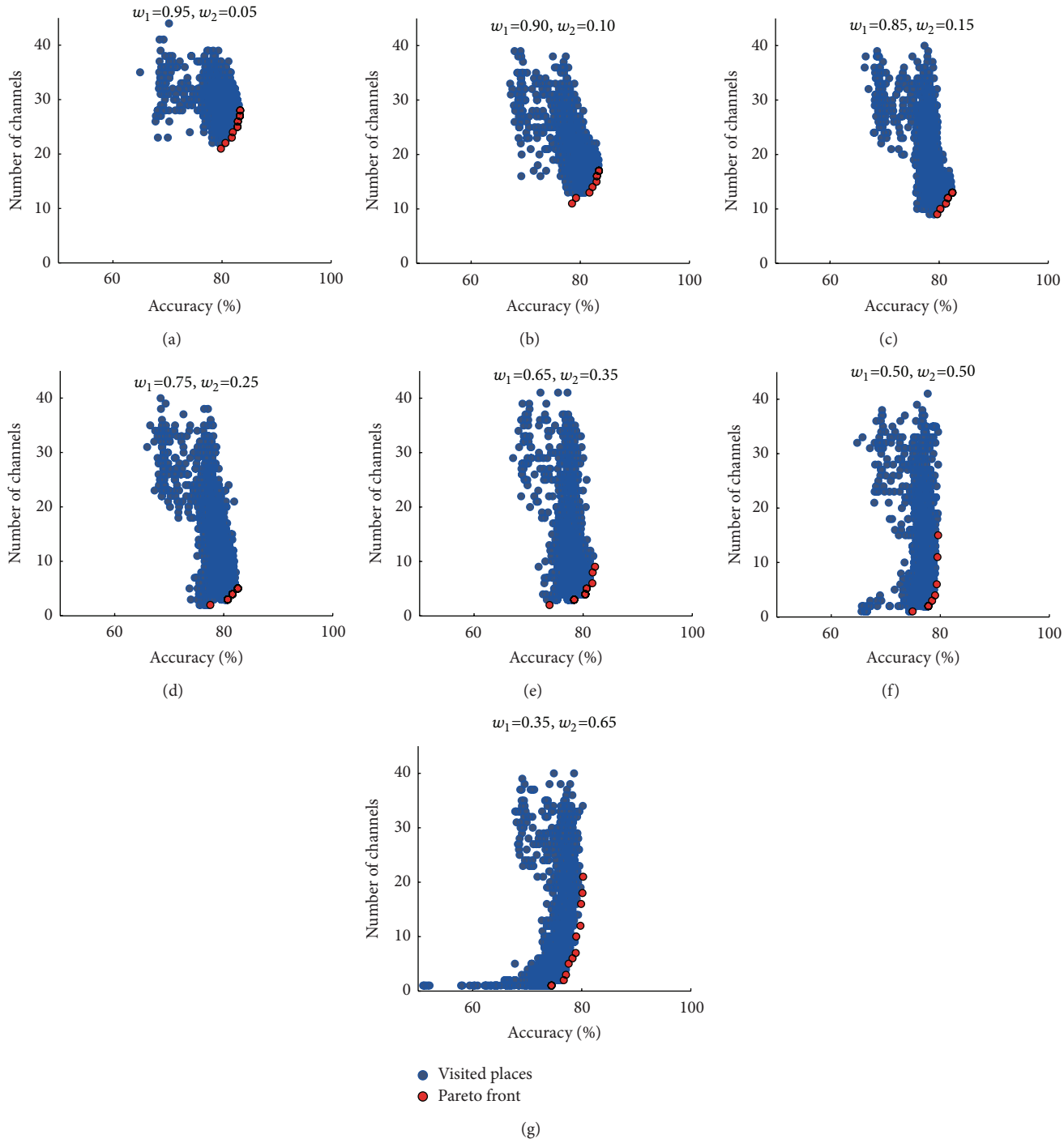


FIGURE 6: Illustration of the Pareto front found by the algorithm in each MHPSO case for a representative subject. The vertical and horizontal axes denote the number of selected channels and the average classification accuracy, respectively. The blue dots show the fitness of all the positions visited by the swarm. The red dots represent are the positions that belong to the Pareto front. The first MHPSO case was omitted because the Pareto front in this case is meaningless.

with GG correction conducted on the single-trial accuracies of the 5 algorithms (factor: case) found a significant difference between cases ($\varepsilon = 0.610$, $F(2.440, 26.838) = 9.880$, $P < 0.001$). After a *post hoc* Holm test, it was found that these differences were significant between pairs Fixed 64-SWLDA ($P < 0.001$) and MHPSO(4)-SWLDA ($P < 0.001$). Other significantly different pairs are Fixed 64-PCA-SWLDA ($P = 0.012$), Fixed 64-Fixed 5 ($P = 0.032$), and MHPSO(4)-PCA-SWLDA ($P = 0.029$). No significant differences were found in the 10 averaged-trial case. These results show that, although

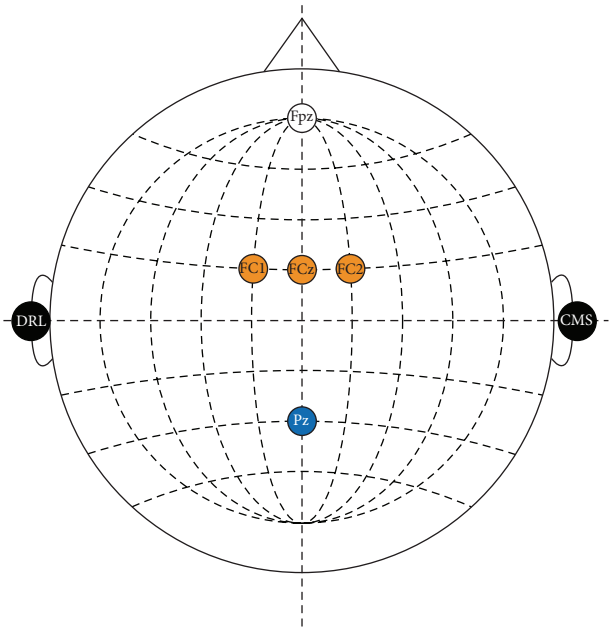


Figure 7: Channel set of 5 fixed channels built from the channels that were most frequently selected by the proposed algorithm.

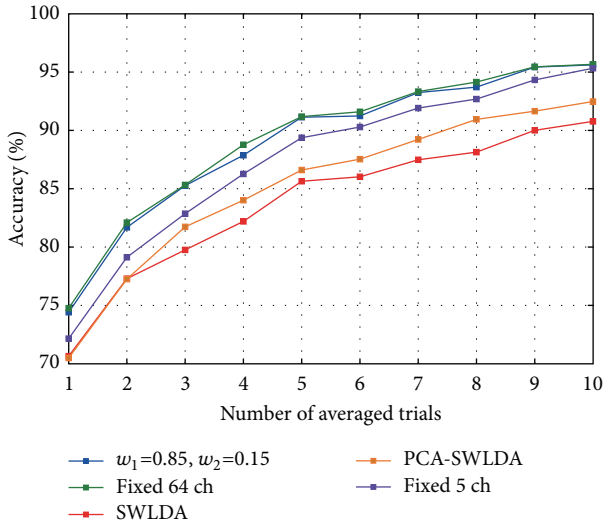


Figure 8: Average classification accuracy across 12 subjects achieved by the Fixed 5 case (purple line). Other colors show the accuracy obtained by the Fixed 64, SWLDA, PCA-SWLDA, and the MHPSO(4) cases.

the algorithm using the full channel set provided the best accuracy, the Fixed 5 and MHPSO(4) algorithms yielded a performance comparable to or better than SWLDA and PCA-SWLDA using considerably fewer channels, thus being adequate for BMI systems where portability and simplicity are important. Although there was no significant difference between the Fixed 5 and MHPSO(4) cases ($P = 0.197$), there also seems to be a difference between using a fixed channel set assessed with MHPSO across all users or using a channel set adapted to each user. This difference, however, may become significant if the number of subjects is increased. Ultimately, the decision of whether to give priority to the accuracy or the compactness of the EEG channel set will depend on the particular constraints of each application. In the case that classification accuracy can be spared, the proposed algorithm can provide hints as to which channels can be used as general purpose set or to adapt a channel set to each particular user.

5. Conclusions

An algorithm based on FDA and MHPSO for the classification of P300 ERP signals was presented. The algorithm used MHPSO to find the FDA parameters and set of the fewest EEG channels that maximized the classification accuracy. The algorithm's performance was evaluated through offline analyses on datasets of auditory ERPs from sound discrimination experiments. The proposed method achieved a higher classification accuracy than that achieved by traditional methods while also using fewer channels. It was also found that it is possible to reduce the number of channels necessary for classification without greatly compromising the classification accuracy. Future work will be aimed at finding why, in addition to channels on the parietal and frontal regions typically associated to the P300 ERP, channels on the frontal polar region were selected. Also, given that the proposed algorithm can be easily extended to spatiotemporal feature selection, further research will focus on a version that tunes more binary variables to select the channel set and the time intervals within each channel that maximize performance. Lastly, future research will also explore the application of other swarm intelligence techniques such as Ant Colony Optimization and the Firefly Algorithm and compare it with the results obtained with MHPSO.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by JSPS KAKENHI Grant nos. 24300051, 24650104, and 24800027.

References

[1] M. Kogure, M. Ebisawa, S. Yano, S. Matsuzaki, and Y. Wada, "Estimation of human's chosen direction using auditory-evoked event-related potentials," Tech. Rep., IEICE, Tokyo, Japan, 2011.

[2] M. Ebisawa, M. Kogure, S.-H. Yano, S.-I. Matsuzaki, and Y. Wada, "Estimation of direction of attention using EEG and out-of-head sound localization," in *Proceedings of the 33rd Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBS '11)*, vol. 2011, pp. 7417–7420, Boston, Mass, USA, September 2011.

[3] I. Nambu, M. Ebisawa, M. Kogure, S. Yano, H. Hokari, and Y. Wada, "Estimating the intended sound direction of the user: toward an auditory brain-computer interface using out-of-head sound localization," *PloS ONE*, vol. 8, no. 2, article e57174, 2013.

[4] D. J. Krusienski, E. W. Sellers, D. J. McFarland, T. M. Vaughan, and J. R. Wolpaw, "Toward enhanced P300 speller performance," *Journal of Neuroscience Methods*, vol. 167, no. 1, pp. 15–21, 2008.

[5] B. Blankertz, S. Lemm, M. Treder, S. Haufe, and K.-R. Müller, "Single-trial analysis and classification of ERP components—a tutorial," *NeuroImage*, vol. 56, no. 2, pp. 814–825, 2011.

[6] S. Kamp, A. R. Murphy, and E. Donchin, "The component structure of event-related potentials in the P300 speller paradigm," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 21, no. 6, pp. 897–907, 2013.

[7] D. J. Krusienski, E. W. Sellers, F. Cabestaing et al., "A comparison of classification techniques for the P300 Speller," *Journal of Neural Engineering*, vol. 3, no. 4, article 007, pp. 299–305, 2006.

[8] S. Derksen and H. J. Keselman, "Backward, forward and stepwise automated subset selection algorithms: frequency of obtaining authentic and noise variables," *British Journal of Mathematical and Statistical Psychology*, vol. 45, no. 2, pp. 265–282, 1992.

[9] A. Rakotomamonjy and V. Guigue, "BCI competition III: dataset II- ensemble of SVMs for BCI P300 speller," *IEEE Transactions on Biomedical Engineering*, vol. 55, no. 3, pp. 1147–1154, 2008.

[10] B. Blankertz, "BCI Competition III Webpage," 2004.

[11] B. Blankertz, R. Tomioka, S. Lemm, M. Kawanabe, and K.-R. Muller, "Optimizing spatial filters for robust EEG single-trial analysis," *IEEE Signal Processing Magazine*, vol. 25, no. 1, pp. 41–56, 2008.

[12] D. J. Krusienski, E. W. Sellers, and T. M. Vaughan, "Common spatio-temporal patterns for the P300 speller," in *Proceedings of the 3rd International IEEE EMBS Conference on Neural Engineering (CNE '07)*, pp. 421–424, Kohala Coast, Hawaii, USA, May 2007.

[13] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Proceedings of the IEEE International Conference on Neural Networks*, vol. 4, pp. 1942–1948, Perth, Australia, December 1995.

[14] R. Poli, "Analysis of the publications on the applications of particle swarm optimisation," *Journal of Artificial Evolution and Applications*, vol. 2008, Article ID 685175, 10 pages, 2008.

[15] N. Jin and Y. Rahmat-Samii, "Hybrid real-binary particle swarm optimization (HPSO) in engineering electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 58, no. 12, pp. 3786–3794, 2010.

[16] J. Kennedy and R. Eberhart, "A discrete binary version of the particle swarm algorithm," in *Proceedings of the IEEE International Conference on Systems, Man, and Cybernetics*, vol. 5, pp. 4104–4108, Orlando, Fla, USA, October 1997.

[17] N. Jin and Y. Rahmat-Samii, "Advances in particle swarm optimization for antenna designs: real-number, binary, single-objective and multiobjective implementations," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, part 1, pp. 556–567, 2007.

[18] R. A. Fisher, "The use of multiple measurements in taxonomic problems," *Annals of Eugenics*, vol. 7, no. 7, pp. 179–188, 1936.

[19] J. Shawe-Taylor and N. Cristianini, *Kernel Methods for Pattern Analysis*, Cambridge University Press, New York, NY, USA, 2004.

[20] R. Eberhart and Y. Shi, "Particle swarm optimization: developments, applications and resources," in *Proceedings of the Congress on Evolutionary Computation*, vol. 1, pp. 81–86, Seoul, Republic of Korea, May 2001.

[21] J. Robinson and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Transactions on Antennas and Propagation*, vol. 52, no. 2, pp. 397–407, 2004.

[22] J. Polich, "Updating P300: an integrative theory of P3a and P3b," *Clinical Neurophysiology*, vol. 118, no. 10, pp. 2128–2148, 2007.

[23] D. McFarland, C. Anderson, K. R. Müller, A. Schlögl, and D. Krusienski, "BCI meeting 2005-workshop on BCI signal processing: feature extraction and translation," *IEEE Transactions on Neural Systems and Rehabilitation Engineering*, vol. 14, no. 2, pp. 135–138, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 964870, 13 pages
http://dx.doi.org/10.1155/2014/964870

Research Article

A Fusion Method of Gabor Wavelet Transform and Unsupervised Clustering Algorithms for Tissue Edge Detection

Burhan Ergen

Department of Computer Engineering, Faculty of Engineering, Firat University, 23119 Elazig, Turkey

Correspondence should be addressed to Burhan Ergen; burhanergen@gmail.com

Received 3 December 2013; Accepted 20 February 2014; Published 23 March 2014

Academic Editors: S. Balochian and V. Bhatnagar

This paper proposes two edge detection methods for medical images by integrating the advantages of Gabor wavelet transform (GWT) and unsupervised clustering algorithms. The GWT is used to enhance the edge information in an image while suppressing noise. Following this, the *k*-means and Fuzzy *c*-means (FCM) clustering algorithms are used to convert a gray level image into a binary image. The proposed methods are tested using medical images obtained through Computed Tomography (CT) and Magnetic Resonance Imaging (MRI) devices, and a phantom image. The results prove that the proposed methods are successful for edge detection, even in noisy cases.

1. Introduction

Magnetic Resonance Imaging (MRI) and Computed Tomography (CT) devices are currently the most important diagnostic tools for medical examination. These imaging techniques provide a wealth of information about biological tissues and the condition of anatomical structures. The determination of tissue boundaries plays an important role in many medical applications aimed at identifying the abnormalities in anatomical structures and tissues [1–3]. Also, clinicians planning invasive or noninvasive treatments, such as surgery or radiotherapy, require the correct edge of tumours or tissues to be identified in order to operate. To determine the accurate edge of abnormal tissues helps clinicians to create or modify a treatment plan or to select the region and the path of the operation. Generally, the edge detection and the segmentation are performed by trained radiologists manually. Because the edge detection process is a time consuming task and is subject to radiologist error, researchers have concentrated on developing edge detection methods which detect the accurate edge [4–6]. The detection of the accurate or the acceptable edge is very difficult due to the properties of the analysed image and the variety of modalities. The medical images received from medical devices, such as MRI or CT, are always corrupted by noise and the artifacts of the devices [7–10]. Therefore, one of the difficulties for edge detection of medical images is the effect of noise and artifacts disturbing the edge of the analyzed images.

It is not enough that detected edges are visually soft and nice. The detected edges should be determined accurately. In the edge detection process, it is commonly considered that an edge detection method should detect all the edges of the objects in the examined image at their correct positions and should not detect non-edge. In an image, the edge is generally obtained using gradient, texture, and intensity, which are the measurable features of the examined image. Early edge detection methods employ local operators to compute the first and second gray-level gradients of an image in the spatial domain. After this, the local maximum locations of the first derivatives or the second derivatives are assumed as the edge points in the analyzed image. In the same way that classical edge detection operators considered benchmark methods [11], Sobel operator, Prewitt operator, LOG operator, and Robert operators compute the derivatives or the gradients. Some edge-detector methods use second-order derivatives like Laplacian or DoG (Difference of Gaussian). If the image

to be analyzed has poor contrast values and noise between the interested regions and the weak boundaries, the operator-based edge detection methods encounter difficulties in the detection of the correct edge [7, 8].

However a medical image cannot be free of noise and artifacts; the medical images acquired from CT or MRI devices usually suffer from noise and artifacts. These factors reduce the quality of the medical images used to detect the edges. Therefore, a lot of edge detection methods have been put forward recently which detect the boundaries of tissues in a medical image such as wavelet transform, mathematical morphological method, neural networks, or fuzzy methods [12–15].

Even if these studies are successful in identifying the external shape of the interested tissues, it is observed that the usage of the prior information and the shape models does not accurately identify the internal structural changes of the interested tissues. In order to identify the external and internal edge of a tissue, automatic edge detection methods could help because a shape model is not able to model the edge of the internal structures. Thus, some new methods based on soft computing algorithms such as fuzzy, neural, or genetic algorithms have been proposed so that the internal structures may be appropriately determined [16, 17].

Since automatic edge detection and segmentation are very difficult, some studies concentrating on a particular problem have used shape models or prior information about the tissues [18]. Although soft computing algorithms have been successful in providing more reliable edges than the traditional and shape modeling methods, it is also reported that these methods are highly sensitive to noise and artifacts [11, 19, 20]. FCM and k-means segmentation methods are frequently encountered methods among the soft computing methods used for segmentation problems [21, 22].

Whereas these methods just consider the intensity of the given image, it is reported that FCM cannot give good results if the image is noisy or does not have homogeneous structures [13, 19, 23]. If the existence of noise and artifacts in medical images is taken into consideration, it is clear the FCM is insufficient for the detection of accurate edge. However many algorithms have been proposed for the improvement of FCM; the FCM-based algorithms, when used alone, are still not robust against noise and nonhomogeneity [19, 21, 24]. The performances of soft computing methods for edge detection are decreased when they are used alone because the noise obscures the weak edges. Traditional edge detectors and soft computing algorithms, which may identify the edge of the internal structures for tissue segmentation, are highly sensitive to noise. If the image has a low signal-to-noise ratio, the traditional edge detectors and the soft computing algorithms fail to determine the contours of the anatomical structures correctly [25].

Therefore, many noise suppression methods have been proposed for the enhancement of the image to be analyzed. However some of the image denoising methods use the pixel relation in spatial domain; the rest assumes that the rapid change in frequency domain refers to noise [26].

While several noise suppression methods which filter the background noise of an image have been proposed, the traditional noise reduction methods are based on the median filter such as Adaptive Weighted Median Filter (AWMF) or the mean filter such as Homogeneous Region Growing Mean Filter (HRGMF) [27]. An improved version of the HRGMF filter, Aggressive Region Growing Filter (ARGF), is proposed in [28, 29]. This filter uses an adaptive homogeneity threshold instead of the constant threshold value of the HRGMF filter.

Some of the image denoising methods work on frequency domain such as wavelet-based methods. In wavelet-based image denosing process, the image is decomposed into four subimages with respect to their frequency bands in one level decomposition. Afterwards, the small detail coefficients are properly eliminated [26]. Any noise reduction method makes the image blurred less or more. However the noise reduction methods increase the SNR of the given image, and the weak edges in the images become invisible and undetected.

Although edge detection is a very difficult task, humans can easily determine the boundaries within an image without realizing consciously that they are doing so. The human visual system can be modeled as a filter bank. This filter bank can be represented using Gabor functions having different orientation and frequencies. The output of the representation using Gabor function can be accepted as the responses of the human visual system. In particular, the Gabor wavelet transform has demonstrated good performance in texture representation and discrimination [30, 31], and it has been successfully applied to face recognition, object detection [32], palm print recognition [33], and also object tracking [34]. Therefore, we have developed a technique integrating the advantages of Gabor wavelet transform and unsupervised clustering algorithms, FCM and k-means. The GWT is used as a tool for enhancing the edge of images while suppressing noise. The clustering algorithms convert the gray level edge image into a binary image without any thresholding process. The estimation of an appropriate threshold value is a very difficult task if the histogram of the given image has multiple valleys. Here, we have used the clustering methods, k-means and FCM, for the binary conversion avoiding any thresholding process. Figure 1 demonstrates the proposed edge detection method. The proposed algorithm has three steps: the convolution of the input image with Gabor functions, a clustering algorithm to obtain the binary image, and morphological operation to detect the edge.

The paper is organized as follows. Section 2 introduces GWT. In Section 3, k-means and the FCM, the most known and used unsupervised algorithms, are presented. In Section 4, the experiments are performed on medical images and a phantom image, in comparison with traditional edge detection methods, Prewitt, Canny, and Sobel. The conclusion is given in Section 5.

2. Gabor Wavelet Transform (GWT)

Gabor functions, with different frequencies and orientation, can model the human visual system as a filter bank [33, 35]. A Gabor wavelet can be described as a Gaussian kernel function modulated by a sinusoidal plane wave that has an optimal location in both the frequency domain and the

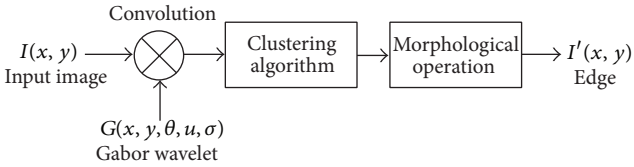


Figure 1: The proposed edge detection method.

space domain. In literature, there is a significant amount of computer vision applications using Gabor functions, such as texture segmentation, image analysis, and discriminations [33, 36].

Gabor wavelets reveal the directional features of an image while providing a fine adjustment for frequency properties [31, 36, 37]. The capability of frequency adjustment is particularly important for the reduction of the background noise in medical images. The preservation of the features of the edge is the most important thing in the noise reduction process. The 2D Gabor wavelet is defined as follows:

$$G(x, y, \theta, u, \sigma) = \frac{1}{2\pi\sigma^2} \exp\left\{-\frac{x^2 + y^2}{2\sigma^2}\right\} \exp\left\{2\pi j \left(ux\cos\theta + uy\sin\theta\right)\right\}, \quad (1)$$

where u is the frequency of the sinusoidal wave, θ adjusts the orientation of the wave, σ is the standard deviation of the Gaussian function in the x and y direction, and $j = \sqrt{-1}$. The output of the Gabor filtering can be given as a 2D convolution of the input image $I(x, y)$ and $G(x, y)$ for particular u, θ, and σ. The result is a 2D complex signal because Gabor wavelet is complex. The absolute of this signal is an image preserving the features of the edge. When the wave vector is perpendicular to the edge, the Gabor wavelets enhance the edge and remove the background information. The result image of the convolution demonstrates the local properties indicating the edge of the analyzed image [31]. Kernels related to angles are obtained by setting orientation factor.

Figure 2 shows a different representation for the 2D Gabor wavelet $G(x, y)$ with parameters $\sigma = 0.03$, $u = 0.3$ and orientation factor $\theta = 0$. Figures 2(a) and 2(c) represent the real part, and Figures 2(b) and 2(d) represent the imaginary part. Actually, Gabor wavelet is a complex wavelet with a few important oscillations relating to frequency parameter. The magnitude decay rate of the oscillations depends on the value of σ. The characteristics of 2D Gabor wavelet are particularly appropriate to extract the directional features, and the waveform is suitable to preserve the edge pixels while suppressing the noise, which is encountered in medical images.

3. Unsupervised Clustering

3.1. K-Means Clustering. A typical unsupervised clustering algorithm is k-means which is attractive in practice because it is simple and fast [38, 39]. The algorithm tries to partition the given input data into k disjoint clusters c_i. For this purpose, it searches the cluster centers by minimizing the sum of squared distances of each data point $(x_1, x_2, \ldots, x_N)$ to its nearest cluster centre c_j. The measure of the distance is commonly chosen as Euclidian distance to minimize the following mean square error (MSE) cost function:

$$C_{\text{MSE}} = \sum_{i=1}^{k} \sum_{x_j \in c_i} \left\| x_j - c_i \right\|^2, \quad (2)$$

where C_{MSE} shows the cost of an examined pixel to assign a cluster, which is the distance.

x and c are the data point and the cluster centre. It can be said that x is in cluster if $\|x_j - c_i\|$ is the minimum of all the k distances.

K-means algorithm can be summarized as follows.

Step 1. Initializations of centre location $(c_1, c_1, \ldots, c_k)$.

Step 2. Assigning each x_i to its nearest cluster centre c_k.

Step 3. Deciding the membership of each pixel to the k clusters, whose centroid is closest to that pixel.

Step 4. Setting c_i to be the centre of mass of all points in cluster C_i for all k cluster centers.

3.2. Fuzzy C-Means Clustering. The clustering process can be expressed as grouping pixels according to the similarities of their features. A clustering algorithm can provide a way of differentiating the regions in an image. Several methods based on the ideas using clustering algorithms have been proposed to partition an image into regions. As an unsupervised technique, Fuzzy c-means (FCM) clustering, was proposed by Bezdek et al. [40] as one of the widely used techniques to determine the boundaries of the objects in an image. It is considered that the reason for the high performance of FCM is due to the fact that through this process each pixel is assigned to a cluster or segment. FCM algorithm groups similar pixels according to their features because an image is represented by its features, such as histogram properties [41].

The cost function, which depends on the distance between the cluster centers and pixels, is calculated iteratively to find a minimum value. The FCM determines the clusters when the cost is minimized. The studies using FCM-based segmentation algorithms reported that FCM-based segmentation methods preserve more information than crisp and hard segmentation methods [42]. However, one of the drawbacks of FCM-based segmentation is sensitivity to noise and imaging artifacts, which is frequently encountered in medical imaging. This disadvantage is due to that fact that spatial information is not taken into account. In FCM, the cluster centers are repositioned after the calculation of an objective function used as c-means. There is flexibility in FCM because the objective function includes a membership value to a cluster [24, 41].

In FCM algorithms, each of the pixels is assigned to suitable categories by using a membership function after calculation of a cost function. The cost function is calculated iteratively to find the minimum value using Euclidean

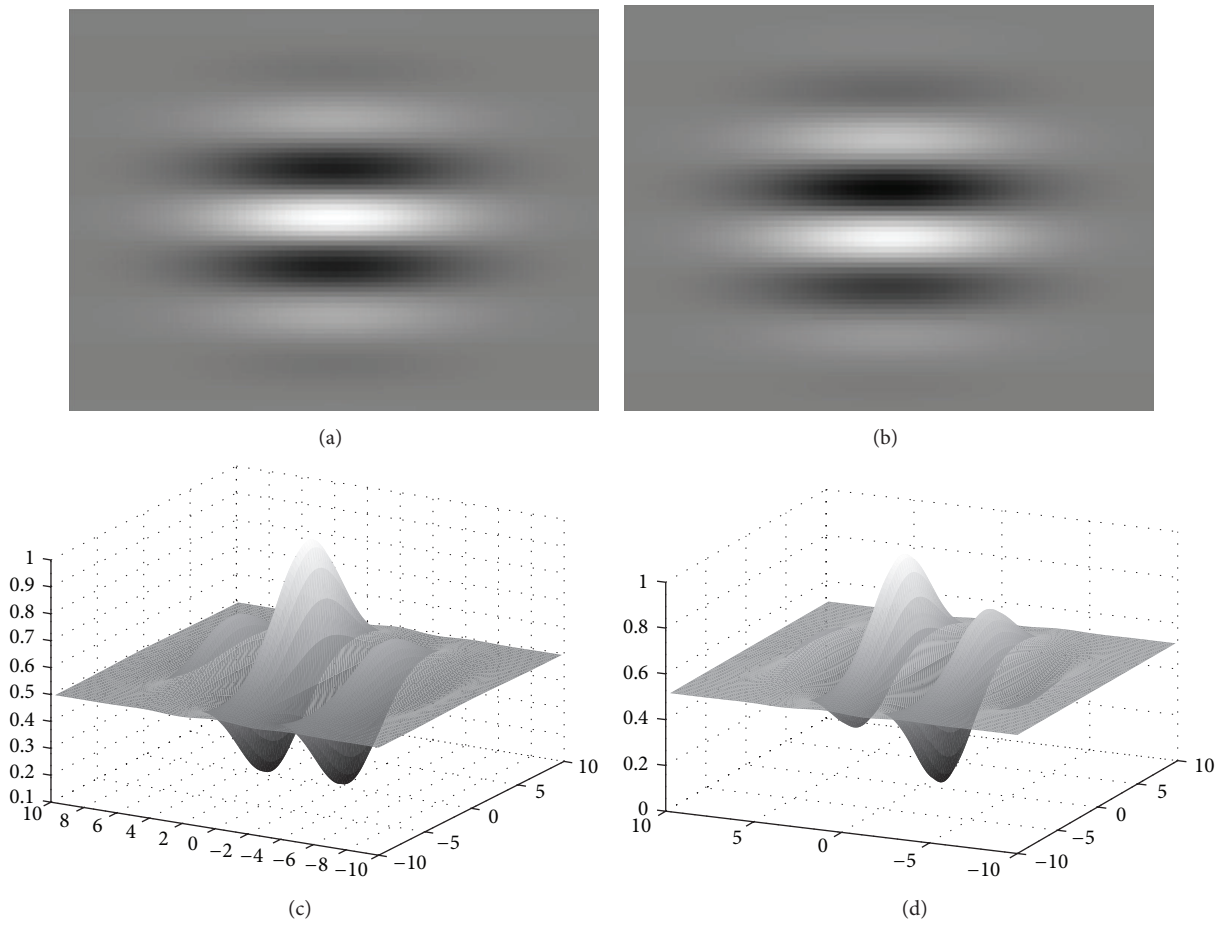


Figure 2: A two-dimensional Gabor wavelet; ((a), (c)) real part, ((b), (d)) imaginary part.

distance between the examined pixel and the centre to be assigned. This can be formulated as follows:

$$E = \sum_{j=1}^{N} \sum_{i=1}^{C} \mu_{ij}^{k} \left\| p_j - c_i \right\|, \tag{3}$$

where E shows the cost of an examined pixel to assign a cluster. μ_{ij} and c_i represent a membership of a pixel to a cluster and the cluster centre, respectively. $\|\cdot\|$ denotes absolute value operator. k is used to adjust the fuzziness as a constant value.

Here, the membership can be expressed as the probability of a pixel belonging to a cluster. This probability depends on the distance of the pixel to a cluster centre. The probability of the pixel belonging to a cluster can be calculated as follows:

$$c_i = \frac{\sum_{j=1}^{N} \mu_{ij}^{k} x_j}{\sum_{j=1}^{N} \mu_{ij}^{k}}, \tag{4}$$

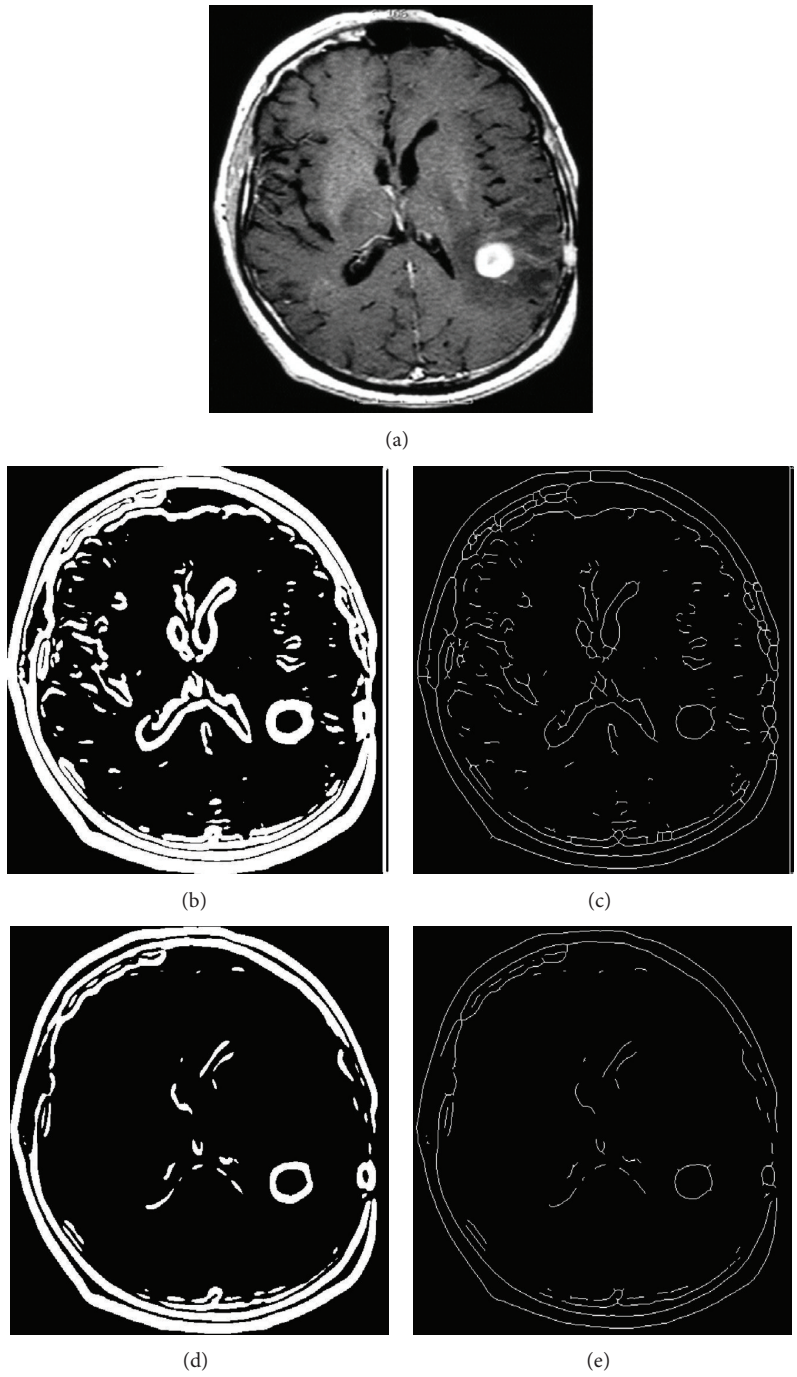
$$\mu_{ij} = \left(\sum_{m=1}^{C} \left(\frac{\left\| x_j - c_i \right\|}{\left\| x_j - c_m \right\|} \right)^{2/(k-1)} \right)^{-1}. \tag{5}$$

The FCM algorithm process these two equations iteratively.

4. Experiments

We have tested our method on several medical images acquired from CT and MRI imaging devices. The edges of the CT brain scan and abdominal images are determined using GWT, an unsupervised clustering algorithm, and morphological skeletonisation. Figure 3 represents a CT brain scan image including a tumour, clustering results, and the tissue edges. The edges are obtained in the three steps in our approach. After applying Gabor wavelet transformation to find out directional edge information, an unsupervised clustering algorithm is used to convert the gray level image into a binary image, which still contains irrelevant pixels. Then, some morphological operations are used to remove the irrelevant pixels in the binary image. As clustering algorithms, we have favored k-means and FCM clustering algorithms because they are unsupervised methods. As a morphological method, the skeletonisation is used in order to eliminate the irrelevant pixels in the binary image.

Figure 4 represents the GWT results of the brain image given in Figure 3 using four different orientations ($\pi/4$, $\pi/2$, $3\pi/4$, and π). This figure shows how the orientation of GWT plays an important role for the enhancement of the boundaries. The image in Figure 4(e) is the total resulting



(a)

(b)

(c)

(d)

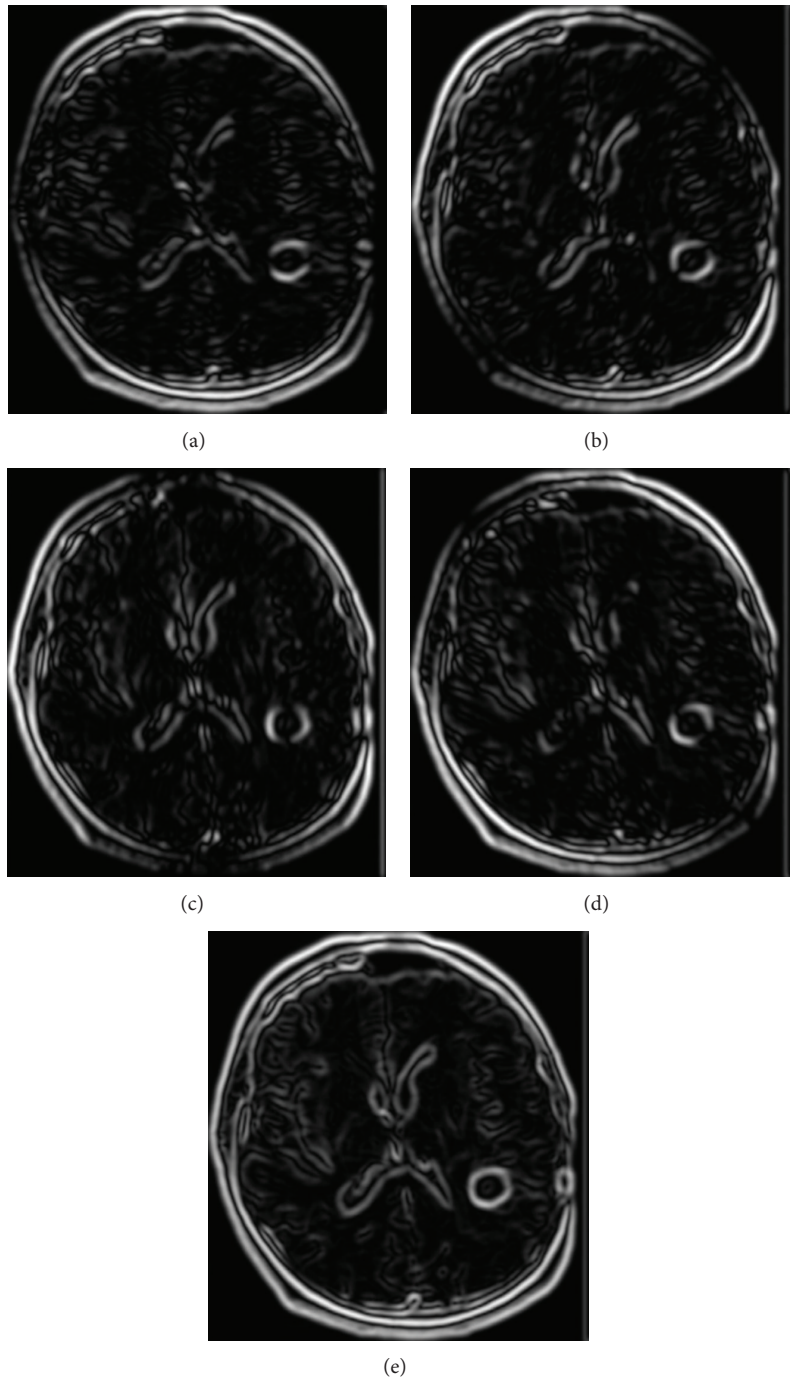
(e)

Figure 3: Edges using k-means and FCM clustering after GWT; (a) original, (b) k-means clustering, (c) skeleton of (b), (d) FCM clustering, and (e) skeleton of (d).

image of the GWT, which contains the total edge information. An unsupervised clustering method is used to convert the gray level image obtained as GWT result into a binary image. Figure 5 represents the binary image obtained using unsupervised methods and their skeletons as an example of abdominal images.

Figures 6 and 7 also represent the results of the proposed method for a CT brain scan image and an abdominal image, respectively. Because the Canny edge detection method is widely used to present the ground truth images in many applications [43], the edge detection results using the Canny method are also given in these figures to carry out a visual comparison.

In fact, it is never possible to identify the accurate edge of a real image. Although there is not a reliable methodology to put forward an appropriate ground truth edge [44], the edge of synthetic image and the manually drawn edge of a real image are used as the ground truth edge.



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

Figure 4: The GWT of a CT scan brain image for different orientation factors ($\sigma = 0.1$ and $\omega = 0.005$); (a) $\pi/4$, (b) $\pi/2$, (c) $3\pi/4$, (d) π, and (e) total result.

Another difficulty is how to define the quality parameter in order to estimate the integrity of edge detection because of application dependency. While several methods have been proposed in the literature to measure the performance of an edge detector objectively, there is no agreement on the quality parameter about edge detection. Nevertheless, the misclassification rate (MCR) and Pratt's figure of Merit (FOM) are proposed to measure the similarity between the ground truth edge and the detected edge in literature.

The MCR can be defined as follows:

$$\text{MCR} = \frac{\sum |B_A \cap B_D| + \sum |F_A \cap F_D|}{\sum (B_A + F_D)} \times 100\%, \quad (6)$$

where F_A and F_D refer to the foreground pixels of actual and detected image while B_A and B_D refer to the background pixels of the actual and the detected image. Indeed, the operation of the numerator refers to logical "exor" operation

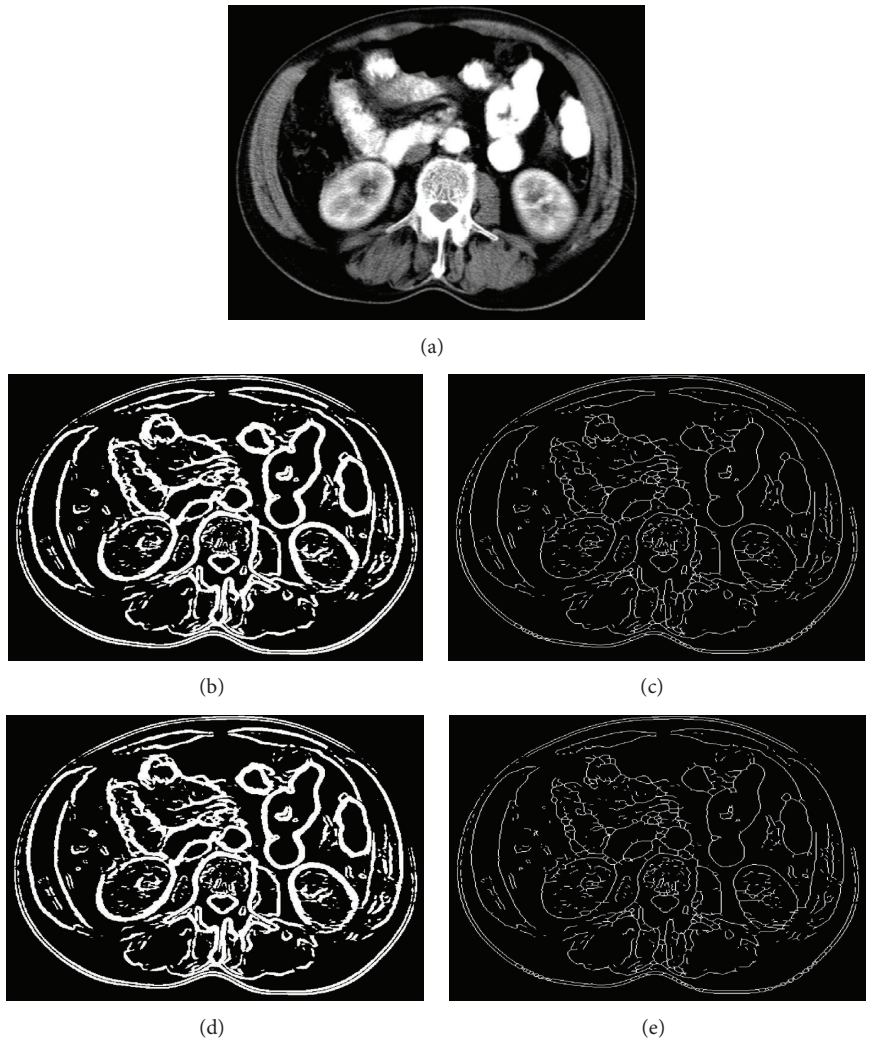


FIGURE 5: Edges of an abdominal CT image ($\sigma = 0.18$ and $\omega = 0.0005$); (a) original, (b) k-means, (c) skeleton of (b), (d) FCM, (e) skeleton of (d).

[22, 24]. The less value of MCR indicates that a good detection is done.

The definition of FOM can be given as follows:

$$\mathrm{FOM} = \frac{1}{\max\left(N_t, N_d\right)} \sum_{i=1}^{N_d} \frac{1}{1 + \alpha L(i)^2}, \tag{7}$$

where N refers to the number of edge pixels, and subscripts d and t denote the detected edge and the accurate edge, respectively. $L(i)$ shows the distance between the ith accurate edge pixel and the detected edge pixel. And parameter α is generally accepted as 1/9 as a scaling factor [43]. When an accurate edge is not detected, or a false edge is detected, or the detected edge is far from the accurate edge, FOM value increases. If the edge is perfectly detected, FOM value is 1. Otherwise, it may decrease to zero. The difference between the comparison methods, MCR and FOM, can be expressed as follows. Whereas MCR algorithms accept only the exact overlapped pixel points between the estimated edge and the accurate edge, the FOM algorithms accept the pixel points of the founded edge if they are very near to the accurate edges.

To present the performance of our proposed method, we applied both of the methodologies on a synthetic phantom image and a real image to obtain the appropriate ground truth edges. The images are ordinary CT and MRI images, and no prefiltering procedure is applied. A real and manually annotated approach was performed under the supervision of a radiologist. Figure 8(a) represents the manually ground truth edge of an abdominal image given as an example in Figure 5(a). The edges of the classical methods (Canny, Sobel, and Prewitt), which are accepted as benchmark methods in literature, are also given in Figure 8. Because many edge detection methods are proposed in the literature, we just compared to the most know methods accepted as benchmark.

To evaluate the performance of the proposed method, the MCR and FOM values were calculated using the edge given in Figure 8(a) and other methods, k-means, FCM, Prewitt, Canny, and Sobel. Table 1 represents the result of the comparison. These results prove that the performance

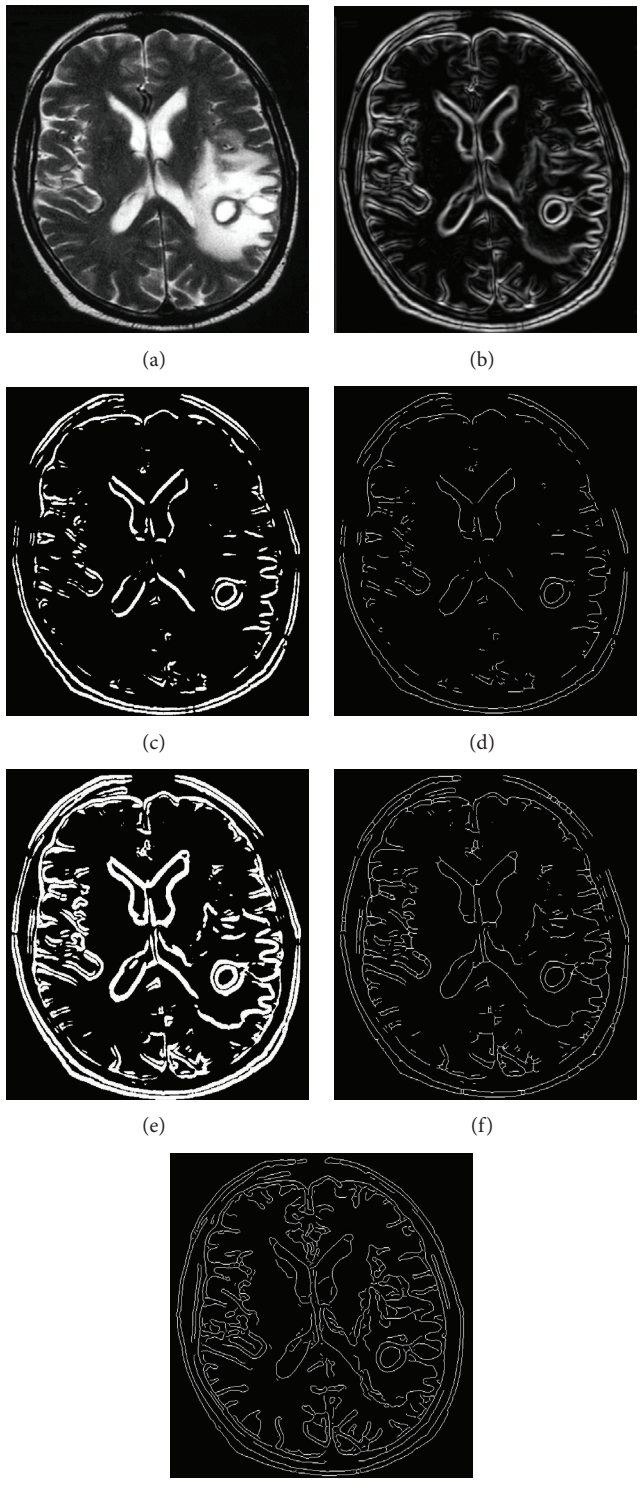
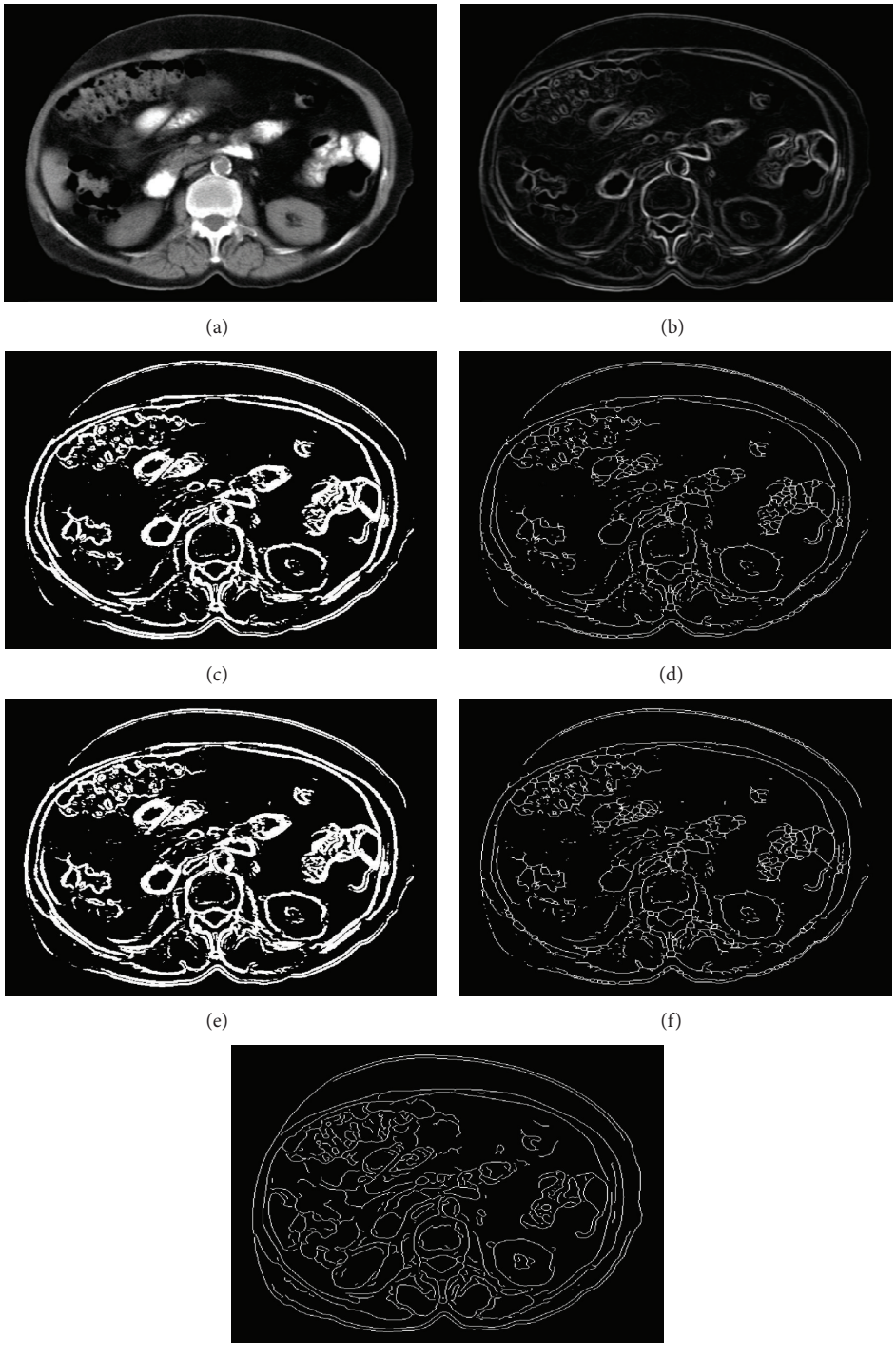


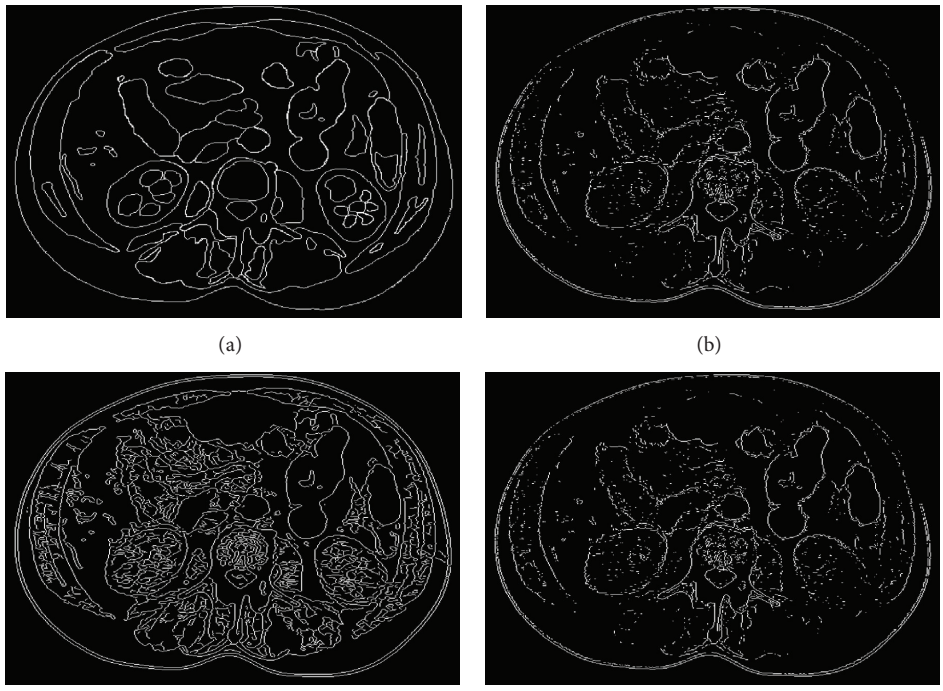
FIGURE 6: Edges of a CT scan brain image ($\sigma = 0.3$ and $\omega = 0.05$); (a) original, (b) GWT, (c) k-means, (d) skeleton of (c), (e) FCM, (f) skeleton of (e), and (g) Canny.



(a) (b) (c) (d) (e) (f) (g)

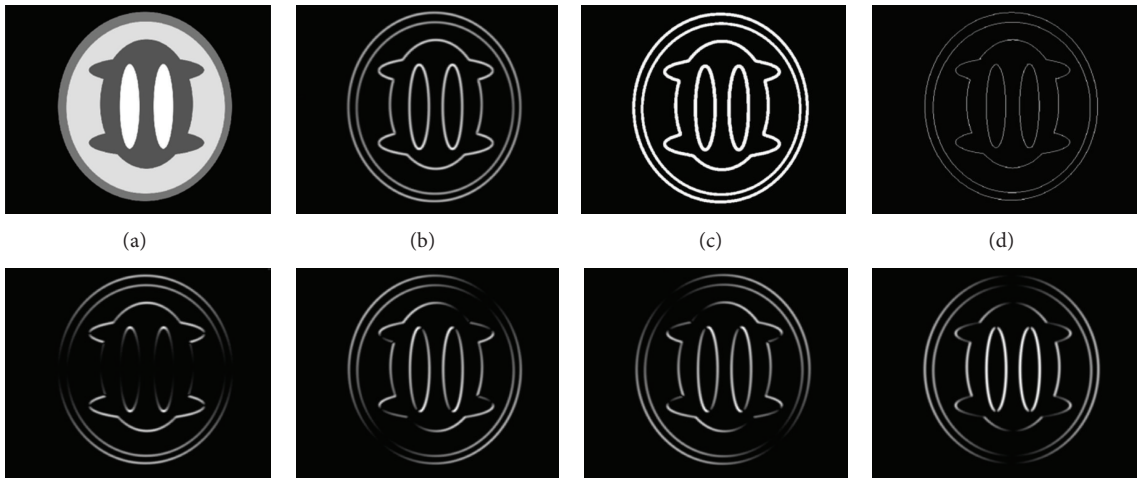
Figure 7: Edges of a CT scan abdominal image ($\sigma = 0.18$ and $\omega = 0.0005$); (a) original, (b) GWT, (c) k-means, (d) skeleton of (c), (d) FCM, (e) skeleton of (d), and (g) Canny.

of the proposed method is higher than the other methods. In particular, FCM represents the highest performance with 0.7981 of FOM values. The Canny edge detection method, which is widely used to obtain a ground truth edge, has the smallest FOM value in the table. The reason for this result is that medical images are always noisy images, and the Canny edge detection method is highly sensitive to noisy signals. K-means-based edge detection also has higher FOM value than Canny, Prewitt, and Sobel methods. According to MCR values in Table 1, FCM- and k-means-based GWT edge detection methods have smaller values, 4.8653 and 5.6980, respectively. The fact that MCR results are compatible



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

Figure 8: Edges of Figure 5(a); (a) manually drawn ground truth edge, (b) Prewitt, (c) Canny, and (d) Sobel.



(a) (b) (c) (d) (e) (f) (g) (h)

Figure 9: The simulated image and its GWT ($\sigma = 0.24$ and $\omega = 0.0055$); (a) original phantom image, (b) GWT, (c) k-means, (d) skeleton of (c), (e) $\pi/4$, (f) $\pi/2$, (g) $3\pi/4$, and (h) π.

with FOM results supports that GWT-based edge detection methods have higher performances.

As stated, it is very difficult to find the edge of a medical image corrupted by noisy signals. Medical images are generally low-density and noisy images depending on the type of imaging device. Furthermore, no medical imaging device can operate independently of the noise. Therefore, the resistance of an edge detection method to noise should be taken into consideration, particularly in medical imaging. In order to measure the resistance to noisy signals, we have performed another experiment using a synthetic phantom image given in Figure 9(a). Why we use the phantom image and its noisy form is to make a more objective assessment.

The phantom image is constituted using several overlapping ellipses. It is assumed each ellipse indicates to a tissue. The accurate edge accepted as the ground truth edge was determined when the phantom image was constituted. The MCR and FOM values are computed for the phantom image

Table 1: MCR and FOM results for the manually ground truth edge in Figure 8(a).

MCR results					FOM results				
k-means	FCM	Prewitt	Canny	Sobel	k-means	FCM	Prewitt	Canny	Sobel
5.6980	4.8653	6.1733	9.3037	6.2557	0.7195	0.7981	0.7010	0.5020	0.7094

Table 2: MCR and FOM results for the phantom image in Figure 9.

PSNR	MCR results					FOM results				
	k-means	FCM	Prewitt	Canny	Sobel	k-means	FCM	Prewitt	Canny	Sobel
35.01	0.9384	0.9285	0.8934	0.7954	0.8968	0.9710	0.9710	0.9723	0.9752	0.9722
30.01	0.9182	0.9350	0.9007	16.8981	0.8938	0.9712	0.9709	0.9721	0.1902	0.9723
25.02	0.9548	0.9563	0.9102	18.1576	0.9243	0.9700	0.9701	0.9614	0.0765	0.9622
23.00	1.2249	1.3115	2.9491	21.2528	2.9976	0.9429	0.8958	0.4788	0.0658	0.4690
23.00*	1.2196	1.3355	2.0824	7.6271	2.0962	0.9351	0.8985	0.7296	0.1831	0.7236

*Preprocessed using median filter.

to conclude an evaluation in order to measure the performance of edge detection methods on the phantom image. The results given in Table 2 show that the proposed algorithm can detect the edges more precisely than the classical method even in the presence of noise. It can be considered that better results have been yielded by the proposed method because the GWT acts as a gradient operator while suppressing noise. The last row ($*$) shows the results of the edge detection methods after applying a median filter on the noisy phantom image. The filter operation is applied to the noisy phantom image only when using classical methods. No filter operation is applied to the phantom image when using the GWT-based methods. Even in this case, the proposed methods give higher FOM values and smaller MCR values, as seen in Table 2. This last result proves that GWT is successful even in noisy cases.

The FOM values of the proposed method may change slightly with respect to the trail number because k-means and FCM clustering methods choose the centroid arbitrarily. According to the results of the phantom image, it is observed that the FCM clustering algorithm is more sensitive than the k-means clustering algorithm. Nonetheless, the proposed method using k-means and FCM is very successful compared to other methods.

5. Conclusion

This work presents two methods of edge detection based on the GWT. These two proposed methods use k-means and FCM clustering method to convert a gray level image into a binary image. The main idea of the proposed method is to integrate the information obtained from the GWT at a different orientation and to incorporate the use of a clustering method. The effect of the GWT can be seen on the regional boundaries of the given image. The GWT enhances the edge information and suppresses the noisy signals in a given image. The tests prove that both methods have a great performance particularly in noisy conditions.

In this paper, we have proposed two kinds of edge detection methods based on GWT. Other methods using GWT could be conceptually explored and adapted. The results showed that the directional information from GWT provides a competitive advantage for edge analysis and detection. Since GWT uses three parameters, sigma, frequency, and orientation, it can be adapted for application-dependent images.

Conflict of Interests

The author wishes to confirm that there is no known conflict of interests associated with this paper and there has been no significant financial support for this work that could have influenced its outcome. The author confirms that the paper has been read and approved and that there are no other persons who satisfied the criteria for authorship and are not listed.

Acknowledgment

The author would like to thank Dr. Murat Baykara for sharing medical images and for giving advice. The author also confirms that he has given due consideration to the protection of intellectual property associated with this work and that there are no impediments to publication, including the timing of publication, with respect to intellectual property. In doing so, the author confirms that he has followed the regulations of his institution concerning intellectual property. He further confirms that any aspect of the work covered in this paper that has involved either experimental animals or human patients has been conducted with the ethical approval of all relevant bodies and that such approvals are acknowledged within the paper. The author understands that the corresponding author is the sole contact for the editorial process (including editorial manager and direct communications with the office). He/she is responsible for communicating with the other authors about progress, submissions of revisions, and final approval of proofs. The author confirms that he has provided a current, correct email address which is accessible by the corresponding author.

References

[1] Z. Ma, J. M. R. S. Tavares, R. N. Jorge, and T. Mascarenhas, "A review of algorithms for medical image segmentation and their applications to the female pelvic cavity," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 13, no. 2, pp. 235–246, 2010.

[2] R.-R. Jorge and B.-C. Eduardo, "Medical image segmentation, volume representation and registration using spheres in the geometric algebra framework," *Pattern Recognition*, vol. 40, no. 1, pp. 171–188, 2007.

[3] Q. Abbas, M. E. Celebi, and I. F. Garćıa, "Breast mass segmentation using region-based and edge-based methods in a 4-stage multiscale system," *Biomedical Signal Processing and Control*, vol. 8, no. 2, pp. 204–214, 2013.

[4] X. Lu, Y. Sun, and Y. Yuan, "Optimization for limited angle tomography in medical image processing," *Pattern Recognition*, vol. 44, no. 10-11, pp. 2427–2435, 2011.

[5] Z. Zhang, S. Ma, H. Liu, and Y. Gong, "An edge detection approach based on directional wavelet transform," *Computers and Mathematics with Applications*, vol. 57, no. 8, pp. 1265–1271, 2009.

[6] A. Fathi and A. R. Naghsh-Nilchi, "Automatic wavelet-based retinal blood vessels segmentation and vessel diameter estimation," *Biomedical Signal Processing and Control*, vol. 8, no. 6, pp. 71–80, 2013.

[7] P. Coupé, J. V. Manjón, E. Gedamu, D. Arnold, M. Robles, and D. L. Collins, "Robust Rician noise estimation for MR images," *Medical Image Analysis*, vol. 14, no. 4, pp. 483–493, 2010.

[8] M. Ikeda, R. Makino, K. Imai, M. Matsumoto, and R. Hitomi, "A method for estimating noise variance of CT image," *Computerized Medical Imaging and Graphics*, vol. 34, no. 8, pp. 642–650, 2010.

[9] Y. Dai and G. L. Niebur, "A semi-automated method for hexahedral mesh construction of human vertebrae from CT scans," *Computer Methods in Biomechanics and Biomedical Engineering*, vol. 12, no. 5, pp. 599–606, 2009.

[10] H. Masoumi, A. Behrad, M. A. Pourmina, and A. Roosta, "Automatic liver segmentation in MRI images using an iterative watershed algorithm and artificial neural network," *Biomedical Signal Processing and Control*, vol. 7, no. 5, pp. 429–437, 2012.

[11] Q. He and Z. Zhang, "A new edge detection algorithm for image corrupted by White-Gaussian noise," *AEU—International Journal of Electronics and Communications*, vol. 61, no. 8, pp. 546–550, 2007.

[12] Z. Dokur and T. Ölmez, "Tissue segmentation in ultrasound images by using genetic algorithms," *Expert Systems with Applications*, vol. 34, no. 4, pp. 2739–2746, 2008.

[13] Z.-X. Ji, Q.-S. Sun, and D.-S. Xia, "A framework with modified fast FCM for brain MR images segmentation," *Pattern Recognition*, vol. 44, no. 5, pp. 999–1013, 2011.

[14] T. Chen, Q. H. Wu, R. Rahmani-Torkaman, and J. Hughes, "A pseudo top-hat mathematical morphological approach to edge detection in dark regions," *Pattern Recognition*, vol. 35, no. 1, pp. 199–210, 2002.

[15] S. Anand, R. S. S. Kumari, S. Jeeva, and T. Thivya, "Directionlet transform based sharpening and enhancement of mammographic X-ray images," *Biomedical Signal Processing and Control*, vol. 8, pp. 391–399, 2013.

[16] J.-Y. Zhang, C. Yan, and X.-X. Huang, "Edge detection of images based on improved sobel operator and genetic algorithms," in *Proceedings of the International Conference on Image Analysis and Signal Processing (IASP '09)*, pp. 31–35, April 2009.

[17] Z. Iscan, A. Yüksel, Z. Dokur, M. Korürek, and T. Ölmez, "Medical image segmentation with transform and moment based features and incremental supervised neural network," *Digital Signal Processing*, vol. 19, no. 5, pp. 890–901, 2009.

[18] V. Grau, A. U. J. Mewes, M. Alcañiz, R. Kikinis, and S. K. Warfield, "Improved watershed transform for medical image segmentation using prior information," *IEEE Transactions on Medical Imaging*, vol. 23, no. 4, pp. 447–458, 2004.

[19] M. A. Balafar, A. R. Ramli, M. I. Saripan, and S. Mashohor, "Medical image segmentation using fuzzy C-mean (FCM), bayesian method and user interaction," in *Proceedings of the International Conference on Wavelet Analysis and Pattern Recognition (ICWAPR '08)*, pp. 68–73, August 2008.

[20] T. Cerciello, P. Bifulco, M. Cesarelli, and A. Fratini, "A comparison of denoising methods for X-ray fluoroscopic images," *Biomedical Signal Processing and Control*, vol. 7, no. 6, pp. 550–559, 2012.

[21] W. Cai, S. Chen, and D. Zhang, "Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation," *Pattern Recognition*, vol. 40, no. 3, pp. 825–838, 2007.

[22] K.-S. Chuang, H.-L. Tzeng, S. Chen, J. Wu, and T.-J. Chen, "Fuzzy c-means clustering with spatial information for image segmentation," *Computerized Medical Imaging and Graphics*, vol. 30, no. 1, pp. 9–15, 2006.

[23] L. O. Hall, A. M. Bensaid, L. P. Clarke, R. P. Velthuizen, M. S. Silbiger, and J. C. Bezdek, "A comparison of neural network and fuzzy clustering techniques in segmenting magnetic resonance images of the brain," *IEEE Transactions on Neural Networks*, vol. 3, no. 5, pp. 672–682, 1992.

[24] T. Chaira, "A novel intuitionistic fuzzy C means clustering algorithm and its application to medical images," *Applied Soft Computing Journal*, vol. 11, no. 2, pp. 1711–1717, 2011.

[25] L. Zhang and P. Bao, "Edge detection by scale multiplication in wavelet domain," *Pattern Recognition Letters*, vol. 23, no. 14, pp. 1771–1784, 2002.

[26] B. Ergen, "Signal and image denoising using wavelet transform," in *Advances in Wavelet Theory and Their Applications in Engineering, Physics and Technology*, D. Baleanu, Ed., pp. 495–514, 2012.

[27] A. Thakur and R. S. Anand, "Speckle reduction in ultrasound medical images using adaptive filter based on second order statistics," *Journal of Medical Engineering and Technology*, vol. 31, no. 4, pp. 263–279, 2007.

[28] Y. Yue, M. M. Croitoru, A. Bidani, J. B. Zwischenberger, and J. W. Clark Jr., "Nonlinear multiscale wavelet diffusion for speckle suppression and edge enhancement in ultrasound images," *IEEE Transactions on Medical Imaging*, vol. 25, no. 3, pp. 297–311, 2006.

[29] C. Zhu, J. Ni, Y. Li, and G. Gu, "Speckle noise suppression techniques for ultrasound images," in *Proceedings of the 4th International Conference on Internet Computing for Science and Engineering (ICICSE '09)*, pp. 122–125, December 2009.

[30] W. D. Richard and C. G. Keen, "Automated texture-based segmentation of ultrasound images of the prostate," *Computerized Medical Imaging and Graphics*, vol. 20, no. 3, pp. 131–140, 1996.

[31] W. Jiang, K.-M. Lam, and T.-Z. Shen, "Efficient edge detection using simplified Gabor wavelets," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 39, no. 4, pp. 1036–1047, 2009.

[32] L. Shen and L. Bai, "A review on Gabor wavelets for face recognition," *Pattern Analysis and Applications*, vol. 9, no. 2-3, pp. 273–292, 2006.

[33] X. Pan and Q.-Q. Ruan, "Palmprint recognition using Gabor-based local invariant features," *Neurocomputing*, vol. 72, no. 7-9, pp. 2040–2045, 2009.

[34] "Kernel based multi-object tracking using gabor functions embedded in a region covariance matrix," in *Pattern Recognition and Image Analysis*, vol. 5524 of *Lecture Notes in Computer Science*, pp. 72–79, 2009.

[35] D. Shen, Y. Zhan, and C. Davatzikos, "Segmentation of prostate boundaries from ultrasound images using statistical shape model," *IEEE Transactions on Medical Imaging*, vol. 22, no. 4, pp. 539–551, 2003.

[36] W. Gu, C. Xiang, Y. V. Venkatesh, D. Huang, and H. Lin, "Facial expression recognition using radial encoding of local Gabor features and classifier synthesis," *Pattern Recognition*, vol. 45, no. 1, pp. 80–91, 2012.

[37] J. V. B. Soares, J. J. G. Leandro, R. M. Cesar Jr., H. F. Jelinek, and M. J. Cree, "Retinal vessel segmentation using the 2-D Gabor wavelet and supervised classification," *IEEE Transactions on Medical Imaging*, vol. 25, no. 9, pp. 1214–1222, 2006.

[38] K. R. Žalik, "An efficient k′-means clustering algorithm," *Pattern Recognition Letters*, vol. 29, no. 9, pp. 1385–1391, 2008.

[39] P. J. Herrera, G. Pajares, and M. Guijarro, "A segmentation method using Otsu and fuzzy k-Means for stereovision matching in hemispherical images from forest environments," *Applied Soft Computing Journal*, vol. 11, no. 8, pp. 4738–4747, 2011.

[40] J. C. Bezdek, R. Ehrlich, and W. Full, "FCM: the fuzzy c-means clustering algorithm," *Computers and Geosciences*, vol. 10, no. 2-3, pp. 191–203, 1984.

[41] Z. Ji, Y. Xia, Q. Chen, Q. Sun, D. Xia, and D. D. Feng, "Fuzzy c-means clustering with weighted image patch for image segmentation," *Applied Soft Computing Journal*, vol. 12, no. 6, pp. 1659–1667, 2012.

[42] Y. Li and G. Li, "Fuzzy C-means cluster segmentation algorithm based on modified membership," in *Advances in Neural Networks: ISNN 2009*, vol. 5552 of *Lecture Notes in Computer Science*, pp. 135–144, 2009.

[43] S. Yi, D. Labate, G. R. Easley, and H. Krim, "A shearlet approach to edge analysis and detection," *IEEE Transactions on Image Processing*, vol. 18, no. 5, pp. 929–941, 2009.

[44] N. L. Fernández-García, A. Carmona-Poyato, R. Medina-Carnicer, and F. J. Madrid-Cuevas, "Automatic generation of consensus ground truth for the comparison of edge detection techniques," *Image and Vision Computing*, vol. 26, no. 4, pp. 496–511, 2008.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 593503, 7 pages
http://dx.doi.org/10.1155/2014/593503

Research Article

Evolutionary Approach for Relative Gene Expression Algorithms

Marcin Czajkowski and Marek Kretowski

Faculty of Computer Science, Bialystok University of Technology, Wiejska 45a, 15-351 Białystok, Poland

Correspondence should be addressed to Marek Kretowski; m.kretowski@pb.edu.pl

Received 6 December 2013; Accepted 24 February 2014; Published 23 March 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

A Relative Expression Analysis (RXA) uses ordering relationships in a small collection of genes and is successfully applied to classiffication using microarray data. As checking all possible subsets of genes is computationally infeasible, the RXA algorithms require feature selection and multiple restrictive assumptions. Our main contribution is a specialized evolutionary algorithm (EA) for top-scoring pairs called EvoTSP which allows finding more advanced gene relations. We managed to unify the major variants of relative expression algorithms through EA and introduce weights to the top-scoring pairs. Experimental validation of EvoTSP on public available microarray datasets showed that the proposed solution significantly outperforms in terms of accuracy other relative expression algorithms and allows exploring much larger solution space.

1. Introduction

Extracting accurate and simple rules that exploit marker genes is crucial in understanding and identifying casual relationships between specific genes. Finding a meaningful and robust classification rule is a real challenge; especially when in different studies of the same cancer, diverse genes are considered to be marked [1, 2].

A Relative Expression Analysis (RXA) was firstly proposed by Geman et al. in [3] and represents simple yet powerful set of classifiers. It is based on the relative orderings among the expressions of a small number of genes. Instead of using expression values directly, only ranks of the expression data are used, making the algorithms insensitive to data normalization procedures. Moreover, use of the ordering relationships for a small collection of genes has potential for identification of gene-gene interactions with plausible biological interpretation and direct clinical applicability [4]. Major and well-known drawback of RXA is a high computational complexity, which grows exponentially with the size of the collection of genes.

In this paper, we propose an Evolutionary Top-Scoring Pairs (EvoTSP) solution that combines the power of evolutionary approach with simplicity of relative expression algorithms. We managed to unify different top-scoring extensions, limit their restrictions, and with application of EA explore larger solution space. We have also changed the unweighted TSP voting, by introducing the weights of each gene pair.

The rest of the paper is organized as follows. In the next section the relative expression algorithms are briefly recalled. Section 3 describes our motivation and Section 4 presents in detail the EvoTSP solution. Next, experimental validation on real-life microarray datasets is performed. The paper is concluded in the last section where possible future works are also sketched.

2. Background

The first and the most popular solution from RXA is called Top-Scoring Pair (TSP) [3]. It is based on pairwise comparisons of gene expression values. Discrimination between two classes depends on finding one pair of genes that achieves the highest ranking value called "score."

Consider a gene expression microarray dataset consisting of P genes and M samples. Let the data be represented as a $P \times M$ matrix in which an expression value of ith gene from jth sample is denoted as x_{ij}. Each row represents observation of a particular gene over M training samples, and each column represents a gene expression instance composed from P genes. Let us for the simplicity of presentation assume

that there are only two classes, C_1 and C_2, and instances with indexes from 1 to M_1 ($M_1 < M$) that belong to the first class (C_1) and instances from range $\langle M_1+1, M\rangle$ to the second class (C_2).

The TSP method focuses on gene pair matching (i, j) $(i, j \in \{1, \ldots, P\}, i \neq j)$ for which there is the highest difference in probability p of an event $x_{im} < x_{jm}$ $(m = 1, 2, \ldots, M)$ between class C_1 and C_2. For each pair of genes (i, j) two probabilities are calculated, $p_{ij}(C_1)$ and $p_{ij}(C_2)$:

$$p_{ij}(C_1) = \frac{1}{|C_1|}\sum_{m=1}^{M_1} I\left(x_{im} < x_{jm}\right),$$
$$p_{ij}(C_2) = \frac{1}{|C_2|}\sum_{m=M_1+1}^{M} I\left(x_{im} < x_{jm}\right), \quad (1)$$

where $|C_1|$ denotes the number of instances from class C_1 and $I(x_{im} < x_{jm})$ is the indicator function defined as

$$I\left(x_{im} < x_{jm}\right) = \begin{cases} 1, & \text{if } x_{im} < x_{jm} \\ 0, & \text{if } x_{im} \geq x_{jm}. \end{cases} \quad (2)$$

TSP is a rank-based method; therefore, for each pair of genes (i, j) the "score" denoted Δ_{ij} is calculated as

$$\Delta_{ij} = \left|p_{ij}(C_1) - p_{ij}(C_2)\right|. \quad (3)$$

In the next step, the algorithm chooses a pair with the highest score. There should be only one top pair in the TSP method; however, it is possible that multiple gene pairs achieve the same top score. In that case a secondary ranking proposed in [5] is used to eliminate draws. It is based on the rank differences in classes and samples.

In the literature, the TSP solution is extended in several directions, each having its pros and cons. In one of the first extensions called k-TSP [5] the number of top-scoring pairs included in the final prediction was increased. The classifier uses no more than k top scoring disjoint gene pairs that have the highest score. The parameter k is determined by the internal cross-validation and the simple majority vote is used to make the final decision.

Different approach for the TSP extension is discussed in [4] where authors instead of using several pairs of genes compare relationships for three genes. A three-gene version of RXA called Top-Scoring Triplet (TST) [4] was proposed as potentially more discriminating than TSP since there are six possible orderings that must be analyzed. With the TST solution authors successfully predict the germline BRCA1 mutations in breast cancer. This method was later extended in [6] where general idea of pairwise or triplet rank comparisons was proposed. The top-scoring N (TSN) algorithm uses generic permutations and dynamically adjusts the size to control both the permutation and combination space available for classification. Variable N denotes the size of the classifier; therefore, in case $N = 2$ the TSN algorithm simply reduces to the TSP method and when $N = 3$, the TSN can be seen as TST. The classifier's size can be defined by user or by internal cross-validation that checks classification accuracy for different values of N (on a training data, in a range specified by the user) and selects the classifier with the highest score.

A hybrid solution of k-TSP and a top-down induced decision tree is proposed in [7]. In each node of the decision tree called TSPDT a test analogous to the k-TSP method is searched. Then, the set of instances is divided according to decision of the best pair (or pairs) of genes in the current node and next; each derived subset goes to the corresponding branch. The process is recursively repeated for each branch until leaf node is reached. This solution was recently extended by global induction of decision tree called GTSPDT [8]. Preliminary experiments showed that this hierarchical evolutionary method can also be a good alternative to traditional relative expression algorithms.

Figure 1 illustrates the extensions of the relative expression algorithms. We can observe that EvoTSP unifies two main extensions of the TSP solution: application of multiple pairs of genes instead of one and comparison relationships for more than two genes.

There exist other solutions in RXA like Weight k-TSP [9] which focuses on the ratio of two genes in order to find more accurate top-scoring pairs. Different look at ranking the genes in microarray classification was also proposed in [10].

The RXA can be used as a feature selection in more complex classifiers [11–13] and as a protein expression classifier [14]. Multiple implementations of TSP-family solutions may be found as *R* package [15] or as a stand-alone application [16].

3. Motivation

The first drawback of RXA is the enormous computational requirement as the complexity of aforementioned algorithms is $O(k * P^N)$, where k is the number of top-scoring groups, P is the number of features, and N is the size of group of genes with which ordering relationships are compared. In the literature, there are some attempts of improving TSP performance by parallelization of the algorithm using graphic processing unit (GPU) for calculations [17]. Although the improvement is significant, the parameter k or/and N still must be small—the highest tested value of N equals 4 with $k = 1$ and only when P was significantly reduced by the feature selection. This illustrates how computationally demanding RXA is.

Finding accurate values of the parameters k and N is the second problem. The TSP extensions define them ad hoc or by internal cross-validation. The first way is strongly dependent on analyzed dataset and the second one is extremely time consuming and decreases the size of the training dataset which is usually very small in case of microarray data. In addition, it is not clear which extension should be preferred: k-TSP or TSN. It should be noted that the k-TSP algorithm cannot be replaced by the TST as k-TSP has restrictions to use only disjoint gene pairs. On the other side, the k-TST or k-TSN was not even analyzed in the literature, probably due to its computational complexity.

In this paper, we would like to limit aforementioned drawbacks of TSP extensions through the evolutionary

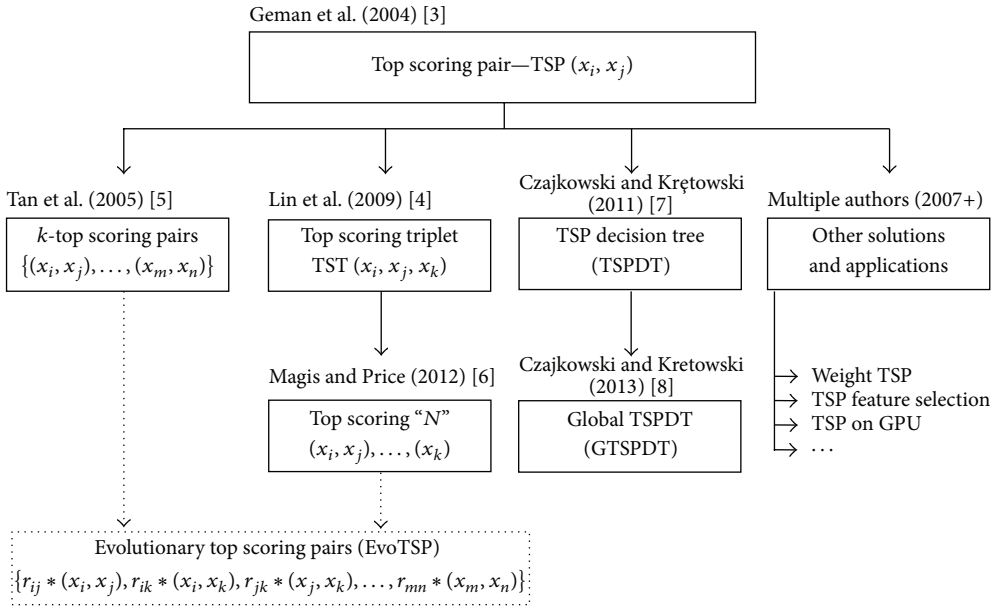


Figure 1: Evolution of the relative expression algorithms.

approach. Our goal is to improve classification accuracy and identification of marker genes interactions. We let the EA to search for the best multiple pairwise comparisons of the gene expression values. The number of top-scoring pairs k is determined also by the evolution and with no restrictions on disjoint gene pairs; EvoTSP may compare relationships for more than two genes like in TSN. Application of EA to the RXA allows exploring larger solution space with reasonable computation time.

4. Evolutionary Top-Scoring Pairs

In this section, we would like to propose EvoTSP—an evolutionary algorithm for top-scoring pairs. Evolutionary algorithms [18] belong to a family of metaheuristic methods which represent techniques for solving a wide variety of difficult optimization problems. The general framework of EA (see Figure 2) is inspired by biological mechanisms of evolution. The algorithm operates on a population of individuals and each individual represents a candidate solution to the target problem. Individuals are assessed using a quality measure named the fitness function which measures their performance and those with higher fitness are usually more often selected for reproduction. Genetic operators such as mutation and crossover modify new generations of individuals, producing new offspring. This guided random search (offspring usually inherits some traits from its ancestors) is stopped when some convergence criteria are satisfied.

4.1. Representation and Initialization. Each individual is represented in its actual form as a potential solution. It is composed of a group of k top-scoring pairs similarly to k-TSP. As there are no restrictions on disjoint gene pairs, the EvoTSP

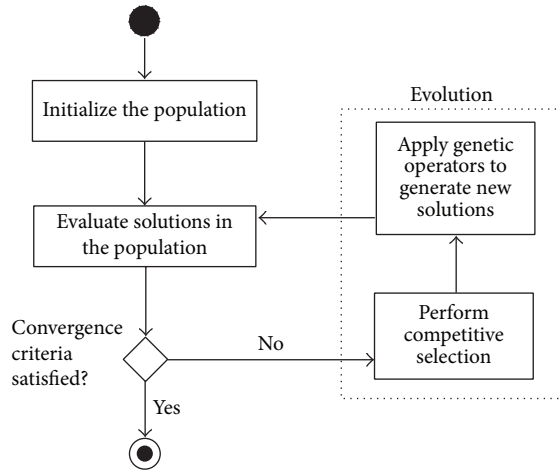


Figure 2: A general framework of evolutionary algorithm.

is able to represent the TST solution with the 2 top-scoring pairs that involve only three genes. In the analogous way, TSN, k-TSP, or even variations of k-TSN can be represented in EvoTSP.

In this paper, we also propose additional parameter r for each pair of genes that represents its weight. This way, some gene pairs have higher influence than others on the final decision. This idea is completely new in TSP as aforementioned algorithms used a simple majority voting where each top-scoring pair's vote has the same weight. The purpose of using unweighted voting in TSP and all its extensions was probably directed by the necessity of limiting computational requirements. Figure 3 shows an example EvoTSP model, which includes possible representation of k-TSP and the TST solution.

EvoTSP

Weight	Gene pairs	
r_1	$(x_i > x_j)$	} = TST*: $(x_i > x_j > x_t)$
r_2	$(x_j > x_t)$	} = k-TSP*: $\{(x_j > x_t), (x_q > x_w)\}$
r_3	$(x_q > x_w)$	
⋮	⋮	
r_k	$(x_n \leq x_m)$	

*: Pairs are equivalent only if weights = 1

Figure 3: An example representation of EvoTSP model.

We could generate initial population randomly to cover the entire range of possible solutions; however, due to the large solution space, we decided to speed up evolutionary search and seed initial population with good solutions (default number of individuals in population equals 100).

Each initial individual has a random number of gene pairs ($0 < k \leq 5$) created with the mixed dipole strategy [19] and constructed as follows. Among feature vectors located in the node two objects from different classes are randomly chosen. Next, an effective top-scoring pair is constructed with 2 randomly selected genes. By the effective top-scoring pair, we understand the pair of genes which separates two objects from different classes. In other words, genes i and j can constitute effective top-scoring pair only if there are at least two instances q and w that are from different classes and one of the relations is satisfied:

$$x_{iq} > x_{jq}, \qquad x_{iw} \leq x_{jw}, \tag{4}$$

or the opposite:

$$x_{iq} \leq x_{jq}, \qquad x_{iw} > x_{jw}. \tag{5}$$

This operation is repeated until k pairs is selected. All created gene pairs have equal weights (parameter $r_i = 1$ where $i = 1, \ldots, k$). With this strategy we are able to limit the number of initial individuals which select only one class.

4.2. Fitness Function. Fitness function is one of the most important and sensitive elements in the design of the evolutionary algorithm. It drives the evolutionary search process by measuring how good a single individual is in terms of meeting the problem objective. Direct minimization of prediction error measured on the learning set usually results in overfitting and leads to spurious results.

In case of EvoTSP, we need to balance the error of classification and the number of genes that build the classifier. We have applied a similar idea that was used in the cost complexity pruning in the CART system [20]. The fitness function is maximized and has the following form:

$$\text{Fitness} = Q_{\text{Reclass}} - \alpha * (2 * k + u), \tag{6}$$

where Q_{Reclass} is the reclassification quality on the training set, k is the number of gene pairs, and u is the number of unique genes in top-scoring pairs that were used to build the classifier. The α parameter is the relative importance of the complexity term specified by user (default value is 0.005). Penalty associated with the classifier complexity increases proportionally with the number of genes that constitute the top-pairs. To reduce overfitting and to encourage searching relation between more than two genes, unique genes are doubly penalized. It should be noticed that there is no optimal value of α for all possible datasets and tuning it may improve classifier results for specific problem. Further research on setting this parameter automatically on a particular training data is planned.

4.3. Genetic Operators. To maintain genetic diversity, two specialized genetic operators corresponding to the classical cross-over and mutation were applied. Each evolutionary iteration starts with selecting individuals from the population that will be affected by the genetic operators. Probability of applying a cross-over operator equals 0.5 for each individual. With the same probability a mutation operator can also be applied. Next, one of the variants of genetic operator is selected.

We propose two variants of recombination:

(i) a randomly chosen pair of genes is exchanged between two affected individuals. Probability of pairs to exchange equals 0.9;

(ii) a randomly chosen pair from the best individual founded so far replaces a random pair from the affected individual. In this variant only one individual is modified and the probability of this variant equals 0.1.

If the mutation operator is chosen, one of the variants with equal probability of being drawn is applied to the individual:

(i) add a new pair of genes created with the mixed dipole strategy;

(ii) remove randomly chosen pair;

(iii) replace randomly chosen pair by the new one created with the mixed dipole strategy;

(iv) exchange one feature from randomly chosen pair;

(v) increase/decrease the weight of the randomly chosen pair (by multiplying or dividing by 2);

(vi) switch the relation sign among randomly chosen pair.

4.4. Selection and Termination Condition. Ranking linear selection [18] is applied as a selection mechanism. In each iteration, a single individual with the highest value of fitness function in current population is copied to the next one *(elitist strategy)*. In addition, this strategy is partially boosted by possible cross-over of individuals from current population with the best individual founded so far. Evolution terminates when fitness of the best individual in the population does not improve during fixed number of generations (default value: 1000). In case of a slow convergence, maximum number of generations is also specified (default value: 10000), which allows us to limit the computation time.

Table 1: Details of tested gene expression datasets.

Datasets	Number of features	Number of instances
GDS2771	22215	192
GSE10072	22284	107
GSE17920	54676	130
GSE19804	54613	120
GSE25837	18631	93
GSE27272	24526	183
GSE3365	22284	127
GSE6613	22284	105

5. Results and Discussions

In this section, all performed experiments are presented. At first, we share some details about datasets and settings of tested algorithms. Next, we validate and discuss the overall performance of EvoTSP solution and its competitors with respect to classification accuracy and its size.

5.1. Datasets and Setup. Performance of classifiers was investigated on several public available microarray datasets deposited in NCBI's Gene Expression Omnibus [21] and summarized in Table 1. All datasets are binary classification problems and mainly refer to the studies of human cancer. As the data was not predivided we used typical 10-fold cross-validation as it was the only option in AUREA software [16].

We confront EvoTSP with three competitors: the primary solution TSP and its two main extensions: k-TSP and TST (TSN with $N = 3$). To obtain comparison results, we used the AUREA software, which is an open-source system for identification of relative expression molecular signatures [16]. Classification was performed with default parameters for all algorithms through all datasets and to ensure stable results average score of 20 runs is shown. A statistical analysis of all obtained results was performed with the Friedman test and the corresponding Dunn's multiple comparison test (significance level equal to 0.05) as recommended by Demšar [22].

The AUREA software sets the maximum number of top-scoring pairs (parameter k) for k-TSP to 10 by default. In addition, all algorithms except EvoTSP operate on a subset of genes for analysis based on the differential expression of the presented gene set (the Wilcoxon signed-rank test was used to choose the most differentially expressed genes between the defined classes). Authors [16] state that this feature selection step have dramatic effect on the computational complexity of the algorithms and by limiting the set of genes, problem of over-fitting can be mitigated. In case of EvoTSP we have decided not to use any feature selection and allow searching for relations through all high and low-ranked genes.

5.2. Comparison of Top-Scoring Family Algorithms Methods. Table 2 summaries classification performance for the proposed solution EvoTSP and its competitors: TSP, TST, and k-TSP. The model size of TSP and TST is not shown as it is fixed and equals correspondingly 2 and 3. We had to use approximation of k-TSP size as AUREA software did not allow checking the k value during cross-validation; therefore, the value of k on full dataset treated as a training set is presented.

Results show that, in general, the existing extensions, TST and k-TST, outperform TSP in terms of accuracy. The price for better performance is the higher complexity of the classification model, which for k-TSP is 5.75 times higher (an average value from 8 datasets) than TSP size and almost 4 times than TST. Slightly larger size of classification model is not a problem, as all tested algorithms are simple to analyze; however, checking several different genes per model may be considered difficult in biological interpretation, which is the case for k-TSP.

In the last two columns of Table 2 we present the results of the proposed solution. We can observe that the accuracy of the classifier in 6 out of 8 datasets is the highest. However, for the last two databases EvoTSP accuracy score is slightly lower than k-TSP. Additional experiments showed that the convergence of EA in EvoTSP is too slow for that particular set. When the maximum number of generation in EA was increased, the proposed algorithm managed to have similar or even outperform k-TSP on both datasets.

According to the Friedman test, there is a statistically significant difference (P value of 0.0003) in the accuracy of all versions. Based on Dunn's Multiple Comparison Test Difference, there is a statistically significant difference in classification quality between EvoTSP, TSP, and the TST algorithm. Although there were no statistical differences in accuracy between EvoTSP and k-TSP, there is one in the size of their models. The size of classification model of proposed solution remains small, in contrast to k-TSP, making the EvoTSP a good tool for identifying gene-gene interactions with direct clinical applicability. In Table 2 we can also observe that the standard deviation of accuracy for solutions was on similar level.

Total time to build an EvoTSP model varies between 1 and 8 minutes on a typical PC (Intel Core I5, 4 GB RAM), depending on the dataset and it is few times longer than for AUREA software which was from tens of seconds to a minute. However, it should be noted that EvoTSP works without any feature selection which is a must for AUREA software (checking of all combinations of pairs would take many orders of magnitude more).

6. Conclusion

In this paper, we propose the EvoTSP system for solving classification problems using microarray data. Our approach is a hybrid solution that combines the power of EA and relative expression algorithms. We have designed several variants of specialized operators to mutate and cross-over individuals and a fitness function that helps mitigating the overfitting problem. With the new weighted gene pairs voting and extended representation of top-scoring pairs that involve different variants of TSP, we were able to significantly improve TSP accuracy with still relatively small size of classification

Table 2: Comparison of top-scoring algorithms, including accuracy with its standard deviation and the number of unique genes that build classifier's model.

Datasets	TSP	TST	k-TSP		EvoTSP	
	Accuracy	Accuracy	Accuracy	Size	Accuracy	Size
GDS2771	57.2 ± 2.4	61.9 ± 2.8	62.9 ± 3.3	10	65.6 ± 2.0	4.0
GSE10072	88.7 ± 2.6	89.4 ± 2.1	90.1 ± 2.5	6	96.5 ± 1.3	2.1
GSE17920	64.9 ± 3.5	63.7 ± 4.7	67.2 ± 3.2	10	78.1 ± 2.6	2.8
GSE19804	93.5 ± 1.7	92.8 ± 1.5	94.1 ± 1.6	10	96.2 ± 1.1	2.1
GSE25837	56.0 ± 4.0	60.5 ± 5.1	58.4 ± 4.0	14	66.9 ± 5.6	3.1
GSE27272	47.3 ± 4.8	50.1 ± 3.8	56.2 ± 2.2	18	66.2 ± 1.1	2.7
GSE3365	81.9 ± 2.6	84.2 ± 2.7	87.2 ± 2.1	14	86.1 ± 2.8	4.1
GSE6613	49.5 ± 3.5	51.7 ± 2.8	55.8 ± 5.3	10	53.6 ± 5.4	6.1
Average	67.4 ± 3.3	69.3 ± 3.2	71.5 ± 3.0	11.5	76.2 ± 2.7	3.4

model. Application of EAs allows exploring much larger solution space and searching for different, more complex relations between genes.

In this paper we only focus on the general concept of EvoTSP as an effective tool; therefore, we do not enclose any biological aspects of the rules generated by proposed system or case studies on particular datasets. Furthermore improvement is still required especially in terms of fitness functions to handle cost-sensitive and multiclass problems. Speeding up the convergence of the EA is also desirable and can be achieved by application of local optimizations (memetic algorithms), new specialized operators, and self-adaptive parameters. Finally, more work on preprocessing datasets, gene selection, and using additional problem-specific knowledge is also required to improve EvoTSP classification accuracy and rule discovery.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Grant S/WI/2/13 from Bialystok University of Technology. The authors thank John C. Earls for his help with AUREA software.

References

[1] P. S. Nelson, W. G. Nelson, A. V. D'Amico, N. Rosen, and M. A. Rubin, "Predicting prostate cancer behavior using transcript profiles," *Journal of Urology*, vol. 172, no. 5, pp. S28–S33, 2004.

[2] M. Logotheti, O. Papadodima, N. Venizelos, A. Chatziioannou, and F. Kolisis, "A comparative genomic study in schizophrenic and in bipolar disorder patients, based on microarray expression proling meta-analysis," *The Scientic World Journal*, vol. 2013, Article ID 685917, 14 pages, 2013.

[3] D. Geman, C. D'Avignon, D. Q. Naiman, and R. L. Winslow, "Classifying gene expression profiles from pairwise mRNA comparisons," *Statistical Applications in Genetics and Molecular Biology*, vol. 3, no. 1, article 19, 2004.

[4] X. Lin, B. Afsari, L. Marchionni et al., "The ordering of expression among a few genes can provide simple cancer biomarkers and signal BRCA1 mutations," *BMC Bioinformatics*, vol. 10, article 1471, p. 256, 2009.

[5] A. C. Tan, D. Q. Naiman, L. Xu, R. L. Winslow, and D. Geman, "Simple decision rules for classifying human cancers from gene expression profiles," *Bioinformatics*, vol. 21, no. 20, pp. 3896–3904, 2005.

[6] A. T. Magis and N. D. Price, "The top-scoring 'N' algorithm: a generalized relative expression classification method from small numbers of biomolecules," *BMC Bioinformatics*, vol. 13, no. 1, p. 227, 2012.

[7] M. Czajkowski and M. Krętowski, "Top scoring pair decision tree for gene expression data analysis," *Advances in Experimental Medicine and Biology*, vol. 696, pp. 27–35, 2011.

[8] M. Czajkowski and M. Kretowski, "Global top-scoring pair decision tree for gene expression data analysis," in *Proceedings of the 16th European conference on Genetic Programming (EuroGP '13)*, pp. 229–240, 2013.

[9] M. Czajkowski and M. Kretowski, "Novel extension of k-TSP algorithm for micro-array classification," in *Proceedings of Proceedings of the 21st international conference on Industrial, Engineering and Other Applications of Applied Intelligent Systems: New Frontiers in Applied Artificial Intelligence (IEA-AIE '08)*, pp. 456–465, LNAI, 2008.

[10] J. H. Phan, A. N. Young, and M. D. Wang, "Robust microarray meta-analysis identifies differentially expressed genes for clinical prediction," *The Scientic World Journal*, vol. 2012, Article ID 989637, 9 pages, 2012.

[11] S. Yoon and S. Kim, "K-Top Scoring Pair Algorithm for feature selection in SVM with applications to microarray data classification," *Soft Computing*, vol. 14, no. 2, pp. 151–159, 2010.

[12] P. Shi, S. Ray, Q. Zhu, and M. A. Kon, "Top scoring pairs for feature selection in machine learning and applications to cancer outcome prediction," *BMC Bioinformatics*, vol. 12, article 375, 2011.

[13] H. Zhang, H. Wang, Z. Dai, M. Chen, and Z. Yuan, "Improving accuracy for cancer classification with a new algorithm for genes selection," *BMC Bioinformatics*, vol. 12, no. 298, 2012.

[14] P. Kau, D. Schlatzer, K. Cooke, and M. R. Chance, "Pairwise protein expression classifier for candidate biomarker discovery for early detection of human disease prognosis," *BMC Bioinformatics*, vol. 13, no. 191, 2012.

[15] J. T. Leek, "The tspair package for finding top scoring pair classifiers in R," *Bioinformatics*, vol. 25, no. 9, pp. 1203–1204, 2009.

[16] J. C. Earls, J. A. Eddy, C. C. Funk, Y. Ko, A. T. Magis, and N. D. Price, "AUREA: an open-source software system for accurate and user-friendly identification of relative expression molecular signatures," *BMC Bioinformatics*, vol. 14, no. 78, 2013.

[17] A. T. Magis, J. C. Earls, Y.-H. Ko, J. A. Eddy, and N. D. Price, "Graphics processing unit implementations of relative expression analysis algorithms enable dramatic computational speedup," *Bioinformatics*, vol. 27, no. 6, Article ID btr033, pp. 872–873, 2011.

[18] Z. Michalewicz, *Genetic Algorithms + Data Structures = Evolution Programs*, Springer, 3rd edition, 1996.

[19] M. Kretowski and M. Grzes, "Evolutionary induction of mixed decision trees," *International Journal of Data Warehousing and Mining*, vol. 3, no. 4, pp. 68–82, 2007.

[20] L. Breiman and J. Friedman, *Classification and Regression Trees*, Wadsworth, 1984.

[21] R. Edgar, M. Domrachev, and A. E. Lash, "Gene Expression Omnibus: NCBI gene expression and hybridization array data repository," *Nucleic Acids Research*, vol. 30, no. 1, pp. 207–210, 2002.

[22] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine Learning Research*, vol. 7, pp. 1–30, 2006.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 536434, 12 pages
http://dx.doi.org/10.1155/2014/536434

Research Article

New Fuzzy Support Vector Machine for the Class Imbalance Problem in Medical Datasets Classification

Xiaoqing Gu, Tongguang Ni, and Hongyuan Wang

School of Information Science and Engineering, Changzhou University, Changzhou 213164, China

Correspondence should be addressed to Hongyuan Wang; tiddyddd@163.com

Received 25 November 2013; Accepted 20 February 2014; Published 23 March 2014

Academic Editors: V. Bhatnagar and Y. Zhang

In medical datasets classification, support vector machine (SVM) is considered to be one of the most successful methods. However, most of the real-world medical datasets usually contain some outliers/noise and data often have class imbalance problems. In this paper, a fuzzy support machine (FSVM) for the class imbalance problem (called FSVM-CIP) is presented, which can be seen as a modified class of FSVM by extending manifold regularization and assigning two misclassification costs for two classes. The proposed FSVM-CIP can be used to handle the class imbalance problem in the presence of outliers/noise, and enhance the locality maximum margin. Five real-world medical datasets, breast, heart, hepatitis, BUPA liver, and pima diabetes, from the UCI medical database are employed to illustrate the method presented in this paper. Experimental results on these datasets show the outperformed or comparable effectiveness of FSVM-CIP.

1. Introduction

Computer techniques such as machine learning and pattern recognition have been widely adopted by modern medicine. One reason is that an enormous amount of data has to be gathered and analyzed which is very hard or even impossible without making use of computer techniques. The other reason is that computer techniques have led toward digital analysis of pathological diagnosis, automatic classification differentiating, and detecting diseases. In some cases, an early symptom of some diseases is lighter and gives no obvious pointer to a possible diagnosis; moreover, many symptoms look very similar to each other, though they are caused by different diseases. So it may be difficult even for experienced doctors to make correct diagnosis. Therefore, an automatic classification system can help doctor diagnose accurately, assess disorders remotely and evaluate the treatment process [1].

In recent years, researchers have proposed a lot of approaches for medicine classification, such as neural network, Bayesian network, and support vector machine (SVM). Among them SVM is considered to be one of the most successful ones [2]. For example, to improve time and accuracy in differentiating diffuse interstitial lung disease for computer-aided quantification, a hierarchical SVM is introduced which shows promise for various real-time and online image-based classification applications in clinical fields [3]. SVM as a classifier is used for liver disorders and its correct classification rate is highly successful compared to the other results attained [4]. A two-stage approach is proposed for medical datasets classification, in which the artificial bee colony algorithm is used for feature selection and SVM is used for classification [5].

The support vector machine (SVM) proposed by Vapnik [6, 7] is a novel approach for solving pattern recognition problems. SVM maps the sample points into a high-dimensional feature space to seek for an optimal separating hyperplane through maximizing the margin between two classes. In addition, SVM is a quadratic programming (QP) problem that assures that its solution is obtained once it is the global unique solution, and the sparsity of solution assures better generalization. However, most of the real-world medical datasets usually contain some outliers and noisy examples. The classical SVM is very sensitive to outliers/noise. To solve this problem, fuzzy support vector machine (FSVM) [8] is proposed, in which each sample is given a fuzzy membership that denotes the attitude of the corresponding point toward

one class. The membership represents how important the sample is to the decision surface.

Nevertheless, many medical datasets are composed of "normal" samples with only a small percentage of "abnormal" ones, which leads to the so-called class imbalance problems. FSM does not take into consideration the class distribution and can be sensitive to the class imbalance problem. As a result, the hyperplane of FSVM can be skewed towards the minority class, and this skewness can degrade the performance of FSVM with respect to the minority class. To tackle this problem, Veropoulos et al. [9] have proposed a method called different error costs (DEC), where the SVM objective function has been modified to assign two different misclassification cost values. It is noticed that One-Class Classification [10, 11] is sometimes used in novelty detection, and it only uses the normal training data. However, in many real medical datasets, abnormal examples exist, although they are very few. Furthermore, in classification tasks, the scatter matrix can play an important role when incorporated with local intrinsic geometry structures of samples [12]. Some methods have been recently proposed to incorporate the structure of the data distribution into SVM. A linear manifold learning method named locality preserving projection (LPP) is proposed in [13, 14], which aims at preserving the local manifold structure of the samples space. Although LPP considers enhancing the local data compactness with each manifold, it does not separate manifolds with different class labels.

In this paper, we propose a new FSVM method for the class imbalance problem (FSVM-CIP) which can be used to address both the problem of class imbalance and outliers/noise. FSVM-CIP not only considers the fuzziness of each training sample but also extends manifold regularization and maximizes the localized relative margin. It takes the positive samples and negative samples into consideration with different misclassification costs according to their unbalanced distributions. We systematically evaluated the FSVM-CIP on five real-world medical datasets and compared its performance with four different SVM methods for classification. The results showed that the proposed method can improve the classification accuracy and handle the classification problems with outliers/noise and imbalanced datasets more effectively.

The rest of this paper is organized as follows. Section 2 briefly reviews the related works. Section 3 presents the details of FSVM-CIP in the linear case. Section 4 presents FSVM-CIP in the nonlinear case in detail. The experimental results on five medical datasets are reported in Section 5, and some concluding remarks are given in Section 6.

2. Related Works

2.1. Fuzzy Support Vector Machines (FSVMs). In traditional SVM, all the data points are considered with equal importance and assigned the same penal parameter in its objective function. However, in many real-world classification applications, some sample points, such as the outliers or noises, may not be exactly assigned to one of these two classes, and each sample point does not have the same meaning to the decision surface. To solve this problem, the theory of fuzzy support vector machine was originally proposed in [8]. Fuzzy membership to each sample point is introduced such that different sample points can make different contributions to the construction of decision surface.

Suppose the training samples are

$$S = \left\{\left(\mathbf{x}_i, y_i, s_i\right),\ i = 1, \ldots, N\right\}, \tag{1}$$

where $\mathbf{x}_i \in \mathbf{R}^n$ is the n-dimension sample point, $y_i \in \{-1, +1\}$ represents its class label, and s_i $(i = 1, \ldots, N)$ is a fuzzy membership which satisfies $\sigma \le s_i \le 1$ with a sufficiently small constant $\sigma > 0$. The quadratic optimization problem for classification is considered as follows:

$$\begin{aligned} \min_{\mathbf{w},s,\xi} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} + C\sum_{i-1}^{l} s_i\xi_i \\ \text{s.t.} \quad & y_i\left(\mathbf{w}^T\mathbf{x}_i + b\right) \ge 1 - \xi_i, \quad \xi_i \ge 0,\ i = 1, \ldots, l, \end{aligned} \tag{2}$$

where $\mathbf{w}$ is a normal vector of the separating hyperplane, b is a bias term, and C is a parameter which has to be determined beforehand to control the tradeoff between the classification margin and the cost of misclassification error. Since s_i is the attitude of the corresponding point $\mathbf{x}_i$ towards one class and the slack variables ξ_i are a measure of error, then the term $s_i\xi_i$ can be considered a measure of error with different weights. It is noted that the bigger the s_i is, the more importantly the corresponding point is treated; the smaller the s_i is, the less importantly the corresponding point is treated; thus, different input points can make different contributions to the learning of decision surface. Therefore, FSVM can find a more robust hyperplane by maximizing the margin by letting some misclassification of less important points.

In order to solve the FSM optimal problem, (2) is transformed into the following dual problem by introducing Lagrangian multipliers α_i:

$$\begin{aligned} \max_{\alpha} \quad & \sum_{i=1}^{N}\alpha_i - \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\alpha_i\alpha_j y_i y_j \mathbf{x}_i\mathbf{x}_j \\ \text{s.t.} \quad & \sum_{i=1}^{N} y_i\alpha_i = 0, \quad 0 \le \alpha_i \le s_i C,\ i = 1, \ldots, N. \end{aligned} \tag{3}$$

Compared with the standard SVM, the above statement only has a little difference, which is the upper bound of the values of α_i. By solving this dual problem in (3) for optimal α_i, $\mathbf{w}$ and b can be recovered in the same way as in the standard SVM.

2.2. Locality Preserving Projections (LPP). Locality preserving projection (LPP) [13, 14] is a linear dimensionality reduction algorithm by feature extraction or projection. It builds an adjacency graph incorporating neighborhood information of the data set using the Laplacian graph and then computes a transformation matrix which maps the data points into a subspace. This linear transformation optimally preserves local neighborhood information in a certain sense. The representation map generated by this method can be

viewed as a linear discrete approximation to a continuous map that naturally arises from the geometry of the manifold.

For a set $X = \{\mathbf{x}_i\}$ ($i \in [1, N]$), let $N_k(\mathbf{x}_i)$ denote k nearest neighbors of node i, and let G denote the adjacency graph of dataset X. Here, the ith node corresponds to the data point x_i and nodes i and j are connected by an edge if node i is among the k nearest neighbors of node j or if node j is among the k nearest neighbors of node i; that is, $\mathbf{x}_i \in N_k(\mathbf{x}_j)$ or $\mathbf{x}_j \in N_k(\mathbf{x}_i)$. The adjacency graph G can be weighed as follows:

$$W_{ij} = \begin{cases} \exp\left(\dfrac{-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2}{t}\right) & \text{if } \mathbf{x}_i \in N_k\left(\mathbf{x}_j\right) \\ & \text{or } \mathbf{x}_j \in N_k\left(\mathbf{x}_i\right) \\ 0 & \text{otherwise,} \end{cases} \tag{4}$$

where $\exp(-\|\mathbf{x}_i - \mathbf{x}_j\|^2/t)$ is called the heart kernel function and t is a constant. $\|\mathbf{x}_i - \mathbf{x}_j\|$ is the Euclidean distance in $\mathbf{R}^n$ between point i and point j. LPP tries to find the transformation vector $\mathbf{w} \in \mathbf{R}^n$ by minimizing the following objective function:

$$\begin{aligned} \min_{\mathbf{w} \neq 0} \quad & \mathbf{w}^T \mathbf{X} \mathbf{L} \mathbf{X}^T \mathbf{w} \\ \text{s.t.} \quad & \mathbf{w}^T \mathbf{X} \mathbf{D} \mathbf{X}^T \mathbf{w} = 1, \end{aligned} \tag{5}$$

where $\mathbf{D}$ is a diagonal matrix whose entries are column sum of $\mathbf{W}$ and $D_{ii} = \sum_j W_{ij}$ normalizes each weight. $\mathbf{L} = \mathbf{D} - \mathbf{W}$ is the Laplacian matrix. The transformation vector $\mathbf{w}$ in the objective function in (5) is given by the minimum eigenvalue solution to the generalized eigenvalue problem. LPP preserves the intrinsic geometry and local structure of the data by minimizing the objective function.

3. FSVM for the Class Imbalance Problem in the Linear Case

In this section, we first define the local within-class preserving scatter matrix in the linear case. Secondly, the optimization problem formulation of FSVM-CIP in the linear case is given. Moreover, the fuzzy membership functions for linear FSVM-CIP are defined. Finally, the algorithm of linear FSVM-CIP is summarized.

3.1. The Local within-Class Preserving Scatter Matrix in the Linear Case. Following the idea of [15], we build the nearest within-class neighbor graph to model intrinsic geometry and local structure of the data. The graph preserves local neighborhood information in a certain sense and it can be viewed as a linear discrete approximation to a continuous map that naturally arises from the geometry of the manifold.

Considering the fact that we have a binary classification problem, one class denoted as C_1 contains sample points $\mathbf{x}_i$ with $y_i = 1$ and the other class denoted as C_2 contains sample points $\mathbf{x}_i$ with $y_i = -1$. Set $|C_1| = m_1$ and $|C_2| = N - m_1$, and the total number of sample points is N.

Definition 1. For each data $\mathbf{x}_i$, suppose its k nearest within-class neighbors set $N_k(\mathbf{x}_i)$ and an edge is put between $\mathbf{x}_j$ and its neighbors. The corresponding weight matrix W_{ij} is

$$W_{ij} = \begin{cases} \dfrac{1}{D_{ii}} \exp\left(\dfrac{-\left\|\mathbf{x}_i - \mathbf{x}_j\right\|^2}{t}\right) & \text{if } \mathbf{x}_i \in N_k\left(\mathbf{x}_j\right) \\ & \text{or } \mathbf{x}_j \in N_k\left(\mathbf{x}_i\right), y_i = y_j \\ 0 & \text{otherwise,} \end{cases} \tag{6}$$

where $D_{ii} = \sum_j W_{ij}$ normalizes each weight.

Definition 2. The local within-class preserving scatter matrix

$$\begin{aligned} & \mathbf{S}_{lw} \\ & = \sum_{k=1}^{2} \sum_{\mathbf{x}_i \in C_k} \left(\mathbf{x}_i - \sum_{\mathbf{x}_i \in N(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N(\mathbf{x}_i)} W_{ij} \mathbf{x}_j\right) \\ & \qquad \times \left(\mathbf{x}_i - \sum_{\mathbf{x}_i \in N(\mathbf{x}_j) \text{ or } \mathbf{x}_j \in N(\mathbf{x}_i)} W_{ij} \mathbf{x}_j\right)^T \\ & = \sum_{k=1}^{2} \mathbf{X}^{(k)} \left(\mathbf{I}^{(k)} - \mathbf{W}^{(k)}\right)^T \left(\mathbf{I}^{(k)} - \mathbf{W}^{(k)}\right) \mathbf{X}^{(k)T}, \end{aligned} \tag{7}$$

where $\mathbf{I}^{(k)}$ is an $N_k \times N_k$ diagonal matrix. In this case, the obtained nearest within-class neighbor graph attempts to preserve the local structure of the data set and $(\mathbf{I}^{(k)} - \mathbf{W}^{(k)})^T(\mathbf{I}^{(k)} - \mathbf{W}^{(k)})$ preserves locality of nearby points with same class label in the embedding space during the unfolding process of nonlinear structures [15]. In fact, a heavy penalty is applied to the objective function through the weight W_{ij} if the neighboring data $\mathbf{x}_i$ and $\mathbf{x}_j$ are mapped far apart. Hence, the minimization criterion is an attempt to ensure points y_i and y_j close to each other as well as $\mathbf{x}_i$ and $\mathbf{x}_j$ being close.

It is worthwhile to note that the local within-class scatter matrix $\mathbf{S}_{lw}$ is symmetric and positive semidefinite. $\mathbf{S}_{lw}$ looks similar to the within-class scatter matrix $\mathbf{S}_w$ [16, 17] and the Laplacian matrix $\mathbf{L}$ in LPP. However, $\mathbf{S}_{lw}$ reflects the intrinsic geometry and local structure of the data, and $\mathbf{S}_w$ only considers the mean value of samples in different classes. $\mathbf{S}_{lw}$ carries the class label information and discriminating information but $\mathbf{L}$ only considers the information of nearest neighbors for each data point in the input space, without considering the class labels.

3.2. FSVM-CIP in the Linear Case. To tackle the imbalance classification problem with noise and outliers, we integrate FSVM, the ideas of imbalance classification problem, and the local within-class preserving scatter. On one hand, as shown in Figure 1, the linear classifier presented by the hyperplane is $(\mathbf{w}^T\mathbf{x} + b = 0)$ and defines a field for majority-class examples $(\mathbf{w}^T\mathbf{x}+b > 1-\xi)$ and another field for minority-class examples $(\mathbf{w}^T\mathbf{x} + b > -(1 + \rho - \xi))$ which is used to weaken the skewness towards the minority class and enhance the locality

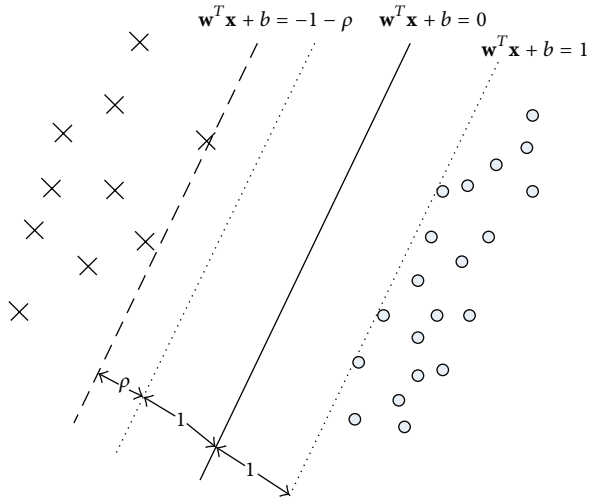


Figure 1: The hyperplanes of linear FSVM-CIP.

maximum margin. On the other hand, by assigning a higher misclassification cost for the minority class examples than the majority class examples, the effect of class imbalance could be reduced. In addition, to minimize the amount of misclassifications, the local within-class scatter matrix $\mathbf{S}_{lw}$ is used to preserve intrinsic geometry and local structure of the data.

Due to this, we define the primal problem of FSVM-CIP as follows:

$$\begin{aligned}
\min_{w,b,\rho,\xi} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} - \nu\rho \\
& + \frac{1}{\nu_1 m_1}\sum_{i=1}^{m_1}\mu_i\xi_i + \frac{1}{\nu_2 m_2}\sum_{j=m_1+1}^{N}\mu_j\xi_j + \frac{\eta}{2}\mathbf{w}^T\mathbf{S}_{lw}\mathbf{w} \\
\text{s.t.} \quad & \mathbf{w}^T\mathbf{x}_i + b \geq 1 - \xi_i, \quad i = 1, \ldots, m_1 \\
& -\left(\mathbf{w}^T\mathbf{x}_j + b\right) \geq 1 + \rho - \xi_j, \quad j = m_1 + 1, \ldots, N \\
& \xi_k \geq 0, \quad k = 1, \ldots, N, \ \rho \geq 0,
\end{aligned} \tag{8}$$

where m_1, m_2 denote the number of positive (normal class or majority class) and negative (abnormal class or minority class) training points, and $m_2 = N - m_1$. ρ is a nonnegative number, and $\rho + 1$ is the margin between the hyperplane and the minority class examples. η is a nonnegative regulation constant which is the tradeoff between the local within-class scatter and the margin. Variables ν_1, ν_2 are positive penalty parameters, which tune penalty cost of the training error for positive and negative training data, respectively. $\xi_i, \xi_j \geq 0$ are the slack variables, and μ_i, μ_j are fuzzy memberships for two-class examples.

Obviously, $\mathbf{w}^T\mathbf{S}_{lw}\mathbf{w}$ provides prior geometrical information into the penalty terms based on manifold regularization. Minimizing $\mathbf{w}^T\mathbf{S}_{lw}\mathbf{w}$ means that close data originally in the same class in the input space are likely to be close in the output place. Therefore, $\mathbf{w}^T\mathbf{S}_{lw}\mathbf{w}$ aims to preserve the local information of the manifold structure.

It is noted that, in FSVM-CIP, we assign different fuzzy membership values for training examples to reflect their different classes of importance. We also showed that it is similar to assign different misclassification costs $\mu_i/\nu_1 m_1 (\mu_j/\nu_2 m_2)$ for different training examples. In order to reduce the effect of class imbalance, we can assign higher membership values μ_j or lower parameter ν_2 for the minority class examples, while we assign lower membership values μ_i or higher ν_1 for the majority class. That is, our proposed method would not tend to skew the separating hyperplane towards the minority class examples as the minority class examples are now assigned with a higher misclassification cost. By means of setting $\mu_i/\nu_1 m_1 (\mu_j/\nu_2 m_2)$ and extending manifold regularization, the learned optimal separating hyperplane enhances the relative maximum margin and FSVM-CIP will be less sensitive to imbalanced class problems.

Then, we transform this problem into its corresponding dual problem as follows.

The primal Lagrangian is

$$\begin{aligned}
& L\left(\mathbf{w}, b, \rho, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, s\right) \\
& = \frac{1}{2}\mathbf{w}^T\mathbf{w} - \nu\rho + \frac{1}{\nu_1 m_1}\sum_{i=1}^{m_1}\mu_i\xi_i + \frac{1}{\nu_2 m_2}\sum_{j=m_1+1}^{N}\mu_j\xi_j \\
& \quad + \frac{\eta}{2}\mathbf{w}^T\mathbf{S}_{lw}\mathbf{w} - \sum_{i=1}^{m_1}\alpha_i\left(\mathbf{w}^T\mathbf{x}_i + b - 1 + \xi_i\right) \\
& \quad + \sum_{j=m_1+1}^{N}\alpha_j\left(\mathbf{w}^T\mathbf{x}_j + b + 1 + \rho - \xi_j\right) - \sum_{i=1}^{N}\gamma_i\xi_i - s\rho,
\end{aligned} \tag{9}$$

with Lagrangian multipliers $\alpha_i \geq 0$, $\gamma_i \geq 0$, and $s \geq 0$. The derivatives of $L(\mathbf{w}, b, \rho, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\gamma}, s)$ with respect to the primal variables using the Karush-Kuhn-Tucker (KKT) conditions should vanish. Consider

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N}\alpha_i y_i = 0, \tag{10}$$

$$\frac{\partial L}{\partial \mathbf{w}} = \mathbf{I}\mathbf{w} + \eta\mathbf{S}_{lw}\mathbf{w} - \sum_{i=1}^{N}\alpha_i y_i \mathbf{x}_i = 0, \tag{11}$$

$$\frac{\partial L}{\partial \rho} = -\nu + \sum_{j=m_1+1}^{N}\alpha_j - s = 0, \tag{12}$$

$$\frac{\partial L}{\partial \xi_i} = \frac{\mu_i}{\nu_1 m_1} - \alpha_i - \gamma_i = 0, \quad i = 1, \ldots, m_1, \tag{13}$$

$$\frac{\partial L}{\partial \xi_j} = \frac{\mu_j}{\nu_2 m_2} - \alpha_j - \gamma_j = 0, \quad j = m_1 + 1, \ldots, N, \tag{14}$$

where $\mathbf{I}$ is an N-dimensional vector of ones, and $\mathbf{I} = [1, \ldots, 1]^T$. We have $\mathbf{w} = (\mathbf{I} + \eta\mathbf{S}_{lw})^{-1}\sum_{i=1}^{N}\alpha_i y_i \mathbf{x}_i$.

Substituting (10)–(14) into (9), we obtain the dual form of the optimization problem:

$$\begin{aligned}
\min_{\boldsymbol{\alpha}} \quad & \frac{1}{2}\boldsymbol{\alpha}^T \mathbf{H}\boldsymbol{\alpha} \\
\text{s.t.} \quad & \sum_{i=+1}^{m_1} \alpha_i = \nu \\
& \sum_{j=m_1+1}^{N} \alpha_j = \nu \\
& 0 \le \alpha_i \le \frac{\mu_i}{\nu_1 m_1}, \quad i = 1, \ldots, m_1 \\
& 0 \le \alpha_j \le \frac{\mu_j}{\nu_2 m_2}, \quad j = m_1 + 1, \ldots, N,
\end{aligned} \tag{15}$$

where $\mathbf{H}$ is a matrix with entry $H_{ij} = y_i y_j \mathbf{x}_i^T (\mathbf{I} + \eta \mathbf{S}_{lw})^{-1} \mathbf{x}_j$, and vectors $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^T$.

Equation (15) is a typical convex quadratic programming problem which is easy to be numerically solved. Suppose $\boldsymbol{\alpha}^* = [\alpha_1^*, \ldots, \alpha_N^*]$ can be used to solve the above optimization problem, and then the optimal weight vector is

$$\mathbf{w}^* = (\mathbf{I} + \eta \mathbf{S}_{lw})^{-1} \sum_{i=1}^{N} \alpha_i^* y_i \mathbf{x}_i. \tag{16}$$

Denote a training sample $\mathbf{x}_i$ $(1 \le i \le N)$ called a support vector (SV) if the corresponding Lagrange multiplier $\alpha_i > 0$. Denote the SV sets as $\mathrm{SV}_1 = \{\mathbf{x}_i \mid 0 < \alpha_i \le \mu_i/\nu_1 m_1,\ 1 \le i \le m_1\}$ and $\mathrm{SV}_2 = \{\mathbf{x}_j \mid 0 < \alpha_j \le \mu_j/\nu_2 m_2,\ 1 + m_1 \le j \le N\}$ while s^+ and s^- denote the number of SVs in SV_1 and SV_2, respectively. According to KKT condition, (15) becomes equations for the input data in SV_1 and SV_2, respectively, with slack variables ξ_i and ξ_j being 0. Thus, the optimal thresholds b^* and ρ^* can be calculated. However, from the numerical perspective, it is better to take the mean value of b^* and ρ^* resulting from all such data. Therefore, the optimal thresholds b^* and ρ^* are computed by the following formula:

$$b^* = 1 - \frac{1}{s^+} \sum_{\mathbf{x}_i \in \mathrm{SV}_1} \left(\mathbf{w}^*\right)^T \mathbf{x}_i, \tag{17}$$

$$\rho^* = -\frac{1}{s^+} \sum_{\mathbf{x}_i \in \mathrm{SV}_1} \left(\mathbf{w}^*\right)^T \mathbf{x}_i + \frac{1}{s^-} \sum_{\mathbf{x}_j \in \mathrm{SV}_2} \left(\mathbf{w}^*\right)^T \mathbf{x}_j. \tag{18}$$

As a result, the corresponding decision function of the linear FSVM-CIP will be

$$\begin{aligned}
f(\mathbf{x}) &= \operatorname{sgn}\left(\mathbf{w}^T \mathbf{x} + b^*\right) \\
&= \operatorname{sgn}\left(\sum_{i=1}^{N} \alpha_i^* y_i \left(\mathbf{x}_i^T (\mathbf{I} + \eta \mathbf{S}_{lw})^{-1} \mathbf{x}\right) + b^*\right).
\end{aligned} \tag{19}$$

Note that, to deal with the small sample size problem, $(\mathbf{I} + \eta\mathbf{S}_{lw})$ is regularized by adding a scale multiple η of the identity matrix $\mathbf{S}_{lw}$ with $\mathbf{I}$ before any inversion takes place. Hence, $(\mathbf{I} + \eta\mathbf{S}_{lw})$ is always nonsingular, and the inverse of $(\mathbf{I} + \eta\mathbf{S}_{lw})$ exists.

Following the terminology in [18], a training sample $\mathbf{x}_i$ $(1 \le i \le N)$ is called a margin error (ME) if the corresponding slack variable $\boldsymbol{\xi}_i > 0$. We give the following theorem for parameter selection later.

Theorem 3. *Let m^+ and m^- denote the number of MEs in the positive and negative classes; s^+ and s^- denote the number of SVs in the positive and negative classes, respectively. Then one has*

$$\overline{\mu_m^+} m^+ \le \nu\nu_1 m_1 \le \overline{\mu_s^+} s^+, \tag{20}$$

$$\overline{\mu_m^-} m^- \le \nu\nu_2 m_2 \le \overline{\mu_s^-} s^-, \tag{21}$$

where $\overline{\mu_m^+}$ and $\overline{\mu_m^-}$ denote the mean fuzzy membership of MEs in the positive and negative classes; $\overline{\mu_s^+}$ and $\overline{\mu_s^-}$ denote the mean fuzzy membership of SVs in the positive and negative classes, respectively.

A proof of the above theorem can be found in Appendix.

3.3. Fuzzy Membership Functions in the Linear Case. In FSVM, the fuzzy membership is used to reduce the effects of outliers or noises and different fuzzy membership functions have different influences on the fuzzy algorithm. Basically, the rule to assign proper membership values to data points can depend on the relative importance of date points to their own classes. In this paper, we consider two fuzzy membership functions given in [19].

Given the sequence of training points, denote the mean of positive class and negative class as $\overline{x}_+$ and $\overline{x}_-$.

Definition 4. The μ_{lin} is called the linear fuzzy membership and μ_{lin} can be defined as

$$\mu_{\text{lin}} = \begin{cases} \dfrac{1 - \left\|\mathbf{x}_i - \overline{\mathbf{x}}_+\right\|}{\left(\max_j \left(\left\|\mathbf{x}_j - \overline{\mathbf{x}}_+\right\|\right) + \delta\right)} & \text{if } y_i = 1 \\ \dfrac{1 - \left\|\mathbf{x}_i - \overline{\mathbf{x}}_-\right\|}{\left(\max_j \left(\left\|\mathbf{x}_j - \overline{\mathbf{x}}_-\right\|\right) + \delta\right)} & \text{if } y_i = -1, \end{cases} \tag{22}$$

where δ is a small positive value, which is used to avoid μ_{lin} becoming zero. $\|\cdot\|$ is the Euclidean distance.

Definition 5. The μ_{exp} is called the exponential fuzzy membership and μ_{exp} can be defined as

$$\mu_{\text{exp}} = \begin{cases} \dfrac{2}{1 + \exp\left(\lambda \left\|\mathbf{x}_i - \overline{\mathbf{x}}_+\right\|\right)} & \text{if } y_i = 1 \\ \dfrac{2}{1 + \exp\left(\lambda \left\|\mathbf{x}_i - \overline{\mathbf{x}}_-\right\|\right)} & \text{if } y_i = -1, \end{cases} \tag{23}$$

where parameter $\lambda \in [0, 1]$ determines the steepness of the decay.

3.4. Solution. Based on the above, we can state the approach of proposed FSVM-CIP in the linear case as Algorithm 1.

Input:
Training samples $\{(\mathbf{x}_i, y_i), i = 1, \ldots, N\}$
Testing samples $\{\mathbf{x}_j, j = 1, \ldots, U\}$
Output:
The predicted labels y_j of data $\{\mathbf{x}_j, j = 1, \ldots, U\}$
Procedure:
(1) Compute fuzzy membership μ_i using (22) or (23) for the data $\{(\mathbf{x}_i, y_i), i = 1, \ldots, N\}$
(2) Construct data adjacency graph G using k nearest neighbors and compute the edge weights matrix W_{ij} with N examples
(3) Construct local within-class preserving scatter matrix $\mathbf{S}_{\text{lw}}$ using (8)
(4) Choose parameters t (6); η, ν, ν_1 and ν_2 (8)
(5) Compute α^* using (15) and b^* using (17) with a QP Solver
(6) Using decision function (19) with samples $\mathbf{x}_j$, and output the final class labels

Algorithm 1: FSVM-CIP in the linear case.

4. FSVM for the Class Imbalance Problem in the Nonlinear Case

In this section, we extend the local within-class preserving scatter matrix and FSVM-CIP into feature space. Moreover, the fuzzy membership functions in feature space are defined. Finally, the algorithm of kernel FSVM-CIP is summarized.

4.1. Kernel Extension. In order to handle nonlinear classification, the kernelization trick [20] is used to map the n-dimensional date points into an arbitrary reproducing kernel Hilbert space (RKHS) [21] via a mapping function $\phi : \mathbf{R}^n \mapsto \mathbf{H}$; that is, $\mathbf{x_i} \mapsto \phi(\mathbf{x_i})$. Then a linear hyperplane $f(\mathbf{v}) = \boldsymbol{\alpha}^T\phi(\mathbf{v})+b$ in feature space $\mathbf{H}$ would correspond to a nonlinear hyperplane in the original space $\mathbf{R}^n$ where $\boldsymbol{\alpha}, \phi(\mathbf{v}) \in \mathbf{H}$, $\mathbf{v} \in \mathbf{R}^n$, and $b \in \mathbf{R}$.

Let $\phi(\mathbf{X})$ denote the date matrices in feature space $\mathbf{H}$, $\phi(\mathbf{X}) = [\phi(\mathbf{x}_1), \phi(\mathbf{x}_2), \ldots, \phi(\mathbf{x}_n)]$; then the kernel function $\mathbf{K}$ is a matrix with entry $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T\phi(\mathbf{x}_j)$.

Here the kernel local within-class scatter matrix $\mathbf{S}_{lw}^{\phi}$ in feature space is

$$\begin{aligned}
&\mathbf{S}_{lw}^{\phi} \\
&= \sum_{k=1}^{2}\sum_{i=1}^{N_k}\left(\phi(\mathbf{x}_i) - \sum_{j=1}^{N_k}\mathbf{W}_{ij}^{\phi k}\phi(\mathbf{x}_j)\right) \\
&\qquad \times \left(\phi(\mathbf{x}_i) - \sum_{j=1}^{N_k}\mathbf{W}_{ij}^{\phi k}\phi(\mathbf{x}_j)\right)^T \qquad (24)\\
&= \mathbf{K}^{(1)}\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi(1)}\right)^T\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi(1)}\right)\mathbf{K}^{(1)T} \\
&\quad + \mathbf{K}^{(2)}\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi(2)}\right)^T\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi(2)}\right)\mathbf{K}^{(2)T},
\end{aligned}$$

where $\mathbf{I}^{(1)}$, $\mathbf{I}^{(2)}$ are N_1-order, N_2-order identity matrixes, respectively. Based on the above notations, $\mathbf{K}^{(1)}$, $\mathbf{K}^{(2)}$ are $N \times m_1$, $N \times (N - m_1)$ matrixes, respectively; thus $\mathbf{K} = [\mathbf{K}^{(1)}, \mathbf{K}^{(2)}]$.

The weight matrixes $\mathbf{W}^{\phi(1)}$ and $\mathbf{W}^{\phi(2)}$ are the nonlinear version of $\mathbf{W}^{(1)}$ and $\mathbf{W}^{(2)}$, respectively. $\mathbf{W}^{\phi(1)}$ and $\mathbf{W}^{\phi(2)}$ could be built by W_{ij}^{ϕ}, and the nonlinear version of W_{ij}^{ϕ} is

$$W_{ij}^{\phi} = \begin{cases} \dfrac{1}{D_{ii}^{\phi}}\exp\left(\dfrac{-\left(K_{ii} + K_{jj} - 2K_{ij}\right)}{t}\right) & \text{if } \mathbf{x}_i \in N_k\left(\mathbf{x}_j\right) \\ & \text{or } \mathbf{x}_j \in N_k\left(\mathbf{x}_i\right), \\ & y_i = y_j \\ 0 & \text{otherwise,} \end{cases} \qquad (25)$$

where $D_{ii}^{\phi} = \sum_j W_{ij}^{\phi}$ is a normalizer.

Thus, the kernel FSVM-CIP can be easily achieved by solving the following quadratic problem:

$$\begin{aligned}
\min_{w,b,\rho,\xi} \quad & \frac{1}{2}\mathbf{w}^T\mathbf{w} - \nu\rho + \frac{1}{\nu_1 m_1}\sum_{i=1}^{m_1}\mu_i\xi_i + \frac{1}{\nu_2 m_2}\sum_{j=m_1+1}^{N}\mu_j\xi_j \\
& + \frac{\eta}{2}\mathbf{w}^T\mathbf{S}_{lw}^{\phi}\mathbf{w} \\
\text{s.t.} \quad & \mathbf{w}^T\phi\left(\mathbf{x}_i\right) + b \geq 1 - \xi_i, \quad i = 1, \ldots, m_1 \\
& \mathbf{w}^T\phi\left(\mathbf{x}_j\right) + b \geq 1 + \rho - \xi_j, \quad j = m_1 + 1, \ldots, N \\
& \xi_k \geq 0, \quad k = 1, \ldots, N, \ \rho \geq 0.
\end{aligned} \qquad (26)$$

Like its linear counterpart, the solution to this optimization problem can be easily found using Lagrange multipliers. By using the representer theorem, $\mathbf{w}$ can be given by $\mathbf{w} = \sum_{i=1}^{N}\beta_i\phi(\mathbf{x}_i)$. We obtain the dual form of the optimization problem:

$$\begin{aligned}
\min_{\alpha} \quad & \frac{1}{2}\boldsymbol{\alpha}^T\mathbf{M}\boldsymbol{\alpha} \\
\text{s.t.} \quad & \sum_{i=1}^{m_1}\alpha_i = \nu \\
& \sum_{j=m_1+1}^{N}\alpha_j = \nu
\end{aligned}$$

$$0 \le \alpha_i \le \frac{\mu_i}{\nu_1 m_1}, \quad i = 1, \ldots, m_1$$
$$0 \le \alpha_j \le \frac{\mu_j}{\nu_2 m_2}, \quad j = m_1 + 1, \ldots, N, \tag{27}$$

where $\mathbf{M} = \mathbf{Y}\mathbf{K}^T\mathbf{Q}^{-1}\mathbf{K}\mathbf{Y}$ and $\mathbf{Q} = \mathbf{K} + \eta\mathbf{K}^{(1)}(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}})^T(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}})\mathbf{K}^{(1)T} + \eta\mathbf{K}^{(2)}(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}})^T(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}})\mathbf{K}^{(2)T}$. Vectors $\boldsymbol{\alpha} = [\alpha_1, \ldots, \alpha_N]^T$, and $Y = \mathrm{diag}(y_1, y_2, \ldots, y_n)$ is a diagonal matrix.

Equation (27) is a typical convex quadratic programming problem which is easy to be numerically solved. Suppose $\boldsymbol{\alpha}^* = [\alpha_1^*, \ldots, \alpha_N^*]^T$ can be used to solve the above optimization problem; then the optimal weight vector $\boldsymbol{\beta}^* = \mathbf{Q}^{-1}\mathbf{K}\mathbf{Y}\boldsymbol{\alpha}^*$. Therefore, the optimal thresholds b^* and ρ^* are computed by the following formula:

$$b^* = 1 - \frac{1}{s^+} \sum_{\mathbf{x}_i \in \mathrm{sv}_1} \sum_{j=1}^{N} \beta_j^* y_j K\left(\mathbf{x}_i, \mathbf{x}_j\right), \tag{28}$$

$$\rho^* = -\frac{1}{s^+} \sum_{\mathbf{x}_i \in \mathrm{SV}_1} \sum_{j=1}^{N} \beta_j^* y_j K\left(\mathbf{x}_i, \mathbf{x}_j\right) + \frac{1}{s^-} \sum_{\mathbf{x}_i \in \mathrm{SV}_2} \sum_{j=1}^{N} \beta_j^* y_j K\left(\mathbf{x}_i, \mathbf{x}_j\right). \tag{29}$$

Finally, a more robust decision function of kernel FSVM-CIP will be

$$f(\mathbf{x}) = \mathrm{sgn}\left(\sum_{i=1}^{N} \beta_i^* K(\mathbf{x}, \mathbf{x}_i) + b^*\right). \tag{30}$$

Theorem 6. *The matrix* $\mathbf{M}$ *in* (27) *is symmetric and positive semidefinite.*

A proof of the above theorem can be found in Appendix.

Next, we consider fuzzy membership functions in feature space.

Definition 7. The $\mu_{\mathrm{lin}}^{\phi}$ is called the linear fuzzy membership in feature space and $\mu_{\mathrm{lin}}^{\phi}$ can be defined as

$$\mu_{\mathrm{lin}}^{\phi} = \begin{cases} \dfrac{1 - \left\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_+)\right\|}{\left(\max_j \left(\left\|\phi(\mathbf{x}_j) - \phi(\overline{\mathbf{x}}_+)\right\|\right) + \delta\right)} & \text{if } y_i = 1 \\ \dfrac{1 - \left\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_-)\right\|}{\left(\max_j \left(\left\|\phi(\mathbf{x}_j) - \phi(\overline{\mathbf{x}}_-)\right\|\right) + \delta\right)} & \text{if } y_i = -1, \end{cases} \tag{31}$$

where δ is a small positive value. $\|\cdot\|$ is the Euclidean distance.

Definition 8. The $\mu_{\mathrm{exp}}^{\phi}$ is called the exponential fuzzy membership in feature space and $\mu_{\mathrm{exp}}^{\phi}$ can be defined as

$$\mu_{\mathrm{exp}}^{\phi} = \begin{cases} \dfrac{2}{1 + \exp\left(\lambda \left\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_+)\right\|\right)} & \text{if } y_i = 1 \\ \dfrac{2}{1 + \exp\left(\lambda \left\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_-)\right\|\right)} & \text{if } y_i = -1, \end{cases} \tag{32}$$

where parameter $\lambda \in [0, 1]$ determines the steepness of the decay. Consider

$$\phi(\overline{\mathbf{x}}_+) = \frac{1}{m_1} \sum_{x_i \in c_1} \phi(\mathbf{x}_i),$$
$$\phi(\overline{\mathbf{x}}_-) = \frac{1}{N - m_1} \sum_{x_i \in c_2} \phi(\mathbf{x}_i). \tag{33}$$

Thus, the distance $\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_+)\|$ can be given by

$$\left\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_+)\right\| = \sqrt{K(\mathbf{x}_i, \mathbf{x}_i) - \frac{2}{m_1} \sum_{\mathbf{x}_j \in C_1} K\left(\mathbf{x}_i, \mathbf{x}_j\right) + \frac{1}{m_1^2} \sum_{\mathbf{x}_s \in C_1} \sum_{\mathbf{x}_t \in C_1} K\left(\mathbf{x}_s, \mathbf{x}_t\right)}. \tag{34}$$

Likewise, the $\|\phi(\mathbf{x}_i) - \phi(\overline{\mathbf{x}}_-)\|$ can be given in a similar manner.

4.2. Solution. Based on the above, we can state the approach of kernel FSVM-CIP as Algorithm 2.

5. Experiments and Discussions

To evaluate the performance of our proposed FSVM-CIP, in this section, FSVM-CIP is evaluated compared with other related representative methods, such as standard FSVM [8], SVDD [11], FSVM for class imbalance learning (FSVM-CIL) [22], and FSVM with minimum within-class scatter (WCS-FSVM) [23]. We implement FSVM-CIP using the linear fuzzy membership and the exponential fuzzy membership, respectively, which are represented as FSVM-CIP$_{\mathrm{lin}}$ and FSVM-CIP$_{\mathrm{exp}}$. All the experiments are performed in Matlab (R2010a) on personal computer, whose configuration is as follows: CPU 2.99 GHz, 4.0 G RAM, and Microsoft Windows XP.

5.1. Data Preparation. In this section, we use five real-world medical datasets from the UCI repository of machine learning database [24], to demonstrate the classification performance of the method proposed in this paper. These five medical datasets are breast, heart, hepatitis, BUPA liver, and pima diabetes. It is highly likely that these real-world datasets contain some outliers and noisy examples in different amounts [22]. In each of them, the positive class consists of the data corresponding to the healthy, normal, or benign cases, while the negative class contains the data for diseased, abnormal, or malignant cases. Further details of these datasets are provided in Table 1. This contains the total number of positive data #pos, the total number of negative data #neg, the number of positive training examples $m1$, the number of negative training examples $m2$, the positive-to-negative imbalance ratio Ratio, and the data dimensionality d.

5.2. Performance Measure and Experimental Settings. We used the geometric mean of sensitivity (sensitivity = proportion of the positives correctly recognized), specificity (specificity = proportion of the negatives correctly recognized),

Input:
training samples $\{(\mathbf{x}_i, y_i), i = 1, \ldots, N\}$
Testing samples $\{\mathbf{x}_j, j = 1, \ldots, U\}$
Output:
The predicted labels y_j of data $\{\mathbf{x}_j, j = 1, \ldots, U\}$
Procedure:
(1) Choose a kernel function **K**. Compute the Gram matrix $K_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$.
(2) Compute fuzzy membership μ_i^{ϕ} using (31) or (32) for the data $\{(\mathbf{x}_i, y_i), i = 1, \ldots, N\}$
(3) Construct data adjacency graph G using k nearest neighbors and compute the edge weights matrix W_{ij}^{ϕ} with N examples
(4) Construct local within-class preserving scatter matrix $\mathbf{S}_{\text{lw}}^{\phi}$ using (24)
(5) Choose parameters t (25); η, ν, ν_1 and ν_2 (26)
(6) Compute α^* using (27) and b^* using (28) with a QP Solver
(7) Using decision function (30) with samples $\mathbf{x}_j$, and output the final class labels

Algorithm 2: Kernel FSVM-CIP.

Table 1: Characteristics of the selected datasets.

Datasets	#pos	#neg	$m1$	$m2$	Ratio	d
Breast	458	241	240	120	2 : 1	9
Heart	120	150	80	20	4 : 1	13
Hepatitis	123	32	100	10	10 : 1	19
BUPA liver	200	145	150	10	15 : 1	6
Pima diabetes	268	500	180	10	18 : 1	8

and accuracy (accuracy = proportion of correctly classified instances) for the classifier performance evaluation in experiments, as commonly used in medical datasets classification research [7].

Like the existing SVM and FSVM algorithms, the solution is sensitive to the setting of the parameters. In order to evaluate the performance, a strategy is that a set of the parameters is given first and then the best cross-validation mean rate among the set is used to estimate the generalized accuracy. We adopt this strategy in this paper. For FSVM-CIP, the parameter ν is searched in $\{1, 5, 10, 15, \ldots, 80\}$, while ν_1 and ν_2 are selected from $\{0.001, 0.005, 0.01, 0.05\}$. η is selected from $\log_2\eta \in \{-5, -4.5, -4, \ldots, 5.5, 6\}$. The heat kernel parameter t is searched in $\{0.5, 1.0, 1.5, 2.0, 2.5, 3.0, 3.5, 4.0\}$ and the neighborhood parameter k is searched in $\{3, 5, 7, 9, 11, 13, 15\}$. In addition, when the linear fuzzy function is used, we set $\delta = 10^{-6}$. When the exponential fuzzy function is used, the optimal value of λ is chosen from the range $\lambda = \{0.1, 0.2, 0.3, \ldots, 1\}$.

The regularization parameter C for FSVM, SVDD, FSVM-CIL, and WCS-FSVM is selected from the set $\{0.001, 0.01, 0.1, 1, 10, 100\}$. In WCS-FSVM, β is selected from $\log_2\beta \in \{-5, -4.5, -4, \ldots, 5.5, 6\}$. For FSVM-CIL, the fuzzy membership is based on the distance from the actual hyperplane and uses the exponential fuzzy membership λ. λ is chosen from the range $\lambda = \{0.1, 0.2, 0.3, \ldots, 1\}$.

For the kernel-based methods, we use a Gaussian RBF kernel, that is, $\exp(-(u - v)^T(u - v)/\sigma)$, where σ is the spread of Gaussian kernel, and σ is searched in $\{\tau^2/16, \tau^2/8, \tau^2/4, \tau^2/2, \tau^2, 2\tau^2, 4\tau^2, 8\tau^2, 16\tau^2\}$, where τ^2 is the mean norm of the training data.

For parameter selection, we conduct fivefold cross-validation in a stratified manner so that each validation set has the same positive to negative ratio as in the training set. Finally, the experiment is repeated 10 times independently of each dataset.

5.3. Experimental Results. FSVM-CIP method test results developed for the breast, heart, hepatitis, BUPA liver, and pima diabetes datasets are given both in the linear case and nonlinear case. Tables 2, 3, 4, 5, and 6 display the comparison results with the other methods on these five databases, respectively.

The main observations from the performance comparisons include the following.

(1) We can see that, in many real-world applications, a linear classifier seems powerless. In terms of accuracy, kernel method can improve the classification performance for all five medical datasets.

(2) We can clearly observe that the FSVM-CIP outperforms other methods on almost datasets both in the linear case and nonlinear case, which gives higher accuracy. This fortifies the fact that the locality maximum margin and the local structure information presented by local within-class preserving scatter could improve classification performance; furthermore, the method of different misclassification costs based on the number of two classes is a sensitive learning solution to overcome the imbalance problem in SVMs.

(3) It is noted that, for all the datasets considered, the classification accuracy given by the FSVM-CIP$_{\text{exp}}$ setting is higher than the FSVM-CIP$_{\text{lin}}$ setting. Therefore, we can state that FSVM-CIP$_{\text{exp}}$ setting with the appropriate selection

Table 2: Comparison of the classification results (%) on breast dataset.

	Method	Sensitivity	Specificity	Accuracy
Linear	FSVM	95.87 ± 0.017	95.04 ± 0.043	95.58 ± 0.035
	SVDD	97.71 ± 0.065	90.90 ± 0.013	95.28 ± 0.052
	FSVM-CIL	95.87 ± 0.024	95.87 ± 0.015	95.81 ± 0.028
	WCS-FSVM	96.33 ± 0.067	95.04 ± 0.056	95.87 ± 0.047
	FSVM-CIP$_{lin}$	96.98 ± 0.039	96.49 ± 0.022	**96.76 ± 0.040**
	FSVM-CIP$_{exp}$	96.68 ± 0.011	96.69 ± 0.042	**96.76 ± 0.037**
Gaussian kernel	FSVM	96.33 ± 0.023	95.87 ± 0.051	96.17 ± 0.050
	SVDD	97.30 ± 0.065	91.25 ± 0.013	95.44 ± 0.052
	FSVM-CIL	96.79 ± 0.059	95.87 ± 0.042	96.46 ± 0.055
	WCS-FSVM	96.97 ± 0.030	96.69 ± 0.093	96.76 ± 0.067
	FSVM-CIP$_{lin}$	97.25 ± 0.055	96.29 ± 0.032	97.05 ± 0.042
	FSVM-CIP$_{exp}$	97.25 ± 0.055	97.52 ± 0.045	**97.34 ± 0.033**

Table 3: Comparison of the classification results (%) on heart dataset.

	Method	Sensitivity	Specificity	Accuracy
Linear	FSVM	87.50 ± 0.080	80.77 ± 0.069	82.35 ± 0.069
	SVDD	87.03 ± 0.021	77.69 ± 0.005	80.00 ± 0.051
	FSVM-CIL	85.00 ± 0.046	82.04 ± 0.110	82.35 ± 0.072
	WCS-FSVM	87.30 ± 0.071	81.54 ± 0.089	82.94 ± 0.088
	FSVM-CIP$_{lin}$	85.00 ± 0.063	82.31 ± 0.083	82.84 ± 0.054
	FSVM-CIP$_{exp}$	87.50 ± 0.025	82.31 ± 0.083	**83.53 ± 0.055**
Gaussian kernel	FSVM	86.70 ± 0.099	82.61 ± 0.087	83.35 ± 0.042
	SVDD	90.35 ± 0.022	80.77 ± 0.034	82.80 ± 0.070
	FSVM-CIL	87.05 ± 0.034	81.54 ± 0.067	82.94 ± 0.044
	WCS-FSVM	91.00 ± 0.076	81.73 ± 0.083	84.12 ± 0.085
	FSVM-CIP$_{lin}$	90.00 ± 0.045	82.31 ± 0.086	84.12 ± 0.052
	FSVM-CIP$_{exp}$	86.05 ± 0.023	83.08 ± 0.078	**84.71 ± 0.066**

Table 4: Comparison of the classification results (%) on hepatitis dataset.

	Method	Sensitivity	Specificity	Accuracy
Linear	FSVM	82.60 ± 0.053	22.73 ± 0.087	53.33 ± 0.073
	SVDD	73.91 ± 0.071	45.45 ± 0.011	60.00 ± 0.046
	FSVM-CIL	77.66 ± 0.026	45.46 ± 0.082	61.02 ± 0.070
	WCS-FSVM	79.56 ± 0.107	27.27 ± 0.062	53.33 ± 0.059
	FSVM-CIP$_{lin}$	78.26 ± 0.046	45.46 ± 0.032	62.22 ± 0.023
	FSVM-CIP$_{exp}$	78.26 ± 0.068	50.00 ± 0.086	**64.44 ± 0.071**
Gaussian kernel	FSVM	73.91 ± 0.038	31.82 ± 0.012	53.33 ± 0.025
	SVDD	82.60 ± 0.053	42.86 ± 0.025	63.64 ± 0.030
	FSVM-CIL	77.26 ± 0.041	50.00 ± 0.086	63.84 ± 0.064
	WCS-FSVM	78.26 ± 0.015	36.36 ± 0.074	57.78 ± 0.056
	FSVM-CIP$_{lin}$	73.51 ± 0.064	54.55 ± 0.037	64.44 ± 0.058
	FSVM-CIP$_{exp}$	73.91 ± 0.050	59.10 ± 0.011	**66.67 ± 0.036**

of λ value would be an effective choice applied to any medical dataset. In other words, when dealing with medical datasets classification, the performance of the exponential fuzzy membership is better than linear fuzzy membership in FSVM-CIP.

(4) For breast and heart datasets, the class imbalance is not obviously shaped; WCS-FSVM yielded standard FSVM, SVDD, and FSVM-CIL. We can say that the performance can indeed be improved when the structure of the data is taken into consideration. For the other three datasets, the class

Table 5: Comparison of the classification results (%) on BUPA liver dataset.

	Method	Sensitivity	Specificity	Accuracy
Linear	FSVM	88.10 ± 0.008	66.42 ± 0.073	72.19 ± 0.057
	SVDD	87.27 ± 0.021	68.05 ± 0.063	72.72 ± 0.042
	FSVM-CIL	88.00 ± 0.004	67.44 ± 0.042	73.19 ± 0.015
	WCS-FSVM	84.00 ± 0.360	67.15 ± 0.068	71.66 ± 0.051
	FSVM-CIP$_{lin}$	88.00 ± 0.004	67.88 ± 0.063	73.26 ± 0.031
	FSVM-CIP$_{exp}$	86.00 ± 0.048	69.34 ± 0.072	**73.80 ± 0.054**
Gaussian kernel	FSVM	96.00 ± 0.057	66.67 ± 0.026	74.60 ± 0.038
	SVDD	95.43 ± 0.033	71.24 ± 0.050	77.23 ± 0.017
	FSVM-CIL	95.00 ± 0.045	72.59 ± 0.052	78.37 ± 0.050
	WCS-FSVM	90.08 ± 0.070	67.44 ± 0.083	73.73 ± 0.062
	FSVM-CIP$_{lin}$	94.00 ± 0.049	74.10 ± 0.045	79.46 ± 0.048
	FSVM-CIP$_{exp}$	94.00 ± 0.049	73.33 ± 0.084	**79.92 ± 0.074**

Table 6: Comparison of the classification results (%) on pima diabetes dataset.

	Method	Sensitivity	Specificity	Accuracy
Linear	FSVM	91.91 ± 0.022	49.98 ± 0.053	55.36 ± 0.051
	SVDD	88.65 ± 0.081	53.43 ± 0.062	58.45 ± 0.029
	FSVM-CIL	86.36 ± 0.064	55.10 ± 0.059	59.86 ± 0.060
	WCS-FSVM	87.50 ± 0.043	52.65 ± 0.024	57.96 ± 0.030
	FSVM-CIP$_{lin}$	85.23 ± 0.021	57.76 ± 0.064	**61.94 ± 0.043**
	FSVM-CIP$_{exp}$	84.09 ± 0.009	57.96 ± 0.062	**61.94 ± 0.053**
Gaussian kernel	FSVM	93.18 ± 0.031	51.02 ± 0.073	57.44 ± 0.053
	SVDD	91.76 ± 0.025	56.86 ± 0.052	62.57 ± 0.028
	FSVM-CIL	90.91 ± 0.047	58.78 ± 0.084	63.67 ± 0.077
	WCS-FSVM	92.05 ± 0.010	54.69 ± 0.066	60.38 ± 0.053
	FSVM-CIP$_{lin}$	88.84 ± 0.040	61.38 ± 0.063	**65.57 ± 0.063**
	FSVM-CIP$_{exp}$	88.64 ± 0.029	61.43 ± 0.074	**65.57 ± 0.070**

imbalance strikingly improved, the results given by standard FSVM and WCS-FSVM for datasets are biased towards the majority class represented as lower specificity and lower accuracy. These results justify the fact that these two methods are sensitive to the class imbalance problem. Meanwhile, SVDD and FSVM-CIL yielded standard FSVM and WCS-FSVM. BY assigning different misclassification costs for the minority class and majority class, the effect of class imbalance could be reduced.

5.4. Parameter Selection for Kernel FSVM-CIP$_{exp}$. The parameter $\eta > 0$ is an essential parameter in our proposed method which controls the tradeoff between the local within-class scatter and the margin. Figure 2 shows the impact of parameter η on the classification accuracy of FSVM-CIP$_{exp}$ in kernel case with each value of η selected from $\log_2\eta \in \{-5, -4.5, -4, \ldots, 5.5, 6\}$. It can be seen that the best accuracy is obtained for all the datasets and therefore η is searched in a reasonable range.

Compared with standard FSVM, the additional neighbor parameter k is employed in FSVM-CIP. To evaluate the influence of this parameter on the performance, the classification accuracy of kernel FSVM-CIP$_{exp}$ for five medical databases is recorded for each value of k in $\{3, 5, 7, 9, 11, 13, 15\}$. Figure 3 shows the results. It can be seen that the classification accuracy is not high when k value is small and, by increasing k, the classification accuracy increases; however, if k continues to increase, the classification accuracy begins to drop severely down. It is because, when k is too small, the number of nearest neighbors is sparse; when k is too large, the number of nearest neighbors is excessive, so to preserve so much local relation may be inappropriate.

6. Conclusion

Computer tools have improved the medical practice implementation to a greater extent. Although computer tools cannot replace the doctors, they can make their work easier and more effective. In this paper, a new fuzzy support machine called FSVM-CIP, used for medical datasets classification, is proposed. The proposed method is based on local within-class preserving scatter and assigned two misclassification costs in the SVM objective function, which is for learning from imbalance datasets in the presence of outliers/noise and enhancing the locality maximum margin. Experiments were performed on several UCI medical datasets with a comparison of the proposed method with several other related methods such as standard FSVM, SVDD, FSVM-CIL, and

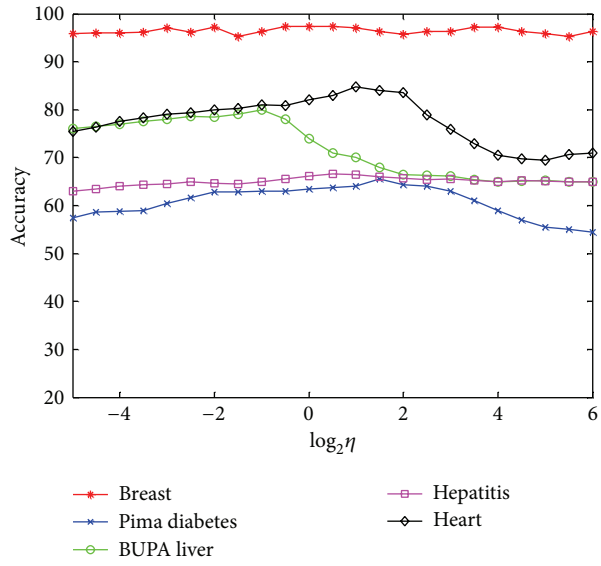


Figure 2: The effect of the parameter η on kernel FSVM-CIP$_{exp}$.

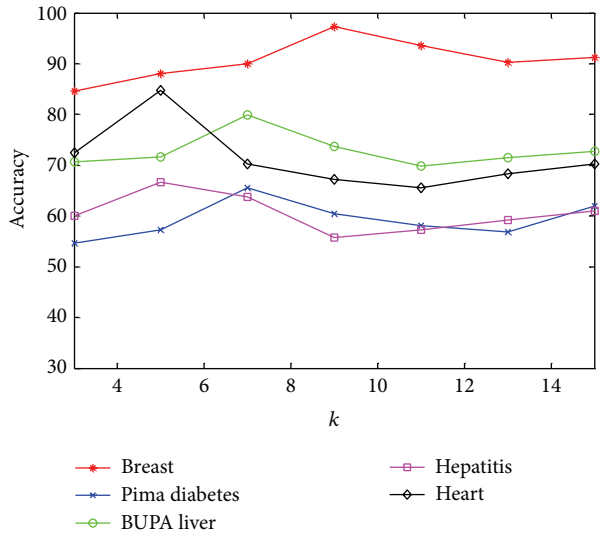


Figure 3: The effect of the parameter k on kernel FSVM-CIP$_{exp}$.

WCS-FSVM. Obtained results show that the performance of the proposed method is highly successful compared to other results attained and seems very promising. Finally, we can recommend that FSVM-CIP$_{exp}$ which uses the exponential fuzzy membership would be an effective choice for medical datasets classification applications. In future work, we intend to perform investigations to large-scale classification problems.

Appendix

Proof of Theorem 3 in Section 3.2.

Proof. According to the dual form of the optimization problem (15), we can derive

$$\sum_{i=1}^{m_1}\alpha_i = \nu. \tag{A.1}$$

Likewise, according to the KKT conditions, $\sum_{i=1}^{N}\alpha_i = \nu$ with $\rho > 0$ satisfy $s = 0$ by (12). According to (11), all samples with $\boldsymbol{\xi}_i > 0$ satisfy $\gamma_i = 0$. In view of (13), this implies that $\alpha_i = \mu_i/\nu_1 m_1$ holds for every positive ME. Summing up α_i over the positive MEs using (A.1), we have

$$\frac{\overline{\mu_m^+}m^+}{\nu_1 m_1} \le \sum_{i=1}^{m_1}\alpha_i = \nu. \tag{A.2}$$

Furthermore, in view of (15), each SV in the positive class can control at most $1/\nu_1 m_1$ to the $\sum_{i=1}^{m_1}\alpha_i$; as a result,

$$\sum_{i=1}^{m_1}\alpha_i \le \frac{\overline{\mu_s^+}s^+}{\nu_1 m_1}. \tag{A.3}$$

Combining (A.2) and (A.3), inequality (20) can hold true. Likewise, inequality (21) can be proven in a similar manner. □

Proof of Theorem 6 in Section 4.1.

Proof. We know that $\mathbf{M} = \mathbf{Y}\mathbf{K}^T\mathbf{Q}^{-1}\mathbf{K}\mathbf{Y}$, and $\mathbf{K}$ is a Gram matrix, so $\mathbf{K}$ is symmetric and positive semidefinite. The transpose of the matrix $\mathbf{Q}$ is

$$\begin{aligned}\mathbf{Q}^T = &\left(\mathbf{K} + \eta\mathbf{K}^{(1)}\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)^T\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)\mathbf{K}^{(1)T}\right.\\ &\left.+\eta\mathbf{K}^{(2)}\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)^T\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)\mathbf{K}^{(2)T}\right)^T = \mathbf{Q}.\end{aligned} \tag{A.4}$$

So $\mathbf{Q}$ is a symmetric matrix and then $\mathbf{M}$ is symmetric. Set $\mathbf{Q} = \mathbf{K} + \eta\mathbf{R}$, where

$$\begin{aligned}\mathbf{R} = &\mathbf{K}^{(1)}\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)^T\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)\mathbf{K}^{(1)T}\\ &+\mathbf{K}^{(2)}\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)^T\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)\mathbf{K}^{(2)T}.\end{aligned} \tag{A.5}$$

For any nonzero vector $\mathbf{u} = (u_1, u_2, \ldots, u_N)^T$,

$$\begin{aligned}\mathbf{u}^T\mathbf{R}\mathbf{u} = &\mathbf{u}^T\mathbf{K}^{(1)}\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)^T\left(\mathbf{I}^{(1)} - \mathbf{W}^{\phi^{(1)}}\right)\mathbf{K}^{(1)T}\mathbf{u}\\ &+\mathbf{u}^T\mathbf{K}^{(2)}\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)^T\left(\mathbf{I}^{(2)} - \mathbf{W}^{\phi^{(2)}}\right)\mathbf{K}^{(2)T}\mathbf{u}\\ = &\boldsymbol{\varsigma}^T\mathbf{S}_{lw}^{\phi}\boldsymbol{\varsigma} \ge 0,\end{aligned} \tag{A.6}$$

where $\boldsymbol{\varsigma} = \sum_{i=1}^{N}u_i\phi(\mathbf{x}_i)$. The local within-class scatter matrix $\mathbf{S}_{lw}^{\phi}$ is semidefinite, so the matrix $\mathbf{R}$ is semidefinite. That is, the matrix $\mathbf{Q}$ is semidefinite, and then $\mathbf{M}$ is semidefinite. □

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported by the National Natural Science Foundation of China under Contact (61070121).

References

[1] S. Ghumbre, C. Patil, and A. Ghatol, "Heart disease diagnosis using support vector machine," in *Proceedings of the International Conference on Computer Science and Information Technology (ICCSIT '11)*, Pattaya, Thailand, 2011.

[2] M. Tong, K. Liu, C. Xu, and W. Ju, "An ensemble of SVM classifiers based on gene pairs," *Computers in Biology and Medicine*, vol. 43, no. 6, pp. 729–737, 2013.

[3] Y. Chang, N. Kim, Y. Lee, J. Lim, and J. B. Seo, "Fast and efficient lung disease classification using hierarchical one-against-all support vector machine and cost-sensitive feature selection," *Computers in Biology and Medicine*, vol. 42, no. 12, pp. 1157–1164, 2012.

[4] D. C. Li, C. W. Liu, and S. C. Hu, "A fuzzy-based data transformation for feature extraction to increase classification performance with small medical data sets," *Artificial Intelligence in Medicine*, vol. 52, no. 1, pp. 45–52, 2011.

[5] M. Serter, U. N. Yilmaz, and O. Inan, "Feature selection method based on artificial bee colony algorithm and support vector machines for medical datasets classification," *The Scientific World Journal*, vol. 2013, Article ID 419187, 10 pages, 2013.

[6] V. N. Vapnik, *The Natural of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.

[7] V. N. Vapnik, *Statistical Learning Theory*, Wiley, New York, NY, USA, 1998.

[8] C. F. Lin and S. D. Wang, "Fuzzy support vector machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 464–471, 2002.

[9] K. Veropoulos, C. Campbell, and N. Cristianini, "Controlling the sensitivity of support vector machines," in *Proceedings of the International Joint Conferences on Artificial Intelligence*, pp. 55–60, Stockholm, Sweden, 1999.

[10] B. Schölkopf and A. J. Smola, *Learning with Kernels*, MIT Press, 2002.

[11] D. M. J. Tax and R. P. W. Duin, "Support vector data description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.

[12] E. Kokiopoulou and Y. Saad, "Orthogonal neighborhood preserving projections: a projection-based dimensionality reduction technique," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2143–2156, 2007.

[13] X. He and P. Niyogi, "Locality preserving projections," in *Proceedings of the Conference on Advances in Neural Information Processing Systems*, pp. 585–591, 2003.

[14] X. He, S. Yan, Y. Hu, P. Niyogi, and H. J. Zhang, "Face recognition using Laplacianfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 3, pp. 328–340, 2005.

[15] X. Wang and Y. Niu, "New one-versus-all ν-SVM solving intra-inter class imbalance with extended manifold regularization and localized relative maximum margin," *Neural Computing*, vol. 115, no. 9, pp. 106–121, 2013.

[16] S. Zafeiriou, A. Tefas, and I. Pitas, "Minimum class variance support vector machines," *IEEE Transactions on Image Processing*, vol. 16, no. 10, pp. 2551–2564, 2007.

[17] I. Kotsia, S. Zafeiriou, and I. Pitas, "Novel multiclass classifiers based on the minimization of the within-class variance," *IEEE Transactions on Neural Networks*, vol. 20, no. 1, pp. 14–34, 2009.

[18] M. Wu and J. Ye, "A small sphere and large margin approach for novelty detection using training data with outliers," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 31, no. 11, pp. 2088–2092, 2009.

[19] X. Jiang, Z. Yi, and J. C. Lv, "Fuzzy SVM with a new fuzzy membership function," *Neural Computing and Applications*, vol. 15, no. 3-4, pp. 268–276, 2006.

[20] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernel-based learning algorithms," *IEEE Transactions on Neural Networks*, vol. 12, no. 2, pp. 181–201, 2001.

[21] G. Wahba, "Support vector machines, reproducing kernel hilbert spaces and the randomized gacv," in *Advances in Kernel Methods: Support Vector Learning*, MIT Press, Cambridge, Mass, USA, 1998.

[22] R. Batuwita and V. Palade, "FSVM-CIL: fuzzy support vector machines for class imbalance learning," *IEEE Transactions on Fuzzy Systems*, vol. 18, no. 3, pp. 558–571, 2010.

[23] W. An and M. Liang, "Fuzzy support vector machine based on within-class scatter for classification problems with outliers or noises," *Neural Computing*, vol. 110, no. 6, pp. 101–110, 2013.

[24] "UCI Repository of machine learning database," http://www.ics.uci.edu/%20mlearn/MLRepository.html.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 364352, 11 pages
http://dx.doi.org/10.1155/2014/364352

Research Article

Chaotic Multiquenching Annealing Applied to the Protein Folding Problem

Juan Frausto-Solis,[1] Ernesto Liñan-García,[2] Mishael Sánchez-Pérez,[3] and Juan Paulo Sánchez-Hernández[1,4]

[1] *Universidad Politécnica del Estado de Morelos Boulevard, Cuauhnáhuac 566, 62660 Jiutepec, Mexico*
[2] *Universidad Autónoma de Coahuila Boulevard, Venustiano Carranza s/n, 25280 Saltillo, Mexico*
[3] *Computational Genomics Research Program, Center for Genomic Sciences, Universidad Nacional Autónoma de México, Avenida Universidad s/n, 62210 Cuernavaca, Mexico*
[4] *Instituto Tecnológico y de Estudios Superiores de Monterrey, Autopista del Sol, 62790 Xochitepec, Mexico*

Correspondence should be addressed to Juan Frausto-Solis; juan.frausto@upemor.edu.mx

Received 15 October 2013; Accepted 19 January 2014; Published 20 March 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

The Chaotic Multiquenching Annealing algorithm (CMQA) is proposed. CMQA is a new algorithm, which is applied to protein folding problem (PFP). This algorithm is divided into three phases: (i) multiquenching phase (MQP), (ii) annealing phase (AP), and (iii) dynamical equilibrium phase (DEP). MQP enforces several stages of quick quenching processes that include chaotic functions. The chaotic functions can increase the exploration potential of solutions space of PFP. AP phase implements a simulated annealing algorithm (SA) with an exponential cooling function. MQP and AP are delimited by different ranges of temperatures; MQP is applied for a range of temperatures which goes from extremely high values to very high values; AP searches for solutions in a range of temperatures from high values to extremely low values. DEP phase finds the equilibrium in a dynamic way by applying least squares method. CMQA is tested with several instances of PFP.

1. Introduction

DNA is a molecule that contains genetic instructions, which are used in protein synthesis process [1]. This molecule has a complete set of hereditary information of any organism. DNA is formed by four different nucleotides, Adenine identified by the letter *A*, Cytosine identified by the letter *C*, Guanine identified by the letter *G*, and Thymine identified by the letter *T*. This molecule is divided into genes; each gene is a sequence of nucleotides that can express a functional protein. The transcription process of DNA creates an RNA molecule, which generates proteins. A protein is a linear polypeptide of amino acids, which are joined by peptide bonds. The atoms of a protein are arranged in a three-dimensional structure geometric model. In principle, function and structure of a protein are determined by its amino acids sequence. A functional protein is conformed in a geometrical model with a global minimum energy [2]; however, there are some exceptions [3]. This structure is usually named native structure (NS). The free energy of a conformation depends on the interaction among the atoms and their relative positions; normally, this energy can be calculated using torsion angles and the distance among atoms.

A protein can take consequently many different conformational structures from its primary structure to its native structure [4]. Therefore, computational methods are currently designed in order to find the optimal solution, which has the minimal free energy and determines the NS. The computational problem involved to find the NS is known as protein folding problem (PFP). Because PFP is a NP problem [5], metaheuristic methods avoid the generation of all possible states of the protein [6]. A particular class of these methods is known as *Ab-Initio*. In other words, *Ab-Initio* methods search for NS only using protein sequence amino acids.

New heuristic methods are used to solve PFP, where simulated annealing (SA) [7, 8] is one of the most successful. However, in order to generate high-quality solutions for PFP, new and more efficient SA should be designed [9]; one of them is named Multiquenching Annealing algorithm (MQA) [10]. This algorithm uses two phases. The first one or quenching phase applies a fast cooling rate to reach a fast solution. In contrast, the second phase applies a slow cooling rate in order to obtain a high-quality solution.

In this paper, a new approach named Chaotic Multi-Quenching Annealing (CMQA) for PFP is presented. CMQA has three phases. The first one applies a quenching process and chaotic functions in several subphases. The second phase implements an annealing process. In the third phase, the stochastic equilibrium is detected by using least squares method.

2. Materials and Methods

In this section the protein folding problem is briefly described and the next methods are explained: SA, MQA, and CMQA. Then chaotic local search (CLS) is introduced and compared with those algorithms trough a set of small proteins.

2.1. Protein Folding Problem. Native structure prediction of a protein is an enormous challenge in the computational biology domain [11, 12]. PFP is an interdisciplinary problem which involves molecular biology, biophysics, computational biology, and computer science [13]. In the case of *Ab-Initio*, NS prediction requires different mechanisms that lead the searching process to a unique biological three-dimensional structure. This process only requires amino acids' sequence. There is an extremely large space of possible conformations of the protein; the size of this space depends on the length of the sequence of amino acids [4].

The function of a protein directly is related to its three-dimensional structure, and misfolded proteins can cause a variety of diseases [14–19]. In addition, PFP is analyzed in protein engineering area [20] where proteins are designed and constructed with desired functions and structures. PFP can be solved by different combinatorial optimization algorithms [21]. An objective function of PFP would be optimized by finding the native structure of a protein. PFP requires the following information:

(i) a sequence of n amino acids $a_1, a_2, \ldots, a_n$ that represents the primary structure of a protein;

(ii) an energy function, $f(\sigma_1, \sigma_2, \ldots, \sigma_m)$, which represents the free energy. The variables $\sigma_1, \sigma_2, \ldots, \sigma_m$ represent the m dihedral angles.

The solution of this problem is to find the native structure such that $f^*(\sigma_1, \sigma_2, \ldots, \sigma_m)$ represents the minimal energy value. The optimal solution $\sigma^* = (\sigma_1, \sigma_2, \ldots, \sigma_m)$ defines the best three-dimensional configuration. Force fields are used to represent the energy of a protein [22]; some of the most common are AMBER [23], CHARMM [24], ECEPP/2, and ECEPP/3 [25]. These fields compute some energy components, for example, the electrostatic energy [25], the torsion energy [23], the hydrogen bond energy, and the Lennard-Jones energy [26].

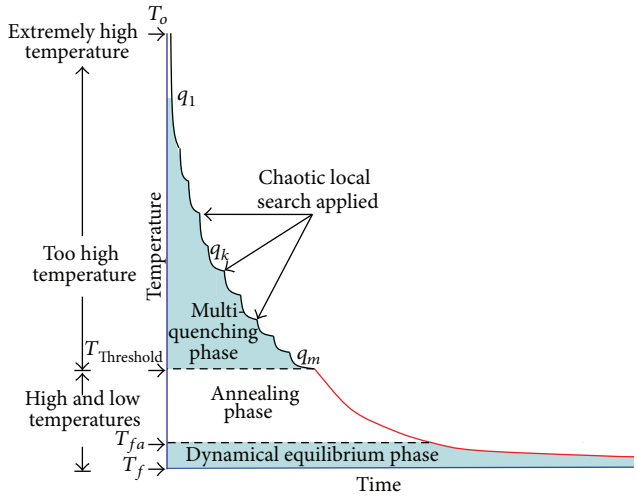


Figure 1: CMQA phases.

Simulated annealing algorithm has generated very good results for PFP [9, 27–29]. This method has been used in many combinatorial optimization problems [9, 10, 30–32]. However, SA has a low convergence feature and requires too much execution time. Thus, it is convenient to develop new SA strategies for improving its effectiveness.

2.2. Chaotic Multiquenching Annealing Algorithm

2.2.1. General Description. The Chaotic Multiquenching Annealing (CMQA) introduced in this paper is composed of three phases as it is shown in Figure 1: (i) multiquenching phase (MQP) applies several quenching processes, all of which implement a chaotic local search at the end of each stage; (ii) annealing phase (AP) is a classical simulated annealing process; and (iii) dynamical equilibrium phase (DEP) detects the stochastic equilibrium in a dynamical way using a regression method. MQP is applied from extremely high temperature to very high values. This phase applies a very fast cooling function to decrease the temperature parameter. MQP is executed from T_0 until $T_{\text{threshold}}$. After this phase, AP is executed until a final threshold temperature (T_{fa}), which is close to the final temperature of the whole algorithm. AP develops an exploration of the solution space with a very slow temperature's decrement. Finally, DEP detects the final temperature T_f by using an efficient implementation of the least squares method.

All CMQA's phases apply a cooling function (1), which is similar to that applied in the classical simulated annealing algorithm. The initial and final temperatures (T_0 and T_f) can be determined experimentally and/or analytically. The α parameter is a decrement temperature factor; it is less than one and greater than a certain value (close to 0.7) as follows:

$$T_{k+1} = \alpha T_k, \quad k = 0, 1, 2, \ldots, \ 0.7 \leq \alpha < 1. \tag{1}$$

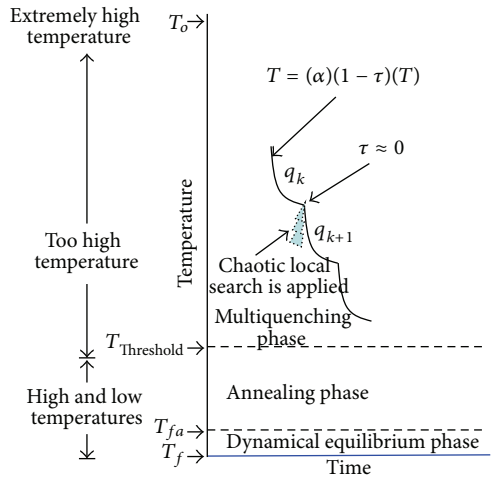


Figure 2: Chaotic local search.

2.2.2. Multiquenching Phase. MQP has several subphases (see Figure 2). It starts at extremely high initial temperature (T_0) and it is finished when a threshold temperature ($T_{\text{Threshold}}$) is reached. MQP uses the cooling function given by (2) and (3). In this case, the temperature is decreased by using $\alpha_{\text{Quenching}}$ and τ parameters. $\alpha_{\text{Quenching}}$ parameter is in the range $(0, 1)$, and it defines how fast each MQP's subphase is decreased. A very low value $\alpha_{\text{Quenching}}$ will decrease the temperature very fast. The τ parameter is ranged in $(0, 1)$, and it defines a quadratic decrement of the temperature. Notice that τ converges to zero, and then (2) is equivalent to (1) as follows:

$$T_{k+1} = \alpha_{\text{Quenching}}\,(1 - \tau)\,T_k, \quad k = 0, 1, 2, \ldots, \; 0.7 \leq \alpha < 1, \tag{2}$$

$$\tau = \tau^2, \quad 0 < \tau < 1. \tag{3}$$

The transition between two subphases is based on τ parameter. It occurs when τ converges to zero ($\tau \approx 0$). In this transition, a chaotic local search (CLS) is started. When CLS is finished, the new MQP subphase (i.e., another quenching process) is started, and τ is set to its initial value. This process continues until the temperature $T_{\text{Threshold}}$ is reached. Actually, this temperature corresponds to the initial temperature of a classical SA algorithm. Therefore, MQP is an additional search procedure that looks for improving the quality solution, even though the execution time is increased. An alternative approach is to increment the iterations' number of the classical SA. However, in this alternative, the quality solution is not significantly improved according to previous experimentation.

Algorithm 1 shows the MQP's pseudocode. In the setting section, MQP's parameters are established. The initial temperature T_0 is defined according to a tuning method [33], while $\alpha_{\text{Quenching}}$ and τ parameters experimentally are set (in this case 0.85 and 0.90, resp.). MQP generates a random initial solution S_i (with an energy $E(S_i)$) at the temperature T_0, which determines an initial minimal solution candidate (S_{min}). Two main cycles can be observed in this algorithm. The first one (external cycle) controls the temperature, which is decreased by applying geometric function (2). The other cycle (or metropolis cycle) generates new solutions S_j by using a perturbation function. This function is a classical probabilistic distribution at the beginning of the process, which is different to that used at the end of the algorithm when a chaotic search procedure is used. In the internal cycle, S_j is always accepted if it is better than a previous solution. When a new solution does not improve the previous one, it is accepted or rejected with the Boltzmann distribution. When S_j is accepted, it replaces the previous solution S_i (i.e., $S_i = S_j$). When a new accepted solution S_i is better than the current minimal solution S_{min}, it is replaced by S_i (i.e., $S_{\text{min}} = S_i$). Each time a metropolis cycle is finished, the τ parameter is updated according to (3), and when its value converges to zero, a chaotic local search (CLS) is executed. Once CLS (explained in Section 2.2.4) is finished, the τ parameter retakes its initial value and a new MQP subphase is started. In this case, a new temperature is calculated using (2), and the process continues until the $T_{\text{Threshold}}$ temperature is obtained.

The parameter T_0 is set to an initial value and is assigned to T (see line 4). The threshold temperature ($T_{\text{Threshold}}$) is set to an initial value (see line 5). $\alpha_{\text{Quenching}}$ and γ are set to initial value (see line 6). S_i is set to initial solution. $E(S_i)$ is calculated, which represents the energy of S_i (see line 8). S_{min} is set to S_i. The energy of S_{min} is set to $E(S_i)$. The external cycle is started (see line 11), and this is finished at line 30. The metropolis cycle is started within the temperature cycle (see line 12), and this cycle is finished at line 23. Within this cycle, S_j is created by applying a uniform perturbation (see line 13). The difference of energies between $E(S_j)$ and $E(S_i)$ is calculated (see line 14). If this difference is less than zero (see line 15), then the S_j is accepted (see line 16). Then, this solution S_j is assigned to S_i. If this difference is greater than zero, then the Boltzmann probability is calculated by using $e^{-(\text{difference}/T)}$ (see line 17). If this probability is greater than a random value between 0 and 1 (see line 17), then the S_j is accepted (see line 18). Then, this solution S_j is assigned to S_i. If S_i is less than S_{min} (see line 20), then S_i is assigned to S_{min} (see line 21). After the metropolis cycle is finished, the variable γ is updated by (3) (see line 24). If γ is very close to zero (see line 25), then γ is set to initial value (see line 26), and the chaotic search is called (see line 27). The temperature value T is set by applying (2) (see line 29).

2.2.3. Setting the Temperature Range. CMQA uses an analytical tuning method to determine the initial and final temperature [33]. This method is based on the acceptance probability of the solutions. At the beginning, the probability of accepting a new solution is very close to one. This occurs at extremely high temperatures; consequently, the deterioration of the cost function is maximal. Therefore, the initial temperature T_0 is associated with the maximal deterioration ΔZ_{max}. On the other hand, the probability of a new solution is very close to zero at very low temperatures; in this case, the deterioration of cost function is minimal. Thus, the final temperature T_f is associated with the minimal deterioration ΔZ_{min}. The acceptance probability based on Boltzmann distribution is

```
(1)  Multi-quenching Phase Procedure( )
(2)  Begin
(3)     //Setting section
(4)     T0 = Initial Temperature, T = T0
(5)     TThreshold = initial value
(6)     AlphaQuenching = initial value, tau = initial value
(7)     //Creation of initial solution
(8)     Si = Initial solution; E(Si) = Energy(Si);
(9)     Smin = Si; E(Smin) = E(Si)
(10)    //Multi-quenching Cycles
(11)    Repeat //External Cycle (Temperature Cycle)
(12)        Repeat //Internal Cycle (Metropolis Cycle)
(13)            Sj = Perturbation (Si) //Uniform perturbation
(14)            DE = E(Sj) − E(Si)
(15)            If DE ≤ 0 Then
(16)                Si = Sj
(17)            else if e^(−DE/T) > random [0,1] Then
(18)                Si = Sj
(19)            end if
(20)            If Si < Smin then //save Smin
(21)                Smin = Si; E(Smin) is saved
(22)            end if
(23)        Until Metropolis Cycle is Finish
(24)        tau = tau ^2
(25)        If (tau is very to close 0) Then
(26)            tau = initial value
(27)            Call Chaotic Search Procedure( )
(28)        end if
(29)        T = AlphaQuenching (1 − tau) T
(30)    Until T > TThreshold //External Cycle
(31) End procedure
```

Algorithm 1: MQP pseudocode.

defined by (4). At extremely high temperatures, this equation leads to (5). On the other hand, at the end of the process, the final temperature is obtained by (6) as follows:

$$P(\Delta Z) = \exp\left(\frac{-\Delta Z}{T}\right), \tag{4}$$

$$T_0 = \frac{-\Delta Z_{\max}}{\ln\left(P\left(\Delta Z_{\max}\right)\right)}, \tag{5}$$

$$T_f = \frac{-\Delta Z_{\min}}{\ln\left(P\left(\Delta Z_{\min}\right)\right)}. \tag{6}$$

Actually, CMQA uses the final temperature only as a guide to detect stochastic equilibrium at dynamical equilibrium phase. This phase is a special process based on least squares method during the last phase of CMQA. DEP is started some cycle before T_f and is explained in Section 2.2.6.

2.2.4. Chaotic Local Search. In order to avoid falling into local optima, CMQA applies CLS procedure at very high temperatures. As it is shown in Algorithm 2, this process has only a search cycle; $S_{\min}$ solution is improved by a chaotic function $f(x)$. This function is named chaotic perturbation in the pseudocode of Algorithm 2. The purpose of this chaotic function is to improve the possibility of escape from any local optimum. In CLS, S_j solution is generated by applying a chaotic perturbation to $S_{\min}$; when S_j is better than $S_{\min}$, then $S_{\min}$ is replaced by S_j. Thus, $S_{\min}$ solution is improved after several iterations (M_{chaot}). Generally, CLS improves $S_{\min}$ when M_{chaot} is equal to the number of instance's variables.

```
(1)  Chaotic Search Procedure
(2)  Begin
(3)      Saux ← Si
(4)      Si ← Smin
(5)      For k = 1 To Mchaot
(6)          Sj = Chaotic Perturbation (Si)
(7)          If Sj < Smin then
(8)              Smin ← Sj
(9)          End if
(10)         Si ← Smin
(11)     Next //end for
(12)     Si ← Saux
(13) End procedure
```

Algorithm 2: CLS pseudocode.

```
(1)  Annealing Phase Procedure
(2)  Begin
(3)     AlphaAnnealing = initial value
(4)     T = Final temperature of MQP (Threshold value)
(5)     T_final = very close to zero
(6)     Beta = value calculated by analytical method
(7)     MC = initial value
(8)     Repeat //External Loop
(9)          k = 1
(10)         Repeat //Internal Loop (Metropolis Cycle)
(11)              S_j = New solution (S_i)
(12)              DE = E(S_j) − E(S_i)
(13)              If DE ≤ 0 Then
(14)                   S_i = S_j
(15)              else if e^(DE/T) > random [0, 1] Then
(16)                   S_i = S_j
(17)              end if
(18)                   If S_i < S_min then
(19)                   S_min = S_i
(20)              end if
(21)         Until k < MC
(22)         T = AlphaAnnealing ∗ T
(23)         MC = Beta ∗ MC
(24)         k = k + 1
(25)    Until T > T_final
(26) End procedure
```

Algorithm 3: AP pseudocode.

The current solution S_i is assigned to S_{aux} (see line 3). The minimal solution S_{min} is assigned to S_i (see line 4). The FOR statement is started at line 5, and it is finished at line 11. Within this FOR statement, the solution S_j is created by applying a chaotic perturbation to S_i (see line 6). If solution S_j is better than S_{min}, then S_j replaces S_{min} (see line 8). The solution S_{min} is assigned to S_i (see line 10). After FOR statement is finished, S_{aux} is assigned to S_i.

2.2.5. Annealing Phase. The annealing phase (AP) corresponds to the classical simulated annealing algorithm and it is shown in Algorithm 3. When CMQA reaches its threshold level ($T_{Threshold}$), AP phase is started with the cooling function (1) using $\alpha_{Quenching}$ as decrement temperature factor. As it is known, AP phase contains two cycles, as it is common in classical SA. This pseudocode uses the same notation previously explained in Section 2.2.2.

2.2.6. Dynamical Equilibrium Based on Least Squares Method. CMQA algorithm dynamically finds the equilibrium by using least squares method. In order to obtain better solutions for PFP, this approach is applied after AP phase. Let (x, E_i) be a set of n points with $i = 1, 2, \ldots, n$. E_i represents the energy of protein at x_i point. The goal is to find a straight line, which is defined as $f(x_i) = ax_i + b$ that approximates the set of points, where a represents the slope of the straight line, and it is calculated by applying least squares method. The b parameter represents the intersection with axis. These parameters are calculated by

$$a = \frac{n\sum_{i=1}^{n} x_i E_i - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} E_i\right)}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2},$$
$$b = \frac{\left(\sum_{i=1}^{n} E_i\right)\left(\sum_{i=1}^{n} x_i^2\right) - \left(\sum_{i=1}^{n} x_i\right)\left(\sum_{i=1}^{n} \left(x_i E_i\right)\right)}{n\sum_{i=1}^{n} x_i^2 - \left(\sum_{i=1}^{n} x_i\right)^2}. \quad (7)$$

It is easy to show that the slope of the straight line can be calculated by

$$a = \frac{12\sum_{i=1}^{n} x_i E_i - 6(n-1)\sum_{i=1}^{n} E_i}{n^3 - n}. \quad (8)$$

CMQA determinates the dynamical equilibrium. This condition is obtained when the slope (a) of the straight line is very close to zero.

Table 1: Instances of PFP.

Instance of PFP	Amino acids	Mchaot (number of variables)
Met^5-enkaphalin	5	19
Proinsulin	31	132
T0549	73	343
T0335 (*Bacillus subtilis*)	85	450
T0281 (hypothetical protein) (1 WHZ)	90	458

3. Results and Discussion

CMQA is tested with five instances of PFP (see Table 1). These instances have different sequence's lengths and different number of variables (dihedral angles). The smallest sequence is Met^5-enkaphalin, which has five amino acids and 19 variables. The largest sequence is a hypothetical protein (CASP T0281), which has 90 amino acids and 458 variables. The proinsulin instance has 31 amino acids and 132 variables; the 2K5E (CASP T0549) has 73 amino acids and 343 variables. The instance *Bacillus subtilis* (CASP T0335) has 85 amino acids and 450 variables. The dihedral angles used in the simulations were phi (Φ), psi (Ψ), omega (ω), and Chi (χ).

Some parameters of MQP phase were determined experimentally. For example, the $\alpha_{Quenching}$ is set to 0.85 value; the initial value for τ is 0.90, and its final value is 0.0009. Different chaotic functions were tested for generating PFP solutions. These chaotic functions are (9), (10), (11), and (12). These equations are graphically shown in Figures 3, 4, 5, and 6, respectively. In AP phase, $\alpha_{Annealing}$ was fixed from different values taken from the range [0.75, 0.95] as follows:

$$f(x) = \sin\left(\frac{1}{x}\right), \quad (9)$$

$$f(x) = \sin\left(\left(\frac{1}{x}\right)\left(\frac{1}{1-x}\right)\right), \quad (10)$$

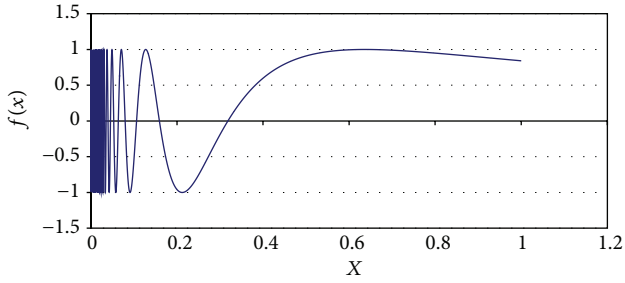


Figure 3: Chaotic function (9).

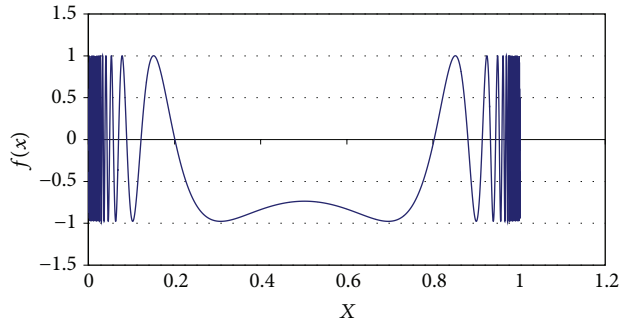


Figure 4: Chaotic function (10).

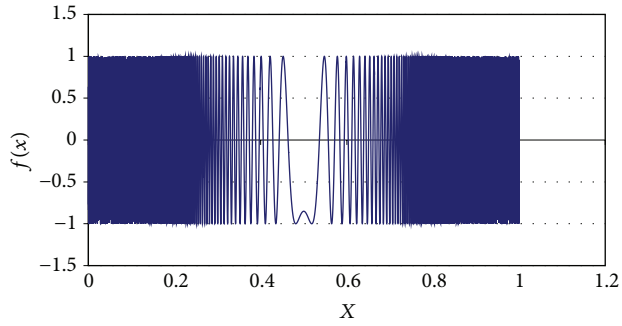


Figure 5: Chaotic function (11).

$$f(x) = \sin\left(\left(\frac{100}{x}\right)\left(\frac{1}{1-x}\right)\right), \quad (11)$$

$$f(x) = \sin\left(\frac{1}{x}\right) * \sin\left(\frac{5}{1-x}\right). \quad (12)$$

The results obtained are shown in Tables 2 to 6, which include information about the average energy of each protein (kcal/mol), its average processing time (minutes), and dRMSD. These results are grouped by $\alpha_{\text{Annealing}}$ values for each chaotic function. For Met^5-enkaphalin, the results are shown in Table 2. The best average solution for this protein was obtained by applying $\alpha_{\text{Annealing}} = 0.95$ and the chaotic function number 12; the best average energy was −5.4390 kcal/mol, with a processing time equal to 1.1191 minutes and dRMSD equal to 0.8913. Figure 7 shows the best solution with a dRMSD close to 0.88 and energy value equal to −7.1804 kcal/mol.

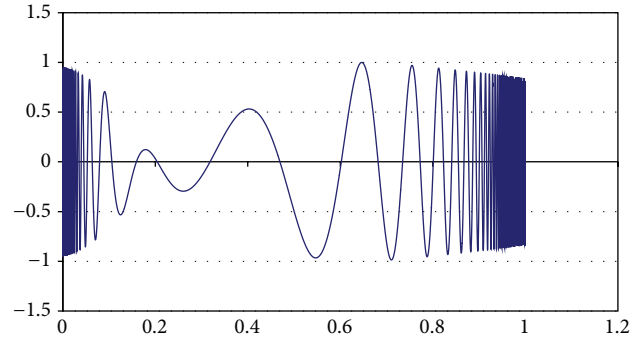


Figure 6: Chaotic function (12).

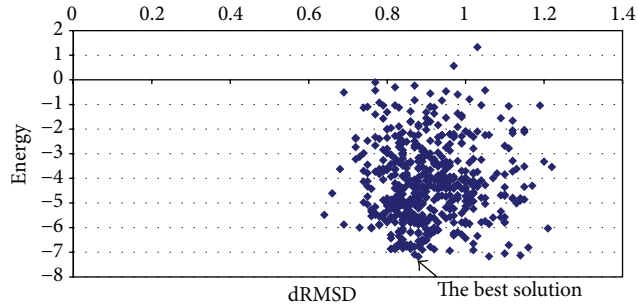


Figure 7: Graphic of average energy and dRMSD (Met^5-enkaphalin instance).

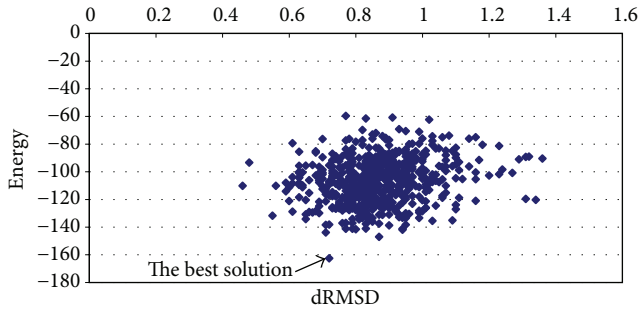


Figure 8: Graphic of average energy and dRMSD (proinsulin instance).

The results obtained for proinsulin are shown in Table 3. The best average solution for this protein was obtained by applying chaotic function number 12. The best solution has −126.9481 kcal/mol obtained with a processing time equal to 38.0507 minutes and dRMSD equal to 0.8233. Notice that the best results are obtained with high values. In Figure 8, the best solution is shown which has −162.5686 kcal/mol and a dRMSD equal to 0.72. The results obtained for T0549 instance are shown in Table 4. The solution with the best average energy is −269.6413 kcal/mol with processing time equal to 288.8558 minutes and dRMSD value of 0.72. Again, the best solution corresponds to the highest value of $\alpha_{\text{Annealing}}$ equal to 0.95. In this case, chaotic function number 10 provided the best results. The graphic of average energy versus dRMSD is shown in Figure 9. The solution with the best quality solution

Table 2: Average results of Met^5-enkaphalin.

$\alpha_{Annealing}$	Chaotic function	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
0.75	(9)	−3.2864	0.2535	0.8877
0.75	(10)	−4.0060	0.2105	0.9467
0.75	(11)	−3.3431	0.2082	0.9017
0.75	(12)	−3.4586	0.2514	0.9380
0.80	(9)	−3.0485	0.2959	0.9130
0.80	(10)	−4.3873	0.2459	0.9197
0.80	(11)	−4.2264	0.2447	0.9123
0.80	(12)	−3.9217	0.2981	0.8927
0.85	(9)	−3.7723	0.4014	0.9160
0.85	(10)	−4.6635	0.3365	0.8857
0.85	(11)	−4.7060	0.3332	0.8757
0.85	(12)	−4.2143	0.3957	0.8910
0.90	(9)	−3.7260	0.5581	0.8963
0.90	(10)	−4.7153	0.4626	0.8827
0.90	(11)	−4.6326	0.4627	0.8987
0.90	(12)	−4.8833	0.5585	0.8953
0.95	(9)	−5.0771	1.3507	0.8957
0.95	(10)	−4.9370	1.1181	0.9137
0.95	(11)	−5.4390	1.1191	0.8913
0.95	(12)	−5.3156	1.3501	0.8963

Table 3: Average results of proinsulin.

$\alpha_{Annealing}$	Chaotic function	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
0.75	(9)	−93.9999	7.8882	0.9127
0.75	(10)	−97.7679	6.3553	0.8643
0.75	(11)	−101.9142	6.3597	0.8703
0.75	(12)	−95.6412	7.9355	0.8960
0.80	(9)	−96.7255	9.3634	0.8830
0.80	(10)	−103.1905	7.5315	0.8847
0.80	(11)	−95.8967	7.5401	0.9290
0.80	(12)	−95.7312	9.3426	0.8920
0.85	(9)	−102.0535	12.0797	0.8523
0.85	(10)	−102.3225	9.7425	0.8893
0.85	(11)	−101.6467	9.7446	0.8590
0.85	(12)	−107.0401	12.1044	0.8933
0.90	(9)	−110.0378	18.3063	0.8427
0.90	(10)	−108.0935	14.7514	0.8503
0.90	(11)	−115.8930	14.7688	0.8503
0.90	(12)	−110.3555	18.3217	0.8310
0.95	(9)	−120.7662	47.1712	0.8503
0.95	(10)	−121.2029	38.0359	0.8550
0.95	(11)	−126.9481	38.0507	0.8233
0.95	(12)	−122.4787	47.2287	0.8240

has an energy value of −317.1750 kcal/mol with dRMSD value of 0.65.

The results obtained for T0335 instance are shown in Table 5. The best average solution for this instance is obtained by applying chaotic function number 11. The best average energy is −377.6919 kcal/mol with a processing time equal to 379.8146 minutes and RMSD value of 0.9787. In Figure 10, the graphic of average energy and dRMSD is shown. There is a solution with high quality (see arrow on graphics). The energy value is −455.0870 kcal/mol with dRMSD value of 0.76.

The results obtained for T0281 are shown in Table 6. The best average solution for this protein is obtained by

Table 4: Average results of T0549 instance.

$\alpha_{\text{Annealing}}$	Chaotic function	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
0.75	(9)	−180.1067	57.3851	0.7787
0.75	(10)	−184.8485	45.5662	0.8820
0.75	(11)	−188.2290	46.2738	0.8567
0.75	(12)	−187.8759	58.0133	0.8077
0.80	(9)	−187.5483	64.8901	0.8887
0.80	(10)	−194.7957	52.4376	0.8333
0.80	(11)	−204.7029	52.9562	0.8333
0.80	(12)	−194.0105	65.0954	0.7963
0.85	(9)	−212.1957	80.4466	0.7827
0.85	(10)	−213.3221	64.6595	0.8300
0.85	(11)	−220.9546	64.9489	0.8423
0.85	(12)	−212.0730	80.7839	0.8763
0.90	(9)	−236.1182	117.9315	0.8190
0.90	(10)	−241.1672	94.6274	0.8143
0.90	(11)	−230.1859	94.9217	0.8357
0.90	(12)	−230.0091	117.9894	0.8093
0.95	(9)	−269.6413	288.8558	0.7200
0.95	(10)	−263.9817	232.2564	0.7643
0.95	(11)	−262.1850	232.2011	0.8203
0.95	(12)	−262.4749	289.0106	0.8123

Table 5: Average results of T0335 instance.

$\alpha_{\text{Annealing}}$	Chaotic function	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
0.75	(9)	−267.9740	103.6328	1.0507
0.75	(10)	−273.8770	82.4738	0.9760
0.75	(11)	−270.0242	82.5450	1.0387
0.75	(12)	−281.0588	102.7148	0.9503
0.80	(9)	−285.2499	114.5852	0.9963
0.80	(10)	−293.6892	89.3586	0.9707
0.80	(11)	−287.6764	89.1567	0.9360
0.80	(12)	−296.9811	113.4518	1.0023
0.85	(9)	−305.8353	135.3040	1.0110
0.85	(10)	−305.2537	107.3173	1.0560
0.85	(11)	−300.5720	108.3275	0.9677
0.85	(12)	−300.6663	134.0739	0.9247
0.90	(9)	−329.4824	194.3791	0.9960
0.90	(10)	−334.7426	155.6107	0.8840
0.90	(11)	−327.1407	155.8174	0.9440
0.90	(12)	−324.0686	194.1348	0.9017
0.95	(9)	−368.4190	473.1948	0.9777
0.95	(10)	−377.6919	379.8146	0.9787
0.95	(11)	−372.4837	380.0762	0.9203
0.95	(12)	−375.3686	473.5348	1.0157

applying chaotic function number 13. The best average energy is −319.9603 kcal/mol with a processing time equal to 554.1053 minutes and dRMSD value of 2.9197. In Figure 11, the graphic of average energy and dRMSD is shown. In these graphics, all energy of T0281 calculated by CMQA is plotted. There is a solution with high quality (see arrow on graphics). The energy value is −403.3333 kcal/mol with dRMSD value of 3.03.

In order to compare the CMQA with other implementations, two algorithms were designed. The Multiquenching Annealing with dynamical equilibrium phase (MQA plus DEP) and classical simulated annealing were implemented. The results obtained are shown in Table 7. In general, CMQA obtained high-quality solutions in comparison with other implementations.

Table 6: Average results of T0281 instance.

$\alpha_{\text{Annealing}}$	Chaotic function	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
0.75	(9)	−206.5214	110.4517	2.9467
0.75	(10)	−215.2062	84.8609	2.9493
0.75	(11)	−205.6181	84.4000	2.7670
0.75	(12)	−211.9487	107.4365	2.9520
0.80	(9)	−224.1211	123.1385	2.9457
0.80	(10)	−233.2700	96.1731	2.9877
0.80	(11)	−223.8302	96.5483	2.9027
0.80	(12)	−222.5050	123.2675	2.8690
0.85	(9)	−251.9814	150.0813	2.9390
0.85	(10)	−245.7741	120.1284	2.8303
0.85	(11)	−251.0341	120.1144	2.8810
0.85	(12)	−259.9042	149.6338	2.8929
0.90	(9)	−273.9763	221.5904	2.8367
0.90	(10)	−260.1847	177.4887	2.8740
0.90	(11)	−281.4230	177.3376	2.9937
0.90	(12)	−290.0598	221.0355	2.8157
0.95	(9)	−314.9119	554.1073	3.000
0.95	(10)	−310.1975	444.6089	2.8633
0.95	(11)	−319.7511	444.5729	2.9873
0.95	(12)	−319.9603	554.1053	2.9197

Table 7: Comparison of results with other implementations.

Instance	Approach	Average energy (Kcal/mol)	Processing time (minutes)	Average dRMSD
Met	CMQA	−5.1922	1.2345	0.8993
Met	MQA plus DEP	−0.3775	2.8278	0.8883
Met	CSA	20.0864	0.0593	1.0267
Proinsulin	CMQA	−122.8490	42.6216	0.8382
Proinsulin	MQA plus DEP	−120.6576	24.8549	0.8357
Proinsulin	CSA	480.2667	1.9144	1.3263
T0549	CMQA	−264.5707	260.5810	0.7793
T0549	MQA plus DEP	−259.5423	187.5398	0.7277
T0549	CSA	1795.7408	12.9269	1.4320
T0335	CMQA	−373.4908	426.6551	0.9731
T0335	MQA plus DEP	−298.4703	130.3261	1.0453
T0335	CSA	3745.1859	3.3071	1.3413
T0281	CMQA	−316.2052	499.3486	2.9426
T0281	MQA plus DEP	−310.6578	407.8754	2.7654
T0281	CSA	2998.1609	22.6357	3.1280

4. Conclusions

In this paper, a new algorithm for protein folding problem named Chaotic Multiquenching Annealing or CMQA is proposed. In order to escape from local optima, this algorithm applies a chaotic function in each subphase of quenching. In addition, a very fast cooling function is applied in order to decrease the temperature values and change the subphase. During the multiquenching phase, solutions of PFP are generated in order to explore the solution space in a very fast way. An annealing phase is applied after the multiquenching phase. In this phase, a very slow cooling function is used in order to decrease temperature values. Besides, the annealing phase searches for solutions from high to lower temperatures. The last phase of CMQA is named dynamical equilibrium phase, in which slope values of energy are calculated using least squares method. The CMQA disadvantage is related to the processing time, which is increased in order to obtain high-quality solving. Therefore, processing time is sacrificed to achieve quality in solving the protein folding problem.

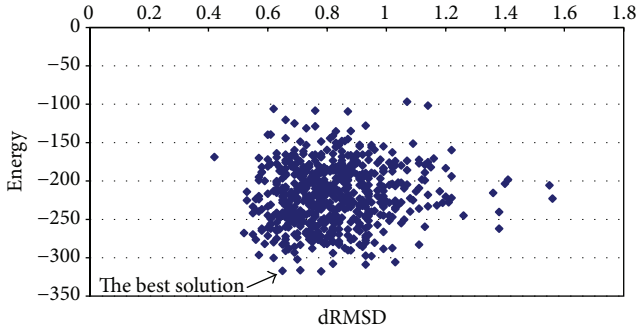


Figure 9: Graphic of average energy and dRMSD (T0549 instance).

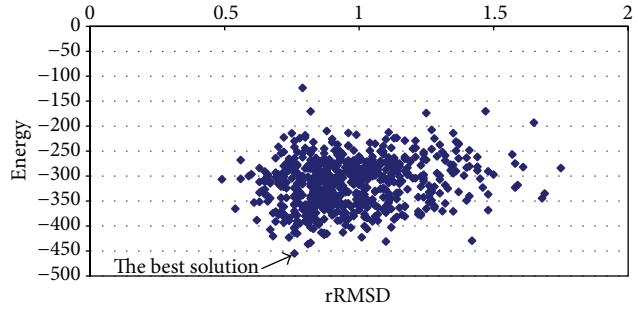


Figure 10: Graphic of average energy and dRMSD (T0335 instance).

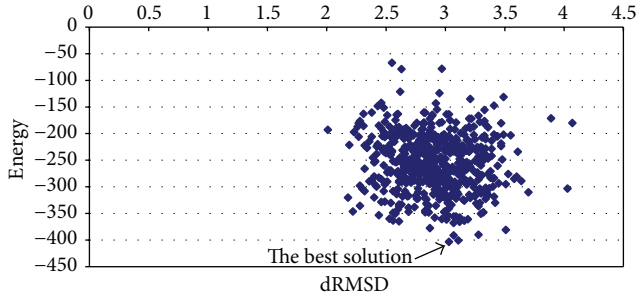


Figure 11: Graphic of average energy and dRMSD (T0281 instance).

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

Authors Juan Frausto-Solis and Ernesto Liñan-García contributed equally to development of this article. The author Mishael Sánchez-Pérez thanked the grant of DGAPA Postdoctoral Fellowships Program at CCG-UNAM.

References

[1] B. Lewin, *Genes 8*, Prentice Hall, New York, NY, USA, 2004.

[2] C. B. Anfinsen, "Principles that govern the folding of protein chains," *Science*, vol. 181, no. 4096, pp. 223–230, 1973.

[3] J. L. Sohl, S. S. Jaswal, and D. A. Agard, "Unfolded conformations of α-lytic protease are more stable than its native state," *Nature*, vol. 395, no. 6704, pp. 817–819, 1998.

[4] C. Levinthal, "Are there pathways for protein folding?" *Journal of Chemical Physics*, vol. 65, pp. 414–445, 1968.

[5] J. T. Ngo and J. Marks, "Computational complexity of a problem in molecular structure prediction," *Protein Engineering*, vol. 5, no. 4, pp. 313–321, 1992.

[6] M. M. Khimasia and P. V. Coveney, "Protein structure prediction as a hard optimization problem: the genetic algorithm approach," *Molecular Simulation*, vol. 19, no. 4, pp. 205–226, 1997.

[7] V. Černý, "Thermodynamical approach to the traveling salesman problem: an efficient simulation algorithm," *Journal of Optimization Theory and Applications*, vol. 45, no. 1, pp. 41–51, 1985.

[8] S. Kirkpatrick, C. D. Gelatt Jr., and M. P. Vecchi, "Optimization by simulated annealing," *Science*, vol. 220, no. 4598, pp. 671–680, 1983.

[9] J. Frausto-Solis, E. F. Román, D. Romero, X. Soberon, and E. Liñán-García, "Analytically tuned simulated annealing applied to the protein folding problem," in *Computational Science—ICCS 2007: Proceedings of the 7th International Conference, Beijing, China, May 27–30, 2007, Part II*, vol. 4488 of *Lecture Notes in Computer Science*, no. 2, pp. 370–377, 2007.

[10] J. Frausto-Solis, X. Soberon-Mainero, and E. Liñán-García, "MultiQuenching annealing algorithm for protein folding problem," in *MICAI 2009: Advances in Artificial Intelligence: Proceedings of the 8th Mexican International Conference on Artificial Intelligence, Guanajuato, México, November 9–13, 2009*, vol. 5845 of *Lecture Notes in Computer Science*, pp. 578–589, 2009.

[11] K. A. Dill, S. B. Ozkan, M. S. Shell, and T. R. Weikl, "The protein folding problem," *Annual Review of Biophysics*, vol. 37, pp. 289–316, 2008.

[12] B. P. Mukhopadhyay and H. R. Bairagya, "Protein folding: grand challenge of nature," *Journal of Biomolecular Structure and Dynamics*, vol. 28, no. 4, pp. 637–638, 2011.

[13] J. M. Yon, "Protein folding: a perspective for biology, medicine and biotechnology," *Brazilian Journal of Medical and Biological Research*, vol. 34, no. 4, pp. 419–435, 2001.

[14] C. Lee and M.-H. Yu, "Protein folding and diseases," *Journal of Biochemistry and Molecular Biology*, vol. 38, no. 3, pp. 275–280, 2005.

[15] C. A. Ross and M. A. Poirier, "Protein aggregation and neurogenerative disease," *Nature Medicine*, vol. 10, supplement, pp. S10–S17, 2004.

[16] M. Stefani, "Protein misfolding and aggregation: new examples in medicine and biology of the dark side of the protein world," *Biochimica et Biophysica Acta*, vol. 1739, no. 1, pp. 5–25, 2004.

[17] D. R. Howlett, "Protein misfolding in disease: cause or response?" *Current Medicinal Chemistry—Immunology, Endocrine and Metabolic Agents*, vol. 3, no. 4, pp. 371–383, 2003.

[18] J. Winderickx, C. Delay, A. De Vos et al., "Protein folding diseases and neurodegeneration: lessons learned from yeast," *Biochimica et Biophysica Acta*, vol. 1783, no. 7, pp. 1381–1395, 2008.

[19] T. K. Chaudhuri and S. Paul, "Protein-misfolding diseases and chaperone-based therapeutic approaches," *FEBS Journal*, vol. 273, no. 7, pp. 1331–1349, 2006.

[20] A. O.-L. F. Barona-Gomez and X. Soberon, "Advances and perspectives in protein engineering: from natural history to

directed evolution of enzymes," in *Advances in Protein Physical Chemistry*, pp. 407–438, 2007.

[21] S. Istrail and F. Lam, "Combinatorial algorithms for protein folding in lattice models: a survey of mathematical results," *Communications in Information and Systems*, vol. 9, no. 4, pp. 303–346, 2009.

[22] J. W. Ponder and D. A. Case, "Force fields for protein simulations," *Advances in Protein Chemistry*, vol. 66, pp. 27–85, 2003.

[23] G. Némethy, K. D. Gibson, K. A. Palmer et al., "Energy parameters in polypeptides. 10. Improved geometrical parameters and nonbonded interactions for use in the ECEPP/3 algorithm, with application to proline-containing peptides," *The Journal of Physical Chemistry*, vol. 96, no. 15, pp. 6472–6484, 1992.

[24] R. Brooks, R. Bruccoleri, B. Olafson, D. States, S. Swaminathan, and M. Karplus, "A program for macromolecular energy, minimization and dynamics calculations," *Journal of Computational Chemistry*, vol. 4, no. 2, pp. 187–217, 1983.

[25] F. A. Momany, R. F. McGuire, A. W. Burgess, and H. A. Scheraga, "Energy parameters in polypepltides. VII. Geometric parameters, partial atomic charges, nonbonded interactions, hydrogen bond interactions, and intrinsic torsional potentials for the naturally occurring amino acids," *The Journal of Physical Chemistry*, vol. 79, no. 22, pp. 2361–2381, 1975.

[26] A. D. Mackerell Jr., "Empirical force fields for biological macromolecules: overview and issues," *Journal of Computational Chemistry*, vol. 25, no. 13, pp. 1584–1604, 2004.

[27] J. Skolnick and A. Kolinski, "Computational studies of protein folding," *Computing in Science and Engineering*, vol. 3, no. 5, pp. 40–50, 2001.

[28] F. P. Agostini, D. D. O. Soares-Pinto, M. A. Moret, C. Osthoff, and P. G. Pascutti, "Generalized simulated annealing applied to protein folding studies," *Journal of Computational Chemistry*, vol. 27, no. 11, pp. 1142–1155, 2006.

[29] G. Ceci, A. Mucherino, M. D'Apuzzo et al., "Computational methods for protein fold prediction: an ab-initio topological approach," in *Data Mining in Biomedicine*, vol. 7 of *Springer Optimization and Its Applications*, pp. 391–429, 2007.

[30] L. Ingber, "Simulated annealing: practice versus theory," *Mathematical and Computer Modelling*, vol. 18, no. 11, pp. 29–57, 1993.

[31] Y. Li, C. E. M. Strauss, and A. Gorin, "Parallel tempering in rosetta practice," in *Advances in Bioinformatics and Its Applications*, vol. 3072, pp. 380–389, 2004.

[32] J. Mingjun and T. Huanwen, "Application of chaos in simulated annealing," *Chaos, Solitons and Fractals*, vol. 21, no. 4, pp. 933–941, 2004.

[33] H. Sanvicente-Sánchez and J. Frausto-Solís, "A method to establish the cooling scheme in simulated annealing like algorithms," in *Computational Science and Its Applications—ICCSA 2004: Proceedings of the International Conference, Assisi, Italy, May 14–17, 2004, Part III*, vol. 3045 of *Lecture Notes in Computer Science*, pp. 755–763, 2004.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 730712, 7 pages
http://dx.doi.org/10.1155/2014/730712

Research Article

Towards Application of One-Class Classification Methods to Medical Data

Itziar Irigoien,[1] Basilio Sierra,[1] and Concepción Arenas[2]

[1] *Department of Computer Sciences and Artificial Intelligence, UPV/EHU, 20018 Donostia, Spain*
[2] *Department of Statistics, UB, 08028 Barcelona, Spain*

Correspondence should be addressed to Basilio Sierra; b.sierra@ehu.es

Received 5 December 2013; Accepted 24 February 2014; Published 20 March 2014

Academic Editors: V. Bhatnagar and Y. Zhang

In the problem of one-class classification (OCC) one of the classes, the target class, has to be distinguished from all other possible objects, considered as nontargets. In many biomedical problems this situation arises, for example, in diagnosis, image based tumor recognition or analysis of electrocardiogram data. In this paper an approach to OCC based on a typicality test is experimentally compared with reference state-of-the-art OCC techniques—Gaussian, mixture of Gaussians, naive Parzen, Parzen, and support vector data description—using biomedical data sets. We evaluate the ability of the procedures using twelve experimental data sets with not necessarily continuous data. As there are few benchmark data sets for one-class classification, all data sets considered in the evaluation have multiple classes. Each class in turn is considered as the target class and the units in the other classes are considered as new units to be classified. The results of the comparison show the good performance of the typicality approach, which is available for high dimensional data; it is worth mentioning that it can be used for any kind of data (continuous, discrete, or nominal), whereas state-of-the-art approaches application is not straightforward when nominal variables are present.

1. Introduction

In one-class classification (OCC), the problem is to classify data when information is available for only one group of observations. Specifically, given one set of data, called the target class, the aim of the OCC methods is to distinguish data belonging to the target class from other possible classes. OCC can be seen as a special type of two-class classification problem, when data from only one class is considered. This is an interesting problem because there are many real situations where a representative set of labeled examples for the second class is too costly, difficult to obtain, or not available at all. This situation can occur, for instance, in medical diagnosis, where data from healthy or even from nonhealthy patients are extremely hard or impossible to obtain: for example, through mammograms for breast cancer detection [1, 2], the one-class recognition of cognitive brain functions [3], in prediction of protein-protein interactions [4], in the lung tissue categorization of patients affected with interstitial lung diseases [5], or in the identification of patients with one or more Nosocomial infections using clinical and other data collected during the survey [6]. Several approaches to OCC have been presented and good overviews can be found in [7–10]. Some of the OCC approaches estimate the density of the reference data and set a threshold on this density, using a Gaussian model, a mixture of Gaussians models, or the Parzen density estimators [11, 12]. Boundary methods, as the k-centers, NN-d [13, 14], and support vector machine SVM [15–18], cover the data set with k small balls with equal radii and they make assumptions about the clustering characteristics of the data or their distribution in subspaces. These methods only achieve good results when the target data have the same distribution tendency in all orientations [19]. The reconstruction methods (k-mean clustering, self-organizing maps, PCA, mixtures of PCAs, and diabolo networks density) make assumptions about the clustering characteristics of the data or their distribution in subspaces, and a set of prototypes is needed (see, e.g., [20]). Many of these methods have data-specific parameters or assume that data follow a specific model; therefore data knowledge is necessary. One-class classification can also be considered as outlier detection, where the classification model can be used

to detect the units deviating significantly from the target class. There are some distance-based outlier detection methods [21, 22], which need the computation of the distances between units in the target class and distances between a new unit and their neighbors in the target class, but in contrast with other OCC methods they are more flexible. Some other state-of-the-art methods are neural networks [23], Bayesian neural networks [24], or Naive Bayesian classifiers [25]. Recently, in [26] the authors formulated a typicality test and this approach is here applied to the OCC problem. Thus, objects in the target class can be considered as typical, while objects in the negative class can be considered as atypical. In order to evaluate the viability of the typicality approach a comparative study is presented. Five reference state-of-the-art techniques, two parametric density methods, the Gaussian and mixture of Gaussians procedures; two nonparametric density methods, the Parzen and naive Parzen procedures; a boundary method, the support vector data description method, are experimentally compared with the typicality approach using biomedical data sets.

The paper is organized as follows. Section 2 presents the six considered OCC procedures that are evaluated for twelve real biomedical data set. The experimental study is summarized in Section 3, while conclusions are drawn in Section 4.

2. One-Class Classification

In this section, the one-class classification problem is formally stated, and the six considered procedures are reviewed.

2.1. The One-Class Classification Problem. Consider a class C, the target class, containing n objects and represented by a S-random vector $\mathbf{Y}$ with probability density function f with respect to a suitable measure λ. Let an object of C be represented by a vector $\mathbf{y}$ containing the values of the measures in p features, not necessarily continuous. The OCC problem can be defined as the problem of assigning or not a new object $\mathbf{y}_0$ to the target class C, when data only from the target class is available. Thus, from data in the target class a classification model should be constructed. The OCC procedures usually consider a training phase using the so-called training data set; that is, either the probability density function or the parameters of the classifier's model should be determined. In OCC the training data set contains only the observations belonging to C, while the testing data set includes the observations from class C and other possible class C'. As in medical care correct diagnosis is very important, it is necessary to evaluate the OCC models, which can be considered as a case/noncase diagnosis where the target class is, for instance, the case class. This diagnosis will misclassify some cases as noncases and some noncases as cases. These two types of misclassifications lead to two important aspects of the performance of the diagnosis, sensitivity, and specificity. As it is known, the sensitivity or true positive rate is the probability that occurs if an object in class C is classified as belonging to this class. The specificity or true negative rate is the probability that occurs if an object not belonging to C is classified as not belonging to C. A very common way of displaying the values of the sensitivity and specificity is by the ROC curve (Receiver Operating Characteristic), which represents the pairs (1-specificity, sensitivity). Therefore, the area under the ROC curve, the AUC, lies between 0 and 1 and takes value 1 for a perfect diagnosis and the value 0.5 for random diagnosis, so that AUC values will be useful to evaluate the performance of OCC models [27].

It is important to note that in one-class classifiers the ability to learn the true characteristics of the data in presence of noise or errors in the feature values is specially important. Furthermore, the number of parameters to be estimated by users should be minimized, and the computational and storage requirements must be in consideration, as there are limiting factors in the use of some of the methods. Finally, one-class classifiers are determined in the training phase using the training data set; thus the standard OCC procedures may be affected by initial settings.

Next, six one-class classification methods will be reviewed. We consider five well known and reference OCC methods: two parametric density methods, the Gaussian and mixture of Gaussians; two nonparametric density methods, the Parzen and naive Parzen; a boundary method, the support vector data description. For these methods, we summarized some of their characteristics and references for more details about the construction of the classification model and properties are given. Finally, a nonparametric typicality approach based on distances is considered. As this method has not yet been considered as a one-class classification procedure, more details about the classification model and properties will be included.

2.2. Gaussian and Mixture of Gaussians. The Gaussian and mixture of Gaussians methods assume that the data is distributed according to the normal distribution or to a mixture of n_G normal distributions [9]. The parameters of the Gaussian model can be found by maximizing the likelihood function over the training data set, being the learning process computationally inexpensive. For the mixture of Gaussians, the parameters can be found efficiently by the EM algorithm. Thus, the learning process using the EM algorithm is more computationally demanding as a number of interactions should be done before the algorithm converges. The methods based on Gaussian models are sensitive to the noise in the training data set, as the noise introduces a significant bias to the estimate covariance matrix. Furthermore, these procedures present a rather high sensitivity to errors in feature values and outliers. In the learning phase, the storage requirements are rather high but very low in the classification phase.

2.3. Parzen and Naive Parzen. Parzen and naive Parzen density estimation are nonparametric procedures and do not need any assumption about the data distribution [6, 28, 29]. The density is estimated directly from the training data and is a function of the number of objects situated in a region of a specific volume with a value h as the length of an edge. The value of h plays the role of a smoothing parameter.

An advantage of the method is that it does not need any estimation of parameters. However, too long values of the smoothing parameter h imply an oversmoothed estimated density. When h is too small then the estimated density contains noise. Furthermore, the method needs to store all the observation vectors and it makes it slower, presenting very low computational requirements of learning but rather high in classification. The method is relatively robust to the outliers in the training data, choosing appropriate distance, and presents rather high sensitivity to errors in feature values. These procedures need to estimate one parameter by the users.

2.4. One-Class Support Vector Data Description. Support vector data description (SVDD) is a boundary method [9, 17]. It defines a hypersphere with a minimum volume covering the entire training data set. The minimization is solved as a quadratic programming problem and can be solved efficiently by introducing Lagrange multipliers [30, 31]. The method is relatively resistant to noise. The number of parameters that are to be estimated is equal to the size of the training data set; thus it is not useful for large training data sets. SVDD presents rather low sensitivity to errors in feature values and outliers. The method presents very high computational requirements of learning but very low in classification and needs to estimate one parameter by the user and learnt the other parameters.

2.5. Typicality Approach. Consider a target class C containing n units measured on p features. Let $\delta(\mathbf{y}, \mathbf{y}')$ be a distance [32] function on S. It is said that δ is an Euclidean distance function if the metric space (S, δ) can be embedded in an Euclidean space $R^q, \Psi : \mathscr{R} \rightarrow R^q$, such that $\delta^2(\mathbf{y}, \mathbf{y}') = \|\Psi(\mathbf{y}) - \Psi(\mathbf{y}')\|^2$, and we may understand $E(\Psi(\mathbf{Y}))$ as the δ-mean of $\mathbf{Y}$. There are various ways of achieving this situation, the most common probably being classical metric scaling, also known as principal coordinate analysis [33, 34]. Given the real-valued coordinates $\mathbf{Z} = \Psi(\mathbf{Y})$, it is possible to apply any standard multivariate technique. Such an approach was used by different authors [35–42]. In this context a general measure of dispersion of $\mathbf{Y}$, the geometric variability V_δ of C, with respect to δ can be defined by

$$V_\delta(C) = \frac{1}{2}\int_{S\times S} \delta^2\left(\mathbf{y}_i, \mathbf{y}_j\right) f\left(\mathbf{y}_i\right) f\left(\mathbf{y}_j\right) \lambda\left(d\mathbf{y}_i\right) \lambda\left(d\mathbf{y}_j\right) \quad (1)$$

which is a variant of Rao's diversity coefficient [43]. The proximity function of a unit $\mathbf{y}_0$ to C is defined as

$$\phi^2(\mathbf{y}_0, C) = \int_S \delta^2\left(\mathbf{y}_0, \mathbf{y}_j\right) f\left(\mathbf{y}_j\right) \lambda\left(d\mathbf{y}_j\right) - V_\delta(C). \quad (2)$$

In applied problems, the distance function is a datum, but the probability distribution for the population is unknown. Natural estimators given a sample $\mathbf{y}_1, \ldots, \mathbf{y}_n$ coming from C are

$$\widehat{V}_\delta(C) = \frac{1}{2n^2}\sum_{i,j}\delta^2\left(\mathbf{y}_i, \mathbf{y}_j\right),$$
$$\widehat{\phi}^2(\mathbf{y}_0, C) = \frac{1}{n}\sum_i \delta^2(\mathbf{y}_0, \mathbf{y}_i) - \widehat{V}_\delta(C), \quad (3)$$

for the geometric variability of C and the proximity function of unit $\mathbf{y}_0$ to C, respectively.

See [44] and references therein for a review of these concepts, their application, different properties, and proofs.

Let $\mathbf{y}_0$ be a new observation and consider the OCC problem to decide whether $\mathbf{y}_0$ belongs to the target class C or, on the contrary, it is an outlier or an atypical observation, belonging to some different and unknown class. Therefore, the OCC problem can be formulated as a hypothesis test with

H_0: $\mathbf{y}_0$ comes from the target class C with δ-mean $E(\Psi(\mathbf{Y}))$,

H_1: $\mathbf{y}_0$ comes from another unknown class.

This test can be considered as a test of typicality, as is formulated in [26]. In our context, with only one known class, the typicality test reduces to compute $\phi^2(\mathbf{y}_0)$. If $\phi^2(\mathbf{y}_0)$ is significant it means that $\mathbf{y}_0$ comes from a different and unknown class.

Sampling distribution of $\phi^2(\mathbf{y}_0)$ can be difficult to find for mixed data, but nevertheless it can be obtained by resampling methods, in particular drawing bootstrap samples: draw N units $\mathbf{y}$ with replacement from C and calculate the corresponding $\phi^2(\mathbf{y})$ values; repeat this process 10P times, with $P \geq 1$. In this way, the bootstrap distribution under H_0 is obtained.

It is worth to point out that this procedure can be used for any kind of data (continuous, discrete, or nominal), whereas other approaches application is not straightforward when nominal variables are present. As the procedure needs the computation of the distances between units in the target class and distances between a new unit and the units in the target class the storage requirements are rather high but very low in the classification phase. The method is relatively robust to the outliers in the training data.

3. Results of the Experimental Study

As there are few benchmark data sets for OCC, we use data sets containing two or multiple classes. Each class in turn is considered as the target class and the units in the other classes are considered as new units to be classified. On the one hand, we used 10 biomedical data sets, none of them containing nominal variables, from the UCI machine learning repository [45] to evaluate the performance of all the above procedures. In order to perform the comparison, the selected data sets are the biomedical data used in [46]; only the target classes considered in that referred work are taken into account in this paper as well. On the other hand, we also applied the typicality approach to two data sets with mixed variables.

In our experiments, we followed the procedure stated in [46]. Thus, all multiclass problems are transformed to one-class classification problems by setting a chosen class as a target class and all remaining classes as nontargets. The target class was randomly split into equal parts between the training and test sets. All one-class classifiers were only trained on the target data, that is, the half of the target data, and tested on the test data, the remaining half of the target data and the nontarget data. The experiments were repeated 10 times

and the AUC average and standard deviation values are reported.

3.1. Results of Data Sets without Nominal Variables. In Table 1, a brief description of these well known data sets is presented. For the typicality method, a suitable distance is selected for each data set, according to the type of data (see Table 2 last column). The considered distances were the Euclidean distance, the Euclidean distance after standardized the data, the Mahalanobis distance, or the correlation distance.

With *breast Wisconsin prognostic*, *E. coli*, *hepatitis,* and *liver disorders* data sets, the typicality model obtained similar AUC average values than the other procedures, as we can see in Table 2. For the *breast Wisconsin origin* data set and taking benign class as target class, the typicality procedure obtained very good results (99.4±0.2) and similar to the obtained by the other procedures. It is worth noting that, for the malignant class as target class, it obtained similar results (97.6 ± 0.5) than the naive Parzen procedure (96.5 ± 0.4) and much better results than the obtained for the other procedures. With the *colon* data set, while mixture of Gaussians is not available and Parzen or SVDD methods gave poor results with high variability (63.6 ± 22.4, 36.4 ± 22.4 for classes 1 and 2, resp.), the typicality method obtained clearly better results (75.4 ± 6.3, 78.3 ± 5.8 for classes 1 and 2, resp.) than Gaussian (61.1 ± 3.8, 70.4 ± 1.1 for class 1 and 2, resp.) or naive Parzen (73.4 ± 3.1, 70.0 ± 1.5 for classes 1 and 2, resp.) methods. When the *leukemia* data set was analyzed, mixture of Gaussians and Parzen procedures were not available at all, and SVDD procedure presented a large variability (58.9±30.2 and 41.1±30.2, resp.). However, similar results were found for Gaussian, naive Parzen, or typicality procedures. For the *METAS* data set, it must point out that when the second class was the target class, the best results were obtained with the typicality procedure (64.5 ± 4.7), showing its good performance with high dimensional data sets. With the *SPECT heart* data set and using the typicality method, a little worse results were found when *class 0* was the target class. However, when the target class was *class 1*, clearly the typicality procedure obtained the best results (69.8 ± 2.5). Finally, for the *thyroid* data set, the typicality results were similar or slightly better than those obtained by the other procedures.

In summary, from the results presented in Table 2 it is clear that, in general, the typicality approach performs equal or better than the other well known procedures, for all the considered UCI data sets. The results show that, while other procedures are affected by small target classes, the typicality approach is more robust. Furthermore, it performs well with high-dimensional data. On the other hand, as shown in Table 2, state-of-the-art algorithms give "NaN"—Not a Number—in some cases; this fact does not appear when the typicality approach is used. Additional statistics on the AUC average values are provided in Figure 1 under the form of boxplots. Black lines correspond to the median values and black segments to the minimum and maximum values of each method. As we can see, the typicality procedure is the more robust for all data sets and it is in the top best methods.

Table 1: Description of ten UCI data sets used in the experiments.

Data sets	Classes	Instances	Features
Breast Wisconsin original	2	241/458	9
Breast Wisconsin prognostic	2	47/151	33
Colon	2	40/22	1908
E. coli	2	52/284	8
Hepatitis	2	123/32	19
Leukemia	2	47/25	3571
Liver disorders	2	145/200	6
METAS	2	46/99	4919
SPECT heart	2	95/254	44
Thyroid	3	93/191/3488	21

3.2. Results on Mixed Variables Data Sets. Next we report the results obtained using two data sets with mixed variables. That means that there are some quantitative, binary, and nominal variables. Therefore, methods that implicitly are based on the Euclidean distance are not adequate. Thus, only the typicality approach was performed with these two data sets. In presence of mixed variables, it is known that Gower's distance is an appropriate distance, presenting good properties in terms of missing values [47, 48].

Statlog (Heart) Data Set. This data set is available in the UCI dataset repository. It is composed by 270 units classified in two classes: *absence* or *presence* of heart disease, with 150 and 120 units, respectively. There are 13 variables, 6 quantitative, 1 ordered, 3 binary, and 3 nominal, and no missing values are present. Taking in turn, *absence* and *presence* class as the target class, the typicality approach reported AUC average and standard deviation values 86.08 ± 2.03 and 84.53 ± 1.62, respectively. Furthermore, Table 3 reports the results obtained when we attempt to achieve a fixed False Alarm Rate (FAR) or false negative rate (1-sensitivity), namely, 0.1. Note that for the two target class, we obtain good results.

Liver Cancer Data Set. We apply the typicality approach to a liver cancer data set [49]. It consists of 213 cases described by 4 nominal variables (type of hepatitis, categorized age, sex, and whether cirrhosis is present) plus 1993 genes. It is worth to mention that for each case at least one missing value is present (9.6% of the values are missing). The data set is divided in three groups. Group T formed by 107 samples from tumors on liver cancer patients, group NT formed by 76 samples from nontumor tissues of liver cancer patients and group N formed by 30 samples from normal livers. In [42] it was shown that there exists a high degree of confusion between groups, so bad one-class classification results are expected. Taking groups N, NT, and T as target classes, the typicality approach obtained AUC average values and standard deviation values 86.04 ± 3.95, 80.86 ± 3.07, and 55.62 ± 3.76, respectively. Results obtained for a fixed FAR equal to 0.1 are reported in Table 4. From Table 4, we can observe that when T is the target class, the method cannot distinguish the other groups. When NT is the target class, units from N group are not distinguished

Table 2: AUC average and standard deviation, in brackets, values on UCI data sets. In the last column the distance used by the typicality method is indicated: c: correlation, E: Euclidean, E-st: Euclidean after standardization, and M: Mahalanobis.

Data sets	Target class	Gaussian	Mixture Gaussians	Naive Parzen	Parzen	Support vector DD	Typicality distance
Breast Wisconsin original	Benign	98.5 (0.1)	98.3 (0.2)	98.7 (0.1)	99.2 (0.1)	99.0 (0.1)	99.4 (0.2)—E
	Malignant	82.3 (0.2)	69.1 (3.2)	96.5 (0.4)	72.3 (0.5)	66.1 (0.8)	97.6 (0.5)—E
Breast Wisconsin prognostic	Returning	63.0 (1.4)	59.1 (1.6)	59.0 (1.9)	59.4 (1.9)	59.6 (1.4)	58.5 (5.4)—M
	Nonreturning	50.8 (0.8)	52.6 (1.6)	53.8 (2.2)	52.2 (1.7)	51.7 (1.7)	55.6 (2.9)—M
Colon	1	61.1 (3.8)	NaN	73.4 (3.1)	63.6 (22.4)	63.6 (22.4)	75.4 (6.3)—c
	2	70.4 (1.1)	NaN	70.0 (1.5)	36.4 (22.4)	36.4 (22.4)	78.3 (5.8)—c
E. coli	Periplasm	92.9 (0.3)	92.0 (0.4)	93.0 (0.8)	92.2 (0.4)	89.4 (0.8)	95.4 (1.3)—E
Hepatitis	Normal	82.1 (1.0)	78.3 (1.0)	80.1 (0.7)	79.0 (1.0)	78.7 (1.1)	80.8 (2.2)—M
Leukemia	1	92.1 (1.8)	NaN	90.2 (4.4)	NaN	58.9 (30.2)	91.2 (3.4)—c
	2	94.7 (2.7)	NaN	96.7 (0.4)	NaN	41.1 (30.2)	90.6 (3.9)—c
Liver disorders	Class 1	58.5 (0.4)	59.3 (0.7)	61.4 (0.7)	58.7 (0.4)	59.0 (0.9)	58.1 (2.5)—M
	Class 2	50.9 (0.5)	49.4 (0.6)	48.4 (0.8)	46.9 (0.8)	49.6 (1.0)	58.0 (3.7)—M
METAS	1	69.1 (1.5)	NaN	65.3 (0.8)	64.8 (21.5)	64.8 (21.5)	67.3 (2.3)—c
	2	36.4 (1.4)	NaN	40.7 (1.2)	35.2 (21.5)	35.2 (21.5)	64.5 (4.7)—c
SPECT heart	Class 0	93.4 (0.9)	95.1 (0.8)	90.7 (1.5)	95.7 (1.0)	89.7 (3.2)	86.1 (3.8)—M
	Class 1	28.4 (0.5)	27.9 (1.3)	26.0 (0.7)	44.5 (0.5)	57.1 (11.1)	69.8 (2.5)—M
Thyroid	Normal	84.3 (0.0)	84.7 (4.4)	96.1 (0.0)	90.6 (0.0)	56.0 (0.0)	98.1 (1.2)—c
	Hyperthyroid	70.3 (0.0)	68.1 (0.9)	75.1 (0.0)	70.6 (0.0)	45.7 (0.0)	65.9 (2.5)—c
	Subnormal	69.6 (0.0)	81.5 (1.0)	84.4 (0.0)	87.4 (0.0)	50.3 (0.0)	88.0 (2.8)—c

Table 3: For Statlog (heart) data set and using the typicality approach, false and true positive, and negative values, for a fixed False Alarm Rate equal to 0.1.

Target class	Classified as	Tested classes	
		Absence	Presence
Absence	Target	135/150	41/120
	Nontarget	15/150	79/120
Target class	Classified as	Tested classes	
		Presence	Absence
Presence	Target	110/120	68/150
	Nontarget	10/120	82/150

Table 4: For Liver cancer data set and using the typicality approach, false and true positive, and negative values, for a fixed False Alarm Rate equal to 0.1.

Target class	Classified as	Tested classes		
		N	NT	T
N	Target	29/30	56/76	16/107
	Nontarget	1/30	20/76	91/107
Target class	Classified as	Tested classes		
		NT	N	T
NT	Target	72/76	28/30	48/107
	Nontarget	4/76	2/30	59/107
Target class	Classified as	Tested classes		
		T	N	NT
T	Target	102/105	30/30	75/76
	Nontarget	3/105	0/30	1/76

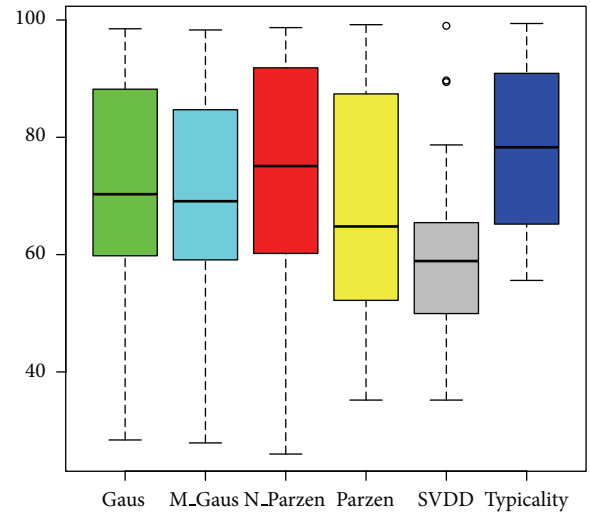


Figure 1: Boxplots of AUC average values of the OCC methods over the experimental data sets.

and only half the units from T are distinguished properly as nontarget. When N is the target class, units from T are very well distinguished as nontarget.

4. Conclusions

A noticeable attention has been devoted to the one-class classification problem in the last years. This type of classification is characterized by the use of observations belonging to only one known class. These methods are particularly useful in

biomedical studies, when observations belonging to other classes are difficult or impossible to obtain. In this paper, reference state-of-the art one-class classification methods have been reviewed, and their suitability has been compared with a recent typicality procedure. To assess the efficiency of this new typicality application, experiments have been conducted on several public data sets from the UCI repository and has been compared to five of the most OCC used procedures, namely, Gaussian, mixture of Gaussians, naive Parzen, Parzen, and support vector DD models [46]. The results show that the typicality approach performs equally well or better than these state-of-the art procedures, thus it will be very valuable in many biomedical applications. The typicality approach does not need any knowledge about the data distribution, does not estimate any parameter, and is applicable to any kind of data, not necessarily continuous. This approach performs well with high dimensional data and it is robust in front of small target classes, whereas other OCC method accuracy rates are not so stable. For all these reasons, the typicality approach can be very useful in many biomedical applications where clinical, pathological, or biological noncontinuous data can be found and where data from healthy or even from nonhealthy patients are extremely hard or impossible to obtain.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgments

This work was supported by the Basque Government Research Team Grant (IT313-10) SAIOTEK Project SA-2013/00397 and the University of the Basque Country UPV/EHU (Grant UFI11/45 (BAILab)).

References

[1] L. Tarassenko, P. Hayton, N. Cerneaz, and M. Brady, "Novelty detection for the identification of masses in mammograms," in *Proceedings of the 4th International Conference on Artificial Neural Networks*, pp. 442–447, June 1995.

[2] M. Costa and L. Moura, "Automatic assessment of scintmammographic images using a novelty filter," *Proceedings of the Annual Symposium on Computer Applications in Medical Care*, pp. 537–541, 1995.

[3] O. Boehm, D. R. Hardoon, and L. M. Manevitz, "Classifying cognitive states of brain activity via one-class neural networks with feature selection by genetic algorithms," *International Journal of Machine Learning and Cybernetics*, vol. 2, no. 3, pp. 125–134, 2011.

[4] J. A. Reyes and D. Gilbert, "Prediction of protein-protein interactions using one-class classification methods and integrating diverse biological data," *Journal of Integrative Bioinformatics*, vol. 4, no. 3, p. 77, 2007.

[5] A. Depeursinge, J. Iavindrasana, A. Hidki et al., "Comparative performance analysis of state-of-the-art classification algorithms applied to lung tissue categorization," *Journal of Digital Imaging*, vol. 23, no. 1, pp. 18–30, 2010.

[6] G. Cohen, H. Sax, and A. Geissbuhler, "Novelty detection using one-class Parzen density estimator. An application to surveillance of nosocomial infections," *Studies in Health Technology and Informatics*, vol. 136, pp. 21–26, 2008.

[7] D. M. J. Tax, *One-class classification [Ph.D. thesis]*, Delft University of Technology, 2001.

[8] C. Dsir, S. Bernard, C. Petitjean, and L. Heutte, "One class random forests," *Pattern Recognition*, vol. 46, pp. 3490–3506, 2013.

[9] O. Mazhelis, "One-class classifiers: a review and analysis of suitability in the context of mobile-masquerader detection," *South African Computer Journal*, vol. 36, pp. 29–48, 2006.

[10] S. S. Khan and M. G. Madden, "A survey of recent trends in one class classification," in *Proceedings of the 20th Irish Conference on Artificial Intelligence and Cognitive Science*, 2009.

[11] C. Bishop, *Neural Networks for Pattern Recognition*, Oxford University Press, Oxford, UK, 1995.

[12] R. O. Duda, P. E. Hart, and D. G. Stork, *Pattern Classification*, John Wiley & Sons, New York, NY, USA, 2000.

[13] A. Ypma, D. M. J. Tax, and R. P. W. Duin, "Robust machine fault detection with independent component analysis and support vector data description," in *Proceedings of the 9th IEEE Workshop on Neural Networks for Signal Processing (NNSP '99)*, Y. H. Hu, J. Larsen, E. Wilson, and S. Douglas, Eds., pp. 67–76, August 1999.

[14] R. O. Duda and P. E. Hart, *Pattern Classification and Scene Analysis*, John Wiley & Sons, New York, NY, USA, 1973.

[15] V. Vapnik, *The Nature of Statistical Learning Theory*, Springer, New York, NY, USA, 1995.

[16] D. M. J. Tax and R. P. W. Duin, "Data domain description using support vectors," in *Proceedings of the 7th European Symposium on Artificial Neural Networks*, pp. 251–256, 1999.

[17] D. M. J. Tax and R. P. W. Duin, "Support Vector data description," *Machine Learning*, vol. 54, no. 1, pp. 45–66, 2004.

[18] I. W. Tsang, J. T. Kwok, and P.-M. Cheung, "Core vector machines: fast SVM training on very large data sets," *Journal of Machine Learning Research*, vol. 6, pp. 363–392, 2005.

[19] D. Wang, D. S. Yeung, and E. C. C. Tsang, "Structured one-class classification," *IEEE Transactions on Systems, Man, and Cybernetics, B: Cybernetics*, vol. 36, no. 6, pp. 1283–1295, 2006.

[20] F. Angiulli, "Prototype-based domain description for one-class classification," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 34, no. 6, pp. 1131–1144, 2012.

[21] E. M. Knorr, R. T. Ng, and V. Tucakov, "Distance-based outliers: algorithms and applications," *VLDB Journal*, vol. 8, no. 3-4, pp. 237–253, 2000.

[22] F. Angiulli, S. Basta, and C. Pizzuti, "Distance-based detection and prediction of outliers," *IEEE Transactions on Knowledge and Data Engineering*, vol. 18, no. 2, pp. 145–160, 2006.

[23] C. M. Bishop, *Neural Networks for Pattern Recognition*, Clarendon Press, 1995.

[24] D. Barber and C. M. Bishop, "Ensemble learning in Bayesian neural networks," in *Neural Networks and Machine Learning*, C. M. Bishop, Ed., vol. 168 of *Series F: Computer and Systems Sciences*, pp. 215–237, Springer, 1998.

[25] B. Lerner and N. D. Lawrence, "A comparison of state-of-the-art classification techniques with application to cytogenetics," *Neural Computing and Applications*, vol. 10, no. 1, pp. 39–47, 2001.

[26] I. Irigoien and C. Arenas, "INCA: new statistic for estimating the number of clusters and identifying atypical units," *Statistics in Medicine*, vol. 27, no. 15, pp. 2948–2973, 2008.

[27] A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, vol. 30, no. 7, pp. 1145–1159, 1997.

[28] R. P. W. Duin, "On the choice of smoothing parameters for Parzen estimators of probabilitydensity functions," *IEEE Transactions on Computers*, vol. 25, no. 11, pp. 1175–1179, 1976.

[29] M. Kraaijveld and R. Duin, "A criterion for the smoothing parameter for parzenestimators of probability density functions," Tech. Rep., Delft University of Technology, 1991.

[30] V. Vapnik, *Statistical Learning Theory*, Wiley Interscience, 1998.

[31] K. R. Müller, S. Mika, G. Rätsch, K. Tsuda, and B. Schölkopf, "An introduction to kernelbased learning algorithms," *IEEE Neural Networks*, vol. 12, pp. 181–201, 2001.

[32] J. C. Gower, "Measures of similarity, dissimilarity and distance," in *Encyclopedia of Statistical Sciences*, S. Kotz, N. L. Johson, and C. B. Read, Eds., pp. 307–316, John Wiley & Sons, New York, NY, USA, 1985.

[33] J. C. Gower, "Some distance properties of latent root and vector methods used in multivariate analysis," *Biometrika*, vol. 53, pp. 325–338, 1966.

[34] W. J. Krzanowski and F. H. C. Marriott, *Multivariate Analysis. Part 1: Distributions, Ordination and Inference*, Kendall's Library of Statistics, Edward Arnold, London, UK, 1994.

[35] C. M. Cuadras and C. Arenas, "A distance based regression-model for prediction with mixed data," *Communications in Statistics A. Theory and Methods*, vol. 19, pp. 2261–2279, 1990.

[36] P. Legendre and M. J. Anderson, "Distancebased redundancy analysis: testing multispecies responses in multifactorial ecological experiments," *Ecological Monographs*, vol. 48, pp. 505–519, 1999.

[37] M. J. Anderson and J. Robinson, "Generalized discriminant analysis based on distances," *Australian and New Zealand Journal of Statistics*, vol. 45, no. 3, pp. 301–318, 2003.

[38] M. J. Anderson and T. J. Willis, "Canonical analysis of principal coordinates: a useful method of constrained ordination for ecology," *Ecology*, vol. 84, no. 2, pp. 511–525, 2003.

[39] W. J. Krzanowski, "Biplots for multifactorial analysis of distance," *Biometrics*, vol. 60, no. 2, pp. 517–524, 2004.

[40] I. Irigoien, C. Arenas, E. Fernández, and F. Mestres, "GEVA: geometric variability-based approaches for identifying patterns in data," *Computational Statistics*, vol. 25, pp. 241–255, 2010.

[41] I. Irigoien, B. Sierra, and C. Arenas, "ICGE: an R package for detecting relevant clusters and atypical units in gene expression," *BMC Bioinformatics*, vol. 13, article 30, 2012.

[42] I. Irigoien, F. Mestres, and C. Arenas, "The depth problem: identifying the most representative units in a data group," *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, vol. 10, pp. 161–172, 2013.

[43] C. R. Rao, "Diversity: its measurement, decomposition, apportionment and analysis," *Sankhyā A*, vol. 44, pp. 1–22, 1982.

[44] C. Arenas and C. M. Cuadras, "Some recent statistical methods based on distances," *Contributions to Science*, vol. 2, pp. 183–191, 2002.

[45] C. Blake, E. Keogh, and C. Merz, "UCI Repository of Machine Learning Database," http://archive.ics.uci.edu/ml/.

[46] P. Juszczak, D. M. J. Tax, E. Pękalska, and R. P. W. Duin, "Minimum spanning tree based one-class classifier," *Neurocomputing*, vol. 72, no. 7–9, pp. 1859–1869, 2009.

[47] J. C. Gower, "A general coefficient of similarity and some of its properties," *Biometrics*, vol. 27, pp. 857–871, 1971.

[48] A. Montanari and S. Mignari, "Notes on the bias of dissimilarity indices for incomplete data sets: the case of archaelogical classifications," *Qüestiió*, vol. 18, pp. 39–49, 1994.

[49] K. Kato, "Adaptor-tagged competitive PCR: a novel method for measuring relative gene expression," *Nucleic Acids Research*, vol. 25, no. 22, pp. 4694–4696, 1997.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 524367, 9 pages
http://dx.doi.org/10.1155/2014/524367

Research Article

Multicompare Tests of the Performance of Different Metaheuristics in EEG Dipole Source Localization

Diana Irazú Escalona-Vargas,[1] Ivan Lopez-Arevalo,[1] and David Gutiérrez[2]

[1] *Information Technology Laboratory, Center for Research and Advanced Studies (Cinvestav), Ciudad Victoria, TAMPS 87130, Mexico*
[2] *Biomedical Signal Processing Laboratory, Center for Research and Advanced Studies (Cinvestav), Apodaca, NL 66600, Mexico*

Correspondence should be addressed to Diana Irazú Escalona-Vargas; descalona@cinvestav.mx

Received 20 December 2013; Accepted 10 February 2014; Published 16 March 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

We study the use of nonparametric multicompare statistical tests on the performance of simulated annealing (SA), genetic algorithm (GA), particle swarm optimization (PSO), and differential evolution (DE), when used for electroencephalographic (EEG) source localization. Such task can be posed as an optimization problem for which the referred metaheuristic methods are well suited. Hence, we evaluate the localization's performance in terms of metaheuristics' operational parameters and for a fixed number of evaluations of the objective function. In this way, we are able to link the efficiency of the metaheuristics with a common measure of computational cost. Our results did not show significant differences in the metaheuristics' performance for the case of single source localization. In case of localizing two correlated sources, we found that PSO (ring and tree topologies) and DE performed the worst, then they should not be considered in large-scale EEG source localization problems. Overall, the multicompare tests allowed to demonstrate the little effect that the selection of a particular metaheuristic and the variations in their operational parameters have in this optimization problem.

1. Introduction

The problem of source localization is of great interest in neuroscience. It has applications in areas such as clinical sciences and brain research [1]. Techniques based on electroencephalography (EEG) measure the electric potentials on the scalp and process them to infer the location of the underlying neural activity. Such inference is mainly based in the solution of two problems: first, the *forward* problem of computing the electric potentials over the scalp given a current source within the brain, which may be solved by selecting a proper model to approximate the volume conductor, in addition to the dipole model already assumed for the source signals; second, the *inverse* problem of finding the current distributions using EEG measurements, which often involves an iterative solution of the forward problem until an optimization criterion is attained. Therefore, it is important to have efficient optimization algorithms in order to solve the inverse problem based on this modeling-optimization approach.

The forward problem delivers an objective function which usually is very complex and has many local optima, especially when the number of dipole sources is large, or when dealing with low signal-to-noise ratio (SNR) conditions. Hence, metaheuristic techniques are promising candidates to help solving this problem as they are designed to escape from local optima and proceed with the exploration of the search space until finding the global optima in modest computation times. The most representative algorithms are simulated annealing (SA), genetic algorithms (GA), particle swarm optimization (PSO), differential evolution (DE), and tabu search (TS). TS and SA (referred to as trajectory methods) work on one or several neighborhood structures imposed on the solutions of the search space. Evolutionary techniques, such as GA, PSO, and DE, incorporate a learning component in the sense that they implicitly or explicitly learn correlations between decision variables to identify high quality areas in the search space. Evolutionary metaheuristics perform a biased sampling of the search space by recombination of solutions.

Comparative studies of metaheuristics for solving the inverse problem have been performed in the past, and they are provided with valuable information regarding the performance of the methods in the localization of one or multiple dipoles. Some examples of such studies are given next: In [2], Lewis and Mosher proposed GA as a promising approach to find minimal source solutions using distributed dipoles modeling in magnetoencephalography (MEG) signals. In [3], Haneishi et al. considered SA in the localization of multiple dipoles using noiseless MEG data, and they found that SA is effective in solving the inverse problem, but it had a high computational performance. Moreover, they detailed the implementation of the SA algorithm in [4], where a modification of the method for its implementation in parallel computers was proposed. In [5], Gerson et al. performed a comparative study between the SA and simplex method using EEG data under noiseless conditions, in which they determined the sensitivity of the methods through simulations under different initial assumption of the dipole's position. They concluded that the simplex algorithm was affected with the different initial solutions whereas the SA performance was not affected. Furthermore, the simplex method delivered large errors even when they proposed solutions near to the global optimum, and forfeited its convergence speed advantage compared with SA. In [6], McNay et al. used GA for the estimation of two dipoles using EEG signals under specific noise conditions. They also studied the localization accuracy of the algorithm using a physical model incorporating potential measurements of two simultaneously active sources embedded in a sphere. In [7], Khosla et al. compared SA and simplex method in localizing three dipoles in EEG data under noise conditions. They used different SA's parameter settings from Gerson et al. [5], and also they concluded that SA is a feasible method even if it does not provide a good solution from the beginning. In [8], Uutela et al. compared a clustering method, GA, and SA for localizing two dipoles using MEG signals under noiseless conditions. GA was the most effective method whereas the clustering method performed well when the number of sources was small. In [9], Nagano et al. used GA with MEG recordings when localizing two dipoles under various noise conditions. The authors concluded that GA was a robust method for high SNR conditions. In [10], Scherg et al. reconstructed multiple sources using EEG/MEG data simultaneously acquired and through GA with sequential dipole fitting strategy. In [11], Jiang et al. compared GA, SA, and TS methods for localizing three dipoles using MEG data. They found that the three algorithms converged to the global optimum if the computational resources are unlimited, but the best results were achieved with a hybrid GA under noise conditions. In [12], Zou et al. performed a comparative study between a hybrid GA and simplex using EEG data. They found that the hybrid GA gives better solutions under specific noise conditions. In [13], Li et al. implemented a DE algorithm in the localization of multiple dipoles under noiseless conditions. They concluded that DE is a feasible metaheuristic in the reconstruction of EEG source localization with single dipole sources but not when there are two or more dipoles active at the same time. In [14], Sequeira et al. performed GA and its hybrid version using MEG data in the localization of deep and cortical sources. There, the authors used two-objective functions to make the algorithm more precise and efficient in terms of computation times. They found that GA was a robust method to determinate the positions of multidipole sources simultaneously. In [15], Qiu et al. compared PSO and GA for high SNR conditions. They found that the PSO algorithm was more accurate and required less computational effort than GA in EEG source localization. In [16], Jiang et al. used GA to perform a rough source location and then applied the music (multiple signal classification) method to refine the search until obtaining an accurate position of MEG sources. They concluded that the GA-music strategy improved the speed and accuracy in source localization under noiseless conditions. Furthermore, they obtained same results when the PSO algorithm was used instead of GA [17], but the PSO-music approach was a better strategy than GA-music. In [18], Alp et al. used PSO algorithm in the localization of the sources of event related potentials (ERP). The authors concluded that PSO was accurate even when the ERP sources generated signals that overlap in time and frequency. In [19], Parsopoulos et al. compared PSO and the unified PSO (UPSO) in MEG source localization. They found that UPSO exhibited better efficiency and robustness in the case of MEG noiseless data, and it seemed to be less affected by relatively small increases in the number of sensors than PSO. Regardless of noise conditions, the efficiency of both algorithms was similar. In [20], Rytsar and Pun compared SA and GA using EEG data. The best results were achieved with GA, but its computational cost was higher compared with SA algorithm. In [21], Shirvany et al. proposed a new global optimization method based on PSO to solve EEG source localization under noiseless conditions. They concluded that the new strategy of PSO found the optimal solution faster than other PSO methods from the literature, and it was less prone to be trapped in local minima.

Metaheuristics are often compared in terms of CPU times, solution quality, parallel speedups, and function evaluation counts whereas most of the studies in solving the EEG/MEG inverse problem evaluate the performance of the metaheuristics in terms of the mean squared error (MSE) under very specific SNR conditions. In our previous work (see [22]), we performed a comparative study of SA, GA, PSO, and DE under realistic conditions and for different values of their operational parameters. We used the concentrated likelihood function (CLF) as objective function and the Cramér-Rao bound (CRB) as a generalized lower-bound to compare the efficiency of the metaheuristics, rather than concentrating only on the MSE. In the case of SA, GA, and PSO methods, we used the operational parameters that have proved to work well in the EEG inverse problem. However, for the case of DE we performed an exhaustive simulation to find a combination of parameters for which an optimal solution was attained in the localization of two correlated dipoles. Our results showed that the performance of SA, GA, PSO, and DE decayed as the noise increased, while SA and PSO seemed to be very sensitive to the correlation between the sources. Overall, GA was the most attractive technique in terms of performance using the CRB as a reference. However, a formal statistical analysis was not performed in order to show if

the differences in the performance between metaheuristics were indeed statistically significant. Hence, in this paper we propose a multicompare analysis of variance (ANOVA) in order to evaluate such differences.

This paper is organized as follows: in Section 2, we pose the source localization problem as the optimization of the concentrated likelihood function (CLF); in Section 3, we describe the multicompare tests by which we evaluate the performance of the metaheuristics; in Section 4, we present the experimental settings when using simulated EEG data; in Section 5, the results of our numerical examples are shown; in Section 6, we give our concluding remarks and future work.

2. The Optimization Problem

The inverse problem may be solved by modeling the source signal and the volume conductor in the following way [23].

(i) A model is proposed for the source signal. In our case, we will use current dipoles, which are widely used to approximate the brain activity in evoked response and event related experiments [24].

(ii) A model is constructed for the volume conductor. The accuracy of the conductor model must be as good as or better than that of the source model. Here, we assume the classical concentric four sphere model to approximate the head geometry. This model is justified for sources near the surface [25].

(iii) At least as many independent EEG measurements are made as the model has independent variables. Under those conditions, the mathematical representation of the problem would have as many equations as unknowns, and the variables of the model could be evaluated.

Based on this idea, it is possible to iteratively solve the inverse problem by assuming known values for the variables of the model and then solving the forward problem for those assumed parameters and finally comparing the computed EEG against the measured data. Then, the process of solving the inverse problem becomes an *optimization* problem if we define an objective function and, at each iteration, the assumed parameters are adjusted and the objective function is reevaluated until an optimality criterion is attained. Next, we go into further detail about defining the concentrated likelihood function, which will be the objective function to be used throughout this paper.

Let us consider that a source of brain activity is modeled as a single dipole with a moment $\mathbf{q} \in \mathbb{R}^3$, which is located at position $\mathbf{r}_q \in \mathbb{R}^3$ within the brain. For the mth-sensor located on the scalp at $\mathbf{r}_m \in \mathbb{R}^3$, $m = 1, \ldots, M$, the surface potential can be expressed as $v_m = g_m^T(\boldsymbol{\theta})\mathbf{q}$, where $g_m^T(\boldsymbol{\theta})$ is the gain vector (or *kernel* vector) which is a function of the vector of parameters $\boldsymbol{\theta}$. Under these conditions, we can define a potential vector as

$$\mathbf{v} = \left[v_1, v_2, \ldots, v_M\right]^T = A(\boldsymbol{\theta})\,\mathbf{q}, \tag{1}$$

where $A(\boldsymbol{\theta})$ is the $M \times 3$ *lead-field* matrix, in which the mth-row corresponds to $g_m^T(\boldsymbol{\theta})$. $A(\boldsymbol{\theta})$ is derived from using the quasistatic approximation of Maxwell's equations on a volume that approximates the head's geometry. In a physical sense, $A(\boldsymbol{\theta})$ represents the material and geometrical properties of the medium in which the sources are submerged. Furthermore, the model in (1) can be extended to a spatiotemporal representation by allowing change in time. Hence, if we assume that the source remains at the same position during the measurement period, we obtain the following:

$$\mathbf{v}(t) = A(\boldsymbol{\theta})\,\mathbf{q}(t), \tag{2}$$

for $t = 1, 2, \ldots, N$ time samples. Finally, in the case of p distinct dipoles, (2) holds with $\mathbf{q}(t)$ and $A(\boldsymbol{\theta})$ substituted with $\mathbf{q}(t) = [\mathbf{q}_1(t), \mathbf{q}_2(t), \ldots, \mathbf{q}_p(t)]^T$, and $A(\boldsymbol{\theta}) = [A_1(\boldsymbol{\theta}), A_2(\boldsymbol{\theta}), \ldots, A_p(\boldsymbol{\theta})]$, respectively.

Equation (2) can be used to represent the forward model as a linear measurement model in the presence of additive noise as follows:

$$Y_k(t) = \mathbf{v}(t) + E_k, \tag{3}$$

where $Y_k(t)$ is the matrix of measurements obtained from $k = 1, 2, \ldots, K$ independent experiments, and E_k is the matrix of noise. Under these conditions, our goal is to determine $\boldsymbol{\theta} = \mathbf{r}_q$ from (3) that best describes the EEG measurements. Here, we consider the maximum likelihood (ML) technique to estimate the position parameters, as it has been shown that an unbiased estimate of $\boldsymbol{\theta}$ (denoted as $\widehat{\boldsymbol{\theta}}$) can be obtained through the following optimization problem [26]:

$$\widehat{\boldsymbol{\theta}} = \min_{\boldsymbol{\theta}} F(\boldsymbol{\theta}), \tag{4}$$

where $F(\boldsymbol{\theta})$ is the concentrated likelihood function (CLF), defined as

$$F(\boldsymbol{\theta}) = \operatorname{tr}\left\{\left(I - A\left(A^T A\right)^{-1} A\right) R\right\}, \tag{5}$$

where $\operatorname{tr}\{\cdot\}$ is the trace operator, I is the identity matrix, $A = A(\boldsymbol{\theta})$ is used for simplicity in the notation, and R is the data's covariance matrix. When unknown, a consistent estimate of R is usually obtained as

$$\widehat{R} = \frac{1}{N}\sum_{t=1}^{N} Y(t)\,Y(t)^T. \tag{6}$$

Therefore, the CLF corresponds to the objective function in the optimization problem from which the dipole parameters will be estimated. In our case, $\boldsymbol{\theta}$ can be estimated by minimizing (5) through the proposed metaheuristics. For that matter, let us consider that Ω_{brain} is the brain's domain where the dipole's location is determined. Then, we pose the following optimization problem:

$$\widehat{\boldsymbol{\theta}} = \min_{\substack{\boldsymbol{\theta} \in \Omega_{\text{brain}} \\ b \le \boldsymbol{\theta} \le a}} F(\boldsymbol{\theta}), \tag{7}$$

where a and b are constant constraints.

Given that the CLF is derived from ML principles, the resulting estimates have a handful of desired properties: they are consistent, asymptotically Gaussian distributed, and asymptotically efficient [27]. Thus, the optimization is expected to converge to the true value for a sufficiently large number of data samples (i.e., $KN \gg M$). In such a case, the bias in the estimation disappears asymptotically and the variance approaches zero. Moreover, no other bias-free estimator exists with a smaller variance. Then, now the question is how to determine the best suited metaheuristic to solve the problem in (7), for which we propose the multicomparison tests described next.

3. Multicompare Tests

Statistical analysis is a powerful tool to evaluate the performance of algorithms, as well as to quantify the relationship between algorithm performance and other factors describing problem characteristics. We find in the literature many methods for such purposes: the analysis of variance (ANOVA), t-test, F-test, and least-squares regression, as well as robust alternatives such as Friedman's test and L1-regression. Pairwise comparisons are the simplest type of statistical tests used to compare the performance of two algorithms in a common set of problems. In multiproblem analysis, a value for each pair of algorithm/problem is required. However, when there are more than two groups, a multicompare approach is needed.

Multiple comparisons of various algorithms must be carried out by first using a statistical method for testing the differences among the related samples means. Once this test rejects the hypothesis of equivalence of means, the detection of the concrete differences among the algorithms can be done with the application of a *post hoc* statistical process. Parametric tests have been commonly used in the analysis of experiments in computational intelligence. Unfortunately, they are based on independence, normality, and heteroscedasticity assumptions, which are most probably not attained when analyzing the performance of stochastic algorithms based on computational intelligence [28]. Nonparametric tests are used to overcome this problem as they do not need prior assumptions related to the sample of data to be analyzed. Furthermore, nonparametric tests have been already used for comparing metaheuristic algorithms in several benchmark functions [29].

In our case, we use the Kruskal-Wallis test, which compares the medians between two or more samples in order to determine if they originate from the same distribution [30]. The Kruskal-Wallis test is used to compare groups when the distribution does not prove prove to be normal or when their variances are different (this latest condition applies to the problem of EEG source localization, as demonstrated in [22]). In our case, the null hypothesis is considered as H_0: *Each metaheuristic has the same median performance*. Then, when the Kruskal-Wallis test leads to significant results, it would indicate that at least one median performance is different from another.

Under these conditions, the analysis of the performance is conducted as follows.

(1) Metaheuristics under different operational parameters are used in the estimation of dipole's position by solving (7) for a fixed number of evaluations of (5);

(2) the optimization process is repeated 100 times under independent noise conditions;

(3) at each trial, the MSE is computed as

$$\text{MSE} = \sqrt{\left(\theta_x - \widehat{\theta}_x\right)^2 + \left(\theta_y - \widehat{\theta}_y\right)^2 + \left(\theta_z - \widehat{\theta}_z\right)^2}, \quad (8)$$

where $\boldsymbol{\theta} = [\theta_x, \theta_y, \theta_z]^T$ and $\widehat{\boldsymbol{\theta}} = [\widehat{\theta}_x, \widehat{\theta}_y, \widehat{\theta}_z]^T$ are the true and estimated Cartesian coordinates of the dipole's position, respectively;

(4) The MSE values are used to perform the Kruskal-Wallis test and, if it reveals that at least one median performance is different, then the Dunn-Sidak *post hoc* test is performed to identify pairs of metaheuristics with significant different performances. The Dunn-Sidak test determines the critical values to reject the null-hypothesis as follows:

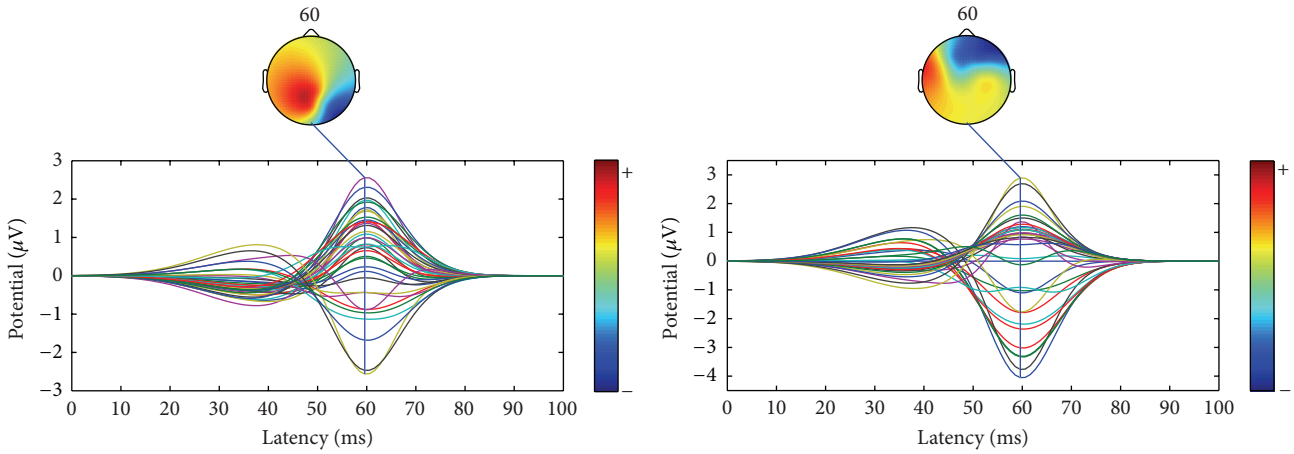
$$\alpha_i = 1 - \left(1 - \alpha_e\right)^{1/n}, \quad (9)$$

for $i = 1, 2, \ldots, n$ groups being compared, α_i is the significance level of each individual test, and α_e is the family-wise or experiment-wise significance level [31].

4. Experimental Settings

We generated EEG data which simulated a typical evoked responses [32] (see Figure 1) for one and two correlated dipole sources using the classical spherical head model with an array of $M = 37$ electrodes. The multishell spherical head model includes four concentric layers for the brain, cerebrospinal fluid (CSF), skull, and scalp. The radii for each of the layers were, respectively, 92.45, 89.29, 85.10, and 83 mm. These layers were considered to be isotropic and to have homogeneous conductivities of 0.33, 0.0042, 1, and 0.33 S/m, respectively. Those values of radii and conductivities were chosen in accordance to the model proposed in [25]. Next, we added uncorrelated random noise to obtain SNR = 0 dB, and SA, GA, PSO, and DE methods were used in the estimation of dipole's position using the parameter settings defined in [22]. The stopping criterion of all metaheuristics was set up for a number of function evaluations of 1000 and 2500 when estimating one and two correlated dipoles, respectively. In all experiments, a PC Intel Xeon E3 quad-core at 3.10 GHz with 8 GB in RAM was used. Then, at each trial we calculated the MSE defined in (8), and those values were used to perform the multicompare test.

In order to simplify the optimization process, we translated the dipole's position to the spherical coordinates ϑ, φ, and ϱ, which correspond to the azimuth angle, elevation, and eccentricity, respectively. Then, the optimization problem was solved as a function of ϑ and φ only, while the eccentricity was kept to a fixed value of $\varrho = 83$ mm. Therefore, a and b in (7)



(a) Single dipole (b) Two dipoles

FIGURE 1: Simulated EEG data under noiseless conditions in a spherical head model.

corresponded to the lower- and upper-bound constraints of ϑ and φ. In our experiments, the single dipole was located at $\vartheta = 0.5235$ rad and $\varphi = -1.2$ rad (see Figure 1(a)). For the case of two correlated dipoles, they were located at $\vartheta_1 = \vartheta_2 = 0.5235$ rad, while their azimuth angles were $\varphi_1 = -0.6$ rad and $\varphi_2 = 0.6$ rad, respectively, (see Figure 1(b)).

5. Results

In this section we present the results of our multicompare tests of the performance.

5.1. Single Dipole. The problem of estimating the location of a single dipole in our settings corresponded to find $\widehat{\boldsymbol{\theta}} = [\widehat{\vartheta}, \widehat{\varphi}]$. Since thirteen different operational parameters were varied among the four metaheuristics evaluated (see [22] for further detail), here we used box plots to show the results of the evaluation of the performance. There, the MSE is given in millimeters. Hence, in Figure 2 we can observe that the median MSE of the generic GA is slightly different in comparison to the other metaheuristics, but the statistical test did not reveal significant differences in the performance at a significance level of 0.01.

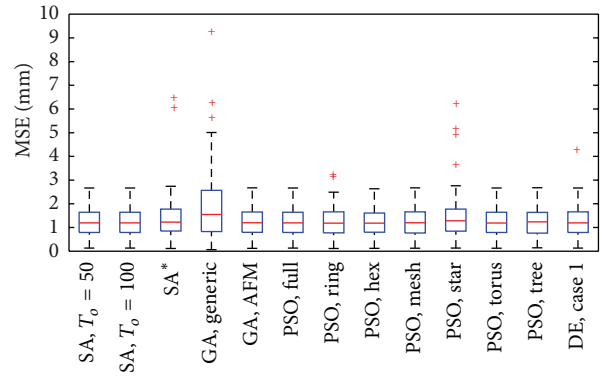
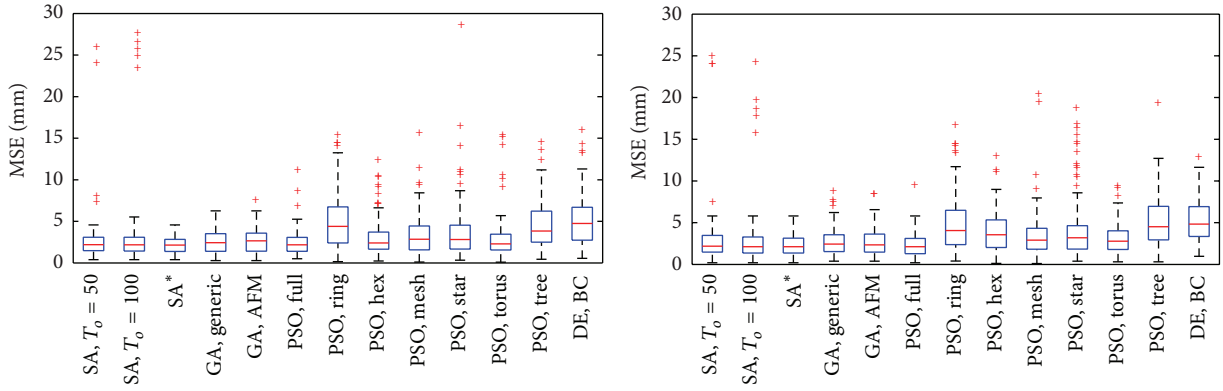


FIGURE 2: MSE of $\widehat{\boldsymbol{\theta}}$ in one-dipole localization using SA, GA, PSO, and DE with different initialization parameters, 1000 evaluations of the objective function and SNR = 0 dB. SA* correspond to the case of SA when $T_o = \Delta\overline{E}/\ln(\beta^{-1})$. AFM refers to the case when adaptive feasible mutation was used for GA. For the PSO algorithm, $c_1 = c_2 = 2.83$ was used in all cases. DE Case 1 corresponds to the experiment using the generic strategy (DE/rand/1/bin) with $\Gamma = \zeta = 0.9$. The central mark in each method corresponds to its median, the edges of the box are the 25th and 75th percentiles, the whiskers extensions are the most extreme data points, and "+" markers correspond to the outliers.

5.2. Two Correlated Dipoles. In this case, the optimization process corresponded to find $\widehat{\boldsymbol{\theta}} = [\widehat{\vartheta}_1, \widehat{\varphi}_1, \widehat{\vartheta}_2, \widehat{\varphi}_2]$. Figure 3 shows the results of the evaluation of the performance. There, we can observe that the algorithms achieved different results. Therefore, a summary of the results of the Dunn-Sidak tests are shown in Table 1, where "√" and "×" indicate if the corresponding metaheuristics had indeed different performances or not, respectively. Note that the family-wise and individual significance levels were, respectively, $\alpha_e = 0.01$ and $\alpha_i = 1 - (1 - 0.01)^{1/n}$, where $n = \binom{13}{2} = 78$. The values that are shown below the metaheuristic's name (denoted as $\widetilde{\boldsymbol{\theta}}_1$) correspond to the median of the MSE evaluated over the $K = 100$ independent trials. Note that we only show the multicompare results corresponding to one of the two dipoles' location as the errors for the other dipole that are estimated with the same metaheuristic were very similar. We can observe from Table 1 that the most viable method was the SA with $T_o = \Delta\overline{E}/\ln(\beta^{-1})$ as it obtained a minimum median error (MSE = 2.15) but, as the analysis in [22] demonstrated, this method had bad performance under low SNR conditions for the case of two correlated dipoles. Therefore, the analysis that is proposed here and the one in [22] are complementary and should be performed together in order to fully evaluate different optimization methods.

Table 1: Multicomparative table using Kruskal-Wallis and Dunn-Sidak post hoc test in two-dipole localization using SA, GA, PSO, and DE with different initialization parameters, 2500 evaluations of the objective function, and SNR = 0 dB. SA* corresponds to the case of SA when $T_o = \Delta\overline{E}/\ln(\beta^{-1})$. AFM refers to the case when adaptive feasible mutation was used for GA. For the PSO algorithm, $c_1 = c_2 = 2.83$ was used in all cases. DE BC corresponds to the experiment using the generic strategy (DE/rand/1/bin) with $\Gamma = 0.5$ and $\zeta = 0.2$. $\widetilde{\boldsymbol{\theta}}_1$ values correspond to the median of the MSE evaluated over the $K = 100$ independent trials.

	SA, $T_o = 100$ $\widetilde{\boldsymbol{\theta}}_1 = 2.16$	SA* $\widetilde{\boldsymbol{\theta}}_1 = 2.15$	GA, generic $\widetilde{\boldsymbol{\theta}}_1 = 2.43$	GA, AFM $\widetilde{\boldsymbol{\theta}}_1 = 2.65$	PSO, full $\widetilde{\boldsymbol{\theta}}_1 = 2.18$	PSO, ring $\widetilde{\boldsymbol{\theta}}_1 = 4.39$	PSO, hex $\widetilde{\boldsymbol{\theta}}_1 = 2.39$	PSO, mesh $\widetilde{\boldsymbol{\theta}}_1 = 2.85$	PSO, star $\widetilde{\boldsymbol{\theta}}_1 = 2.81$	PSO, torus $\widetilde{\boldsymbol{\theta}}_1 = 2.28$	PSO, tree $\widetilde{\boldsymbol{\theta}}_1 = 3.83$	DE, BC $\widetilde{\boldsymbol{\theta}}_1 = 4.74$
SA, $T_o = 50$ $\widetilde{\boldsymbol{\theta}}_1 = 2.19$	×	×	×	×	×	√	×	×	×	×	√	√
SA, $T_o = 100$ $\widetilde{\boldsymbol{\theta}}_1 = 2.16$		×	×	×	×	√	×	×	×	×	√	√
SA* $\widetilde{\boldsymbol{\theta}}_1 = 2.15$			×	×	×	√	×	×	×	×	√	√
GA, generic $\widetilde{\boldsymbol{\theta}}_1 = 2.43$				×	×	√	×	×	×	×	√	√
GA, AFM $\widetilde{\boldsymbol{\theta}}_1 = 2.65$					×	√	×	×	×	×	√	√
PSO, full $\widetilde{\boldsymbol{\theta}}_1 = 2.18$						√	×	×	×	×	√	√
PSO, ring $\widetilde{\boldsymbol{\theta}}_1 = 4.39$							√	×	×	√	×	×
PSO, hex $\widetilde{\boldsymbol{\theta}}_1 = 2.39$								×	×	×	√	√
PSO, mesh $\widetilde{\boldsymbol{\theta}}_1 = 2.85$									×	×	×	√
PSO, star $\widetilde{\boldsymbol{\theta}}_1 = 2.81$										×	×	√
PSO, torus $\widetilde{\boldsymbol{\theta}}_1 = 2.28$											√	√
PSO, tree $\widetilde{\boldsymbol{\theta}}_1 = 3.83$												×



(a) Estimation of the location parameters of dipole 1

(b) Estimation of the location parameters of dipole 2

FIGURE 3: MSE of $\widehat{\boldsymbol{\theta}}$ in two-dipole localization using SA, GA, PSO, and DE with different initialization parameters, 2500 evaluations of the objective function and SNR = 0 dB. SA* correspond to the case of SA when $T_o = \Delta\overline{E}/\ln(\beta^{-1})$. AFM refers to the case when adaptive feasible mutation was used for GA. For the PSO algorithm, $c_1 = c_2 = 2.83$ was used in all cases. DE BC corresponds to the experiment using the generic strategy (DE/rand/1/bin) with $\Gamma = 0.5$ and $\zeta = 0.2$. The central mark in each method corresponds to its median, the edges of the box are the 25th and 75th percentiles, the whiskers extensions are the most extreme data points, and "+" markers correspond to the outliers.

6. Conclusions

In this paper we used nonparametric multicompare tests to evaluate the performance of different metaheuristics in solving an optimization problem for EEG dipole source localization. We evaluated the performance in terms of metaheuristics' operational parameters and for a fixed number of evaluations of the objective function. Through this process, we were able to link the metaheuristics' efficiencies with a common measure of computational cost. The results showed that there were no significant differences between the SA, GA, PSO, and DE in one-dipole estimation. For two correlated dipoles, the SA algorithm with $T_o = \Delta\overline{E}/\ln(\beta^{-1})$ was the most viable method, while the worst performing methods were PSO (ring and tree topologies) and DE. However, the performance of SA is very sensitive to other factors, such as the SNR conditions. Hence, the statistical analysis that is proposed here provides valuable information in terms of the bias of the estimation among different metaheuristics, but a complementary analysis is necessary in order to also evaluate the different methods' robustness.

Our proposed analysis can be easily extended to evaluate other factors, for example, the electrical properties of the surrounding tissues. It is well know that errors are introduced in the solution of the EEG inverse problem due to dissimilarities in the values of conductivities of the tissues among individuals [33–35], then it might be valuable to add those parameters in the optimization framework and evaluate the performance of metaheuristics using both the nonparametric statistical tests that are presented here and the stochastic Cramér-Rao bounds [36] in order to account also for noise effects.

Since adding the value of the conductivities as parameters in the optimization implies a more complex optimization problem, our future work will be dedicated to evaluate advanced versions of current optimization algorithms, such as steady state genetic algorithm (SSGA) [37], adaptive differential evolution (JDE) [38], parameter adaptive differential evolution (JADE) [39], and self-adaptive differential evolution (SADE) [40]. It is also important to explore new strategies such as the backtracking search optimization algorithm (BSA) [41] or the covariance matrix adaptation evolution strategy (G-CMA-ES) [42].

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] E. Niedermeyer and F. H. Lopes da Silva, *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*, Wolters Kluwer Health, 2005.

[2] P. S. Lewis and J. C. Mosher, "Genetic algorithms for minimal source reconstructions," in *Proceedings of the 27th Asilomar Conference on Signals, Systems & Computers*, pp. 335–337, November 1993.

[3] H. Haneishi, N. Ohyama, K. Sekihara, and T. Honda, "Multiple current dipole estimation using simulated annealing," *IEEE Transactions on Biomedical Engineering*, vol. 41, no. 11, pp. 1004–1009, 1994.

[4] K. Sekihara, H. Haneishi, and N. Ohyama, "Details of simulated annealing algorithm to estimate parameters of multiple current dipoles using biomagnetic data," *IEEE Transactions on Medical Imaging*, vol. 11, no. 2, pp. 293–299, 1992.

[5] J. Gerson, V. A. Cardenas, and G. Fein, "Equivalent dipole parameter estimation using simulated annealing," *Electroencephalography and Clinical Neurophysiology*, vol. 92, no. 2, pp. 161–168, 1994.

[6] D. McNay, E. Michielssen, R. L. Rogers, S. A. Taylor, M. Akhtari, and W. W. Sutherling, "Multiple source localization

using genetic algorithms," *Journal of Neuroscience Methods*, vol. 64, no. 2, pp. 163–172, 1996.

[7] D. Khosla, M. Singh, and M. Don, "Spatio-temporal EEG source localization using simulated annealing," *IEEE Transactions on Biomedical Engineering*, vol. 44, no. 11, pp. 1075–1091, 1997.

[8] K. Uutela, M. Hämäläinen, and R. Salmelin, "Global optimization in the localization of neuromagnetic sources," *IEEE Transactions on Biomedical Engineering*, vol. 45, no. 6, pp. 716–723, 1998.

[9] T. Nagano, Y. Ohno, N. Uesugi, H. Ikeda, A. Ishiyama, and N. Kasai, "Multi-source localization by genetic algorithm using MEG," *IEEE Transactions on Magnetics*, vol. 34, no. 5, pp. 2972–2975, 1998.

[10] M. Scherg, T. Bast, and P. Berg, "Multiple source analysis of interictal spikes: goals, requirements, and clinical value," *Journal of Clinical Neurophysiology*, vol. 16, no. 3, pp. 214–224, 1999.

[11] T. Jiang, A. Luo, X. Li, and F. Kruggel, "A comparative study of global optimization approaches to MEG source localization," *International Journal of Computer Mathematics*, vol. 80, no. 3, pp. 305–324, 2003.

[12] L. Zou, S. Zhu, and B. He, "Spatio-temporal EEG dipole estimation by means of a hybrid genetic algorithm," in *Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '04)*, pp. 4436–4439, September 2004.

[13] Y. Li, H. Li, R. He et al., "EEG source localization using differential evolution method," in *Proceedings of the 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC '04)*, pp. 1903–1906, September 2004.

[14] C. Sequeira, F. Sanchez-Quesada, M. Sancho, I. Hidalgo, and T. Ortiz, "A genetic algorithm approach for localization of deep sources in MEG," in *Proceedings of the 1st International Meeting on Applied Physics (APHYS '03)*, pp. 140–142, October 2003.

[15] L. Qiu, Y. Li, and D. Yao, "A feasibility study of EEG dipole source localization using particle swarm optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, pp. 720–726, Edinburgh, UK, September 2005.

[16] C. Jiang, J. Ma, B. Wang, and L. Zhang, "Multiple signal classification based on genetic algorithm for MEG sources localization," in *Advances in Neural Networks—ISNN*, D. Liu, S. Fei, Z. Hou, H. Zhang, and C. Sun, Eds., vol. 4492, pp. 1133–1139, Springer, Berlin, Germany, 2007.

[17] C. Jiang, B. Wang, and L. Zhang, "Particle swarm optimization for MEG source localization," in *Proceedings of the 3rd International Conference on Pattern Recognition in Bioinformatics*, pp. 73–82, 2008.

[18] Y. K. Alp, O. Arikan, and S. Karakas, "ERP source reconstruction by using Particle Swarm Optimization," in *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '09)*, pp. 365–368, Taipei City, Taiwan, April 2009.

[19] K. E. Parsopoulos, F. Kariotou, G. Dassios, and M. N. Vrahatis, "Tackling magnetoencephalography with particle swarm optimization," *International Journal of Bio-Inspired Computation*, vol. 1, no. 1-2, pp. 32–49, 2009.

[20] R. Rytsar and T. Pun, "EEG source reconstruction using global optimization approaches: genetic algorithms versus simulated annealing," *International Journal of Tomography and Statistics*, vol. 14, no. 10, pp. 83–94, 2010.

[21] Y. Shirvany, F. Edelvik, S. Jakobsson, A. Hedström, and M. Persson, "Application of particle swarm optimization in epileptic spike EEG source localization," *Applied Soft Computing*, vol. 13, no. 5, pp. 2515–2525, 2013.

[22] D. I. Escalona-Vargas, D. Gutiérrez, and I. Lopez-Arevalo, "Performance of different metaheuristics in EEG source localization compared to the Cramér-Rao bound," *Neuro-Computing*, vol. 120, pp. 597–609, 2013.

[23] J. Malmivuo and R. Plonsey, *Bioelectromagnetism: Principles and Applications of Bioelectric and Biomagnetic Fields*, Oxford University Press, New York, NY, USA, 1995.

[24] J. W. Rohrbaugh, R. Parasuraman, and R. Johnson, *Event-Related Brain Potentials: Basic Issues and Applications*, Oxford University Press, New York, NY, USA, 1990.

[25] B. N. Cuffin and D. Cohen, "Comparison of the magnetoencephalogram and electroencephalogram," *Electroencephalography and Clinical Neurophysiology*, vol. 47, no. 2, pp. 132–146, 1979.

[26] P. Stoica and A. Nehorai, "On the concentrated stochastic likelihood function in array signal processing," *Circuits, Systems, and Signal Processing*, vol. 14, no. 5, pp. 669–674, 1995.

[27] P. Stoica and A. Nehorai, "MUSIC, maximum likelihood, and Cramer-Rao bound," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 720–741, 1989.

[28] J. Higgins, *Introduction to Modern Nonparametric Statistics*, Duxbury Press, 2003.

[29] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18, 2011.

[30] J. H. Zar, *Biostatistical Analysis*, Prentice-Hall, Englewood Cliffs, NJ, USA, 2009.

[31] Y. Hochberg and A. C. Tamhane, *Multiple Comparison Procedures*, John Wiley & Sons, New York, NY, USA, 1987.

[32] A. Dogandzic and A. Nehorai, "Estimating evoked dipole responses in unknown spatially correlated noise with EEG/MEG arrays," *IEEE Transactions on Signal Processing*, vol. 48, no. 1, pp. 13–25, 2000.

[33] D. Gutiérrez, A. Nehorai, and C. H. Muravchik, "Estimating brain conductivities and dipole source signals with EEG arrays," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 12, pp. 2113–2122, 2004.

[34] G. Şengül and U. Baysal, "An extended Kalman filtering approach for the estimation of human head tissue conductivities by using EEG data: a simulation study," *Physiological Measurement*, vol. 33, no. 4, pp. 571–586, 2012.

[35] S. Vallaghé and M. Clerc, "A global sensitivity analysis of three- and four-layer EEG conductivity models," *IEEE Transactions on Biomedical Engineering*, vol. 56, no. 4, pp. 988–995, 2009.

[36] B. M. Radich and K. M. Buckley, "EEG dipole localization bounds and MAP algorithms for head models with parameter uncertainties," *IEEE Transactions on Biomedical Engineering*, vol. 42, no. 3, pp. 233–241, 1995.

[37] C. Fernandes and A. Rosa, "A study on non-random mating and varying population size in genetic algorithms using a Royal Road function," in *Proceedings of the Congress on Evolutionary Computation*, pp. 60–66, May 2001.

[38] J. Brest, S. Greiner, B. Bošković, M. Mernik, and V. Zumer, "Self-adapting control parameters in differential evolution: a comparative study on numerical benchmark problems," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 6, pp. 646–657, 2006.

[39] J. Zhang and A. C. Sanderson, "JADE: adaptive differential evolution with optional external archive," *IEEE Transactions on Evolutionary Computation*, vol. 13, no. 5, pp. 945–958, 2009.

[40] A. K. Qin and P. N. Suganthan, "Self-adaptive differential evolution algorithm for numerical optimization," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, pp. 1785–1791, September 2005.

[41] P. Civicioglu, "Backtracking search optimization algorithm for numerical optimization problems," *Applied Mathematics and Computation*, vol. 219, no. 15, pp. 8121–8144, 2013.

[42] A. Auger and N. Hansen, "A restart CMA evolution strategy with increasing population size," in *Proceedings of the IEEE Congress on Evolutionary Computation (CEC '05)*, pp. 1769–1776, September 2005.

Hindawi Publishing Corporation
The Scientific World Journal
Volume 2014, Article ID 138548, 7 pages
http://dx.doi.org/10.1155/2014/138548

Research Article

Adaptive Iterated Extended Kalman Filter and Its Application to Autonomous Integrated Navigation for Indoor Robot

Yuan Xu,[1,2] Xiyuan Chen,[1,2] and Qinghua Li[1,3]

[1] *School of Instrument Science and Engineering, Southeast University, Nanjing, China*
[2] *Key Laboratory of Micro-Inertial Instrument and Advanced Navigation Technology, Ministry of Education, Nanjing, China*
[3] *School of Electrical Engineering and Automation, Qilu University of Technology, Jinan, China*

Correspondence should be addressed to Xiyuan Chen; chxiyuan@seu.edu.cn

Received 24 October 2013; Accepted 30 December 2013; Published 13 February 2014

Academic Editors: S. Balochian, V. Bhatnagar, and Y. Zhang

As the core of the integrated navigation system, the data fusion algorithm should be designed seriously. In order to improve the accuracy of data fusion, this work proposed an adaptive iterated extended Kalman (AIEKF) which used the noise statistics estimator in the iterated extended Kalman (IEKF), and then AIEKF is used to deal with the nonlinear problem in the inertial navigation systems (INS)/wireless sensors networks (WSNs)-integrated navigation system. Practical test has been done to evaluate the performance of the proposed method. The results show that the proposed method is effective to reduce the mean root-mean-square error (RMSE) of position by about 92.53%, 67.93%, 55.97%, and 30.09% compared with the INS only, WSN, EKF, and IEKF.

1. Introduction

As the development of automation indoor mobile robots, how to obtain accurate navigation information of indoor mobile robots has received great attention over the past few decades.

To the integrated system, the global positioning systems (GPS)/inertial navigation systems (INS) integrated system is one of the most used methods for the outdoor navigation. Many attempts try to improve the accuracy of the GPS/INS integration. For example, Quinchia et al. compared different error modeling of MEMS applied to GPS/INS integrated systems in [1], Jwo et al. proposed a fuzzy adaptive strong tracking unscented Kalman filter for ultratight GPS/INS integrated systems [2], Chen et al. proposed a GPS/INS system using novel filtering methods for vessel attitude determination [3], and Jwo et al. proposed a nonlinear filtering with IMM algorithm for ultratight GPS/INS integration [4]. Meanwhile, in order to overcome the GPS outage, some attempts try to design bridge methods by using the artificial intelligence algorithms [5] such as Neural Networks (NN) [6–8] and least squares support vector machine (LS-SVM) [9–11]. However, since the accuracy of the integrated system is depending on the GPS, it has poor performance in the indoor environment. In order to get higher positioning accuracy indoor, some literatures try to employ wireless localization to replace the GPS in the integrated system. For instance, S. J. Kim and B. K. Kim proposed an accurate hybrid global self-localization algorithm for indoor mobile robots with two-dimensional isotropic ultrasonic receivers [12], and an accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements was proposed in [13]. On the basis of the navigation strategy, the data fusion algorithm should also be designed seriously. In this field, Kalman filter (KF) is able to achieve the optimal state estimation [14]. However, it is not suitable for nonlinear systems. Thus, the extended KF (EKF) is proposed to overcome this problem by Taylor series expansion, which may introduce a truncated error [15]. In order to overcome this problem, the iterated EKF (IEKF) is proposed. However, the data fusion algorithms mentioned above are difficult to track the accurate state during the target's fast movement since it employs a fixed priori estimates for the process and measurement noise covariances during the whole estimation process [16].

In order to overcome these problems, we employed the noise statistics estimator in the IEKF, which combines the advantages of the AEKF and the IEKF. The remainder of

the paper is organized as follows. Sections 2 and 3 give the introduction for AIEKF and its application to INS/WSN integrated system. The tests and discussion are illustrated in Section 4. Finally, the conclusions are given.

2. Adaptive Iterated Extended Kalman Filter

In this section, a brief introduction to the EKF and IEKF will be given, and then AIEKF will be proposed. It is assumed that a discrete-time model of a nonlinear system is given by

$$\begin{aligned} \mathbf{x}_k &= f\left(\mathbf{x}_{k-1}\right) + \omega_k, \\ \mathbf{y}_k &= h\left(\mathbf{x}_k\right) + \upsilon_k, \end{aligned} \tag{1}$$

where $\mathbf{x}_k$ and $\mathbf{y}_k$ are the state vector and the measurement vector for the filter, $f(\cdot)$ and $h(\cdot)$ are the dynamic model function and the measurement function, respectively, and ω_k and υ_k are the process noise vector and measurement noise vector, respectively. It is assumed that ω_k and υ_k are independent zero-mean white Gaussian sequences with covariance $\mathbf{Q}$ and $\mathbf{R}$, respectively.

2.1. Extended Kalman Filter. The traditional EKF algorithm is utilizing a set of equations as follows [17]:

$$\widehat{\mathbf{X}}_{k|k-1} = \mathbf{A}_{k|k-1}\widehat{\mathbf{X}}_{k-1|k-1} + \widehat{\mathbf{q}}_k, \tag{2}$$

$$\mathbf{P}_{k|k-1} = \mathbf{A}_{k|k-1}\mathbf{P}_{k-1}\mathbf{A}_{k|k-1}^T + \widehat{\mathbf{Q}}_k, \tag{3}$$

$$\mathbf{K}_k = \mathbf{P}_{k|k-1}{\mathbf{H}_k}^T\left[\mathbf{H}_k\mathbf{P}_{k|k-1}{\mathbf{H}_k}^T + \widehat{\mathbf{R}}_k\right]^{-1}, \tag{4}$$

$$\upsilon_k = \mathbf{y}_k - \mathbf{h}\left(\widehat{\mathbf{X}}_{k|k-1}\right), \tag{5}$$

$$\widehat{\mathbf{X}}_{k|k} = \widehat{\mathbf{X}}_{k|k-1} + \mathbf{K}_k\upsilon_k, \tag{6}$$

$$\mathbf{P}_{k|k} = \left[\mathbf{I} - \mathbf{K}_k\mathbf{H}\left(\widehat{\mathbf{X}}_{k|k}\right)\right]\mathbf{P}_{k|k-1}, \tag{7}$$

where $\mathbf{A}_{k|k-1} = \partial\mathbf{f}(\widehat{\mathbf{X}}_{k|k})/\partial\widehat{\mathbf{X}}_{k|k}$, $\mathbf{H}_k = \partial\mathbf{h}(\widehat{\mathbf{X}}_{k|k})/\partial\widehat{\mathbf{X}}_{k|k}$.

2.2. Iterated Extended Kalman Filter. Compared with the EKF, the IEKF employs a few simple iterative operations to reduce the bias and the estimation error after getting $\mathbf{X}_k$ in (2) and $\mathbf{P}_k$ in (3). The corresponding recursive relations are

$$\begin{aligned} \widehat{\mathbf{X}}_{k|k}^{(1)} &= \widehat{\mathbf{X}}_{k|k-1}, \\ \mathbf{P}_{k|k}^{(1)} &= \mathbf{P}_{k|k-1}, \\ \mathbf{K}_k^{(n)} &= \mathbf{P}_{k|k-1}\left(\mathbf{H}^{(n)}\right)^T\left[\mathbf{H}^{(n)}\mathbf{P}_{k|k-1}\left(\mathbf{H}^{(n)}\right)^T + \mathbf{R}\right]^{-1}, \\ \widehat{\mathbf{X}}_{k|k}^{(n+1)} &= \widehat{\mathbf{X}}_{k|k}^{(n)} + \mathbf{K}_k^{(n)}\left[\mathbf{y}_k - \mathbf{h}^{(n)}\left(\widehat{\mathbf{X}}_{k|k}^{(n)}\right) - \mathbf{H}^{(n)}\right. \\ &\qquad \left.\times\left(\widehat{\mathbf{X}}_{k|k-1} - \widehat{\mathbf{X}}_{k|k}^{(n)}\right)\right], \\ \mathbf{P}_{k|k}^{(n)} &= \left[\mathbf{I} - \mathbf{K}_k^{(n)}\mathbf{H}^{(n)}\right]\mathbf{P}_{k|k-1}^{(n)}, \end{aligned} \tag{8}$$

where $\mathbf{H}^{(n)} = \partial\mathbf{h}(\widehat{\mathbf{X}}_{k|k}^n)/\partial\widehat{\mathbf{X}}_{k|k}^n$ and the superscript n ($n = 1, 2, \ldots, m$) is the number of iteration steps, And then,

$$\begin{aligned} \widehat{\mathbf{X}}_{k|k} &= \widehat{\mathbf{X}}_{k|k}^{(m)}, \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k}^{(m)}. \end{aligned} \tag{9}$$

2.3. Adaptive Iterated Extended Kalman Filter. The EKF overcomes the nonlinear problem by ignoring the higher order terms, which may introduce a truncated error. Thus, the IEKF overcomes this problem. However, it is evident that both the $\mathbf{Q}$ and $\mathbf{R}$ for EKF and those for IEKF are prior estimates. In practice, there are uncertainties in the noise description, and the assumptions on the statistics of disturbances are violated since the availability of precisely known model is unrealistic practical situations. In order to overcome these problems, we employed the noise statistics estimator into the IEKF. Meanwhile, when the system noise is stable and the error variance is small, it is able to employ observation noise estimator only. The corresponding recursive relations are

$$\begin{aligned} \widehat{\mathbf{X}}_{k|k}^{(1)} &= \widehat{\mathbf{X}}_{k|k-1}, \\ \mathbf{P}_{k|k}^{(1)} &= \mathbf{P}_{k|k-1}, \\ \mathbf{K}_k^{(n)} &= \mathbf{P}_{k|k-1}\left(\mathbf{H}^{(n)}\right)^T\left[\mathbf{H}^{(n)}\mathbf{P}_{k|k-1}\left(\mathbf{H}^{(n)}\right)^T + \widehat{\mathbf{R}}_{k-1}^{(n)}\right]^{-1}, \\ \widehat{\mathbf{X}}_{k|k}^{(n+1)} &= \widehat{\mathbf{X}}_{k|k}^{(n)} + \mathbf{K}_k^{(n)}\left[\mathbf{y}_k - \mathbf{h}^{(n)}\left(\widehat{\mathbf{X}}_{k|k}^{(n)}\right) - \mathbf{H}^{(n)}\right. \\ &\qquad \left.\times\left(\widehat{\mathbf{X}}_{k|k-1} - \widehat{\mathbf{X}}_{k|k}^{(n)}\right)\right], \\ \mathbf{P}_{k|k}^{(n)} &= \left[\mathbf{I} - \mathbf{K}_k^{(n)}\mathbf{H}^{(n)}\right]\mathbf{P}_{k|k-1}^{(n)}, \end{aligned} \tag{10}$$

where $\widehat{\mathbf{R}}_k^{(n)}$ is computed by the time-varying noise statistics estimators with the following equations:

$$\begin{aligned} \widehat{\mathbf{R}}_k^{(n)} &= \left(1 - d_{k-1}\right)\widehat{\mathbf{R}}_{k-1}^{(n)} \\ &\quad + d_{k-1}\left(\left[\mathbf{I} - \mathbf{H}_k^{(n)}\mathbf{K}_k\right]\upsilon_k\upsilon_k^T\left[\mathbf{I} - \mathbf{H}_k^{(n)}\mathbf{K}_k\right]^T\right. \\ &\quad \left. + \mathbf{H}_k^{(n)}\mathbf{P}_{k|k-1}^{(n)}\left(\mathbf{H}_k^{(n)}\right)^T\right), \end{aligned} \tag{11}$$

here, $d_{k-1} = (1-b)/(1-b^k)$, $0 < b < 1$. And then,

$$\begin{aligned} \widehat{\mathbf{X}}_{k|k} &= \widehat{\mathbf{X}}_{k|k}^{(m)}, \\ \mathbf{P}_{k|k} &= \mathbf{P}_{k|k}^{(m)}, \\ \mathbf{R}_k &= \mathbf{R}_k^{(m)}. \end{aligned} \tag{12}$$

3. Adaptive Iterated Extended Kalman Filter for Integrated Navigation

In this work, we just consider the navigation information for mobile robot in the relative coordinate. The INS error is the accumulation of errors in each time. In order to achieve

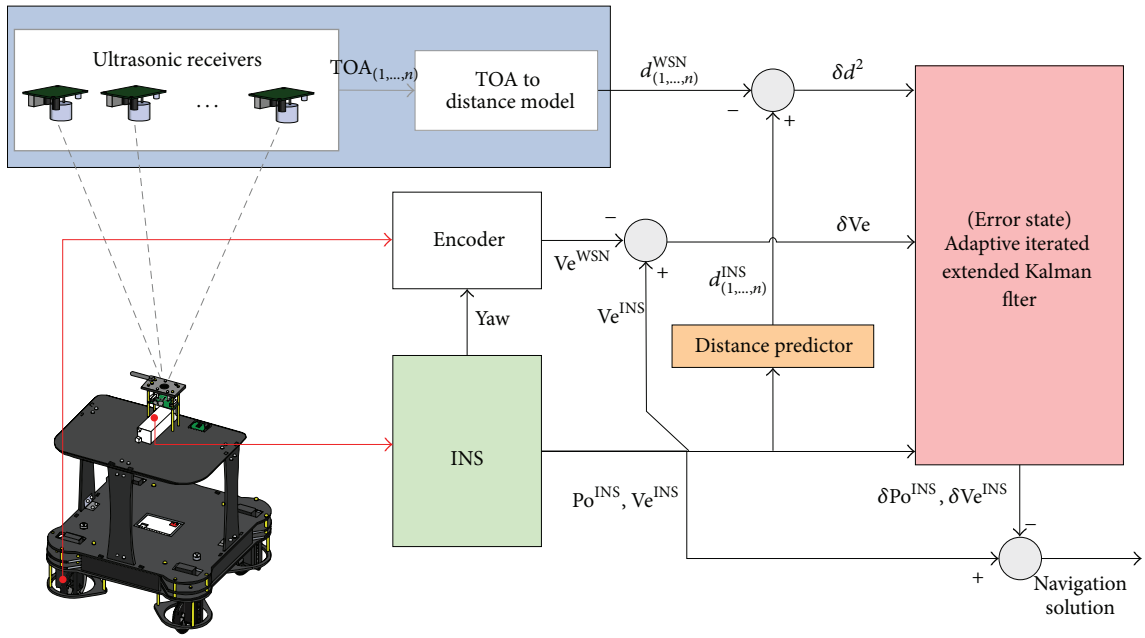


Figure 1: Configuration of the hybrid system.

better estimation accuracy of INS error, the state vector is defined by $\mathbf{x} = [\delta P_E\ \delta P_N\ \delta V_E\ \delta V_N\ \delta \mathrm{Acc}_E\ \delta \mathrm{Acc}_N]$. Here, $(\delta P_{E,k}, \delta P_{N,k})$, $(\delta V_{E,k}, \delta V_{N,k})$, and $(\delta \mathrm{Acc}_{E,k}, \delta \mathrm{Acc}_{N,k})$ are the errors of position, velocity, and accelerometer measured by INS in east and north direction. The system equation for the filter at time k is illustrated in.

$$\underbrace{\begin{bmatrix} \delta P_{E,k} \\ \delta V_{E,k} \\ \delta \mathrm{Acc}_{E,k} \\ \delta P_{N,k} \\ \delta V_{N,k} \\ \delta \mathrm{Acc}_{N,k} \end{bmatrix}}_{\mathbf{x}_k} = \underbrace{\begin{bmatrix} 1 & T & T^2/2 & 0 & 0 & 0 \\ 0 & 1 & T & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & T & T^2/2 \\ 0 & 0 & 0 & 0 & 1 & T \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}}_{\mathbf{A}} \underbrace{\begin{bmatrix} \delta P_{E,k-1} \\ \delta V_{E,k-1} \\ \delta \mathrm{Acc}_{E,k-1} \\ \delta P_{N,k-1} \\ \delta V_{N,k-1} \\ \delta \mathrm{Acc}_{N,k-1} \end{bmatrix}}_{\mathbf{x}_{k-1}} + \mathbf{W}_k, \tag{13}$$

where T is sample time and $\mathbf{W}_k$ is the process noise vector. The measurement equation for the filter at time k is illustrated in.

$$\underbrace{\begin{bmatrix} \Delta V_{E,k} \\ \Delta V_{N,k} \\ \Delta d^2_{1,k} \\ \Delta d^2_{2,k} \\ \vdots \\ \Delta d^2_{m,k} \end{bmatrix}}_{\mathbf{y}_k} = \underbrace{\begin{bmatrix} \delta V_{E,k} \\ \delta V_{N,k} \\ h_{d_1}(\delta P_{E,k}, \delta P_{N,k}) \\ h_{d_2}(\delta P_{E,k}, \delta P_{N,k}) \\ \vdots \\ h_{d_m}(\delta P_{E,k}, \delta P_{N,k}) \end{bmatrix}}_{\mathbf{h}(\mathbf{x}_k)} + \tilde{v}_k. \tag{14}$$

Here, Δd_i^2 is the difference between the distance form reference node (RN) to the mobile robot measured by the INS and WSN, respectively, at time k, and it is expressed as follows:

$$\begin{aligned} \Delta d^2_{i,k} &= \left(d_i^{\mathrm{INS}}\right)^2 - \left(d_i^{\mathrm{WSN}}\right)^2 \\ &= 2\left(P_E^{\mathrm{INS}} - P_E^{\mathrm{RN},i}\right)\delta P_{E,k} + 2\left(P_N^{\mathrm{INS}} - P_N^{\mathrm{RN},i}\right)\delta P_{N,k} \\ &\quad - \left(\delta P^2_{E,k} + \delta P^2_{N,k}\right), \quad i = 1, 2, \ldots, m, \end{aligned} \tag{15}$$

where d_i^{INS} and d_i^{WSN} are the distances from mobile robot to ith RN measured by the INS and WSN, respectively, $(P_E^{\mathrm{INS}}, P_N^{\mathrm{INS}})$ is INS position for mobile robot, and $(P_E^{\mathrm{RN},i}, P_N^{\mathrm{RN},i})$ is ith RN position. And $(\Delta V_E, \Delta V_N)$ is the difference of the WSN and INS velocities in east and north direction, respectively, and $\tilde{v}_k$ is measurement noise vector. The derivation of (15) has a very detailed description in [18]. The configuration of the hybrid system is shown in Figure 1.

4. Indoor Localization Tests and Discussion

4.1. The Architecture of the Integrated Navigation System. In this work, a real testbed is built to evaluate the performance of the proposed method. Figure 2 displays the architecture of the testbed. In this work, the mobile robot (shown in Figure 3) is used to run along the preset trajectory. The IMU fixed to the robot are used to provide INS solution for the position, velocity, and the attitude of the mobile robot. The mobile robot position measured by the WSN is used as ultrasonic sender and the receiver. And the computer is used for saving sensor data.

The sample time used in the test is 0.02 s, and the mobile robot runs along the trajectories shown in Figure 4 with 0.3 m/s. Meanwhile, the RNs are denoted by yellow circles in Figure 4.

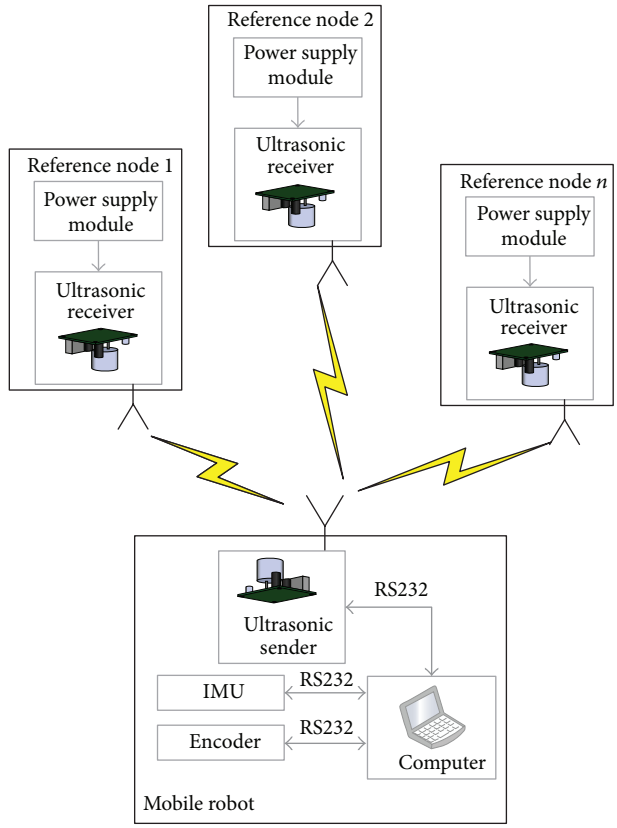


Figure 2: The architecture of the integrated navigation system.

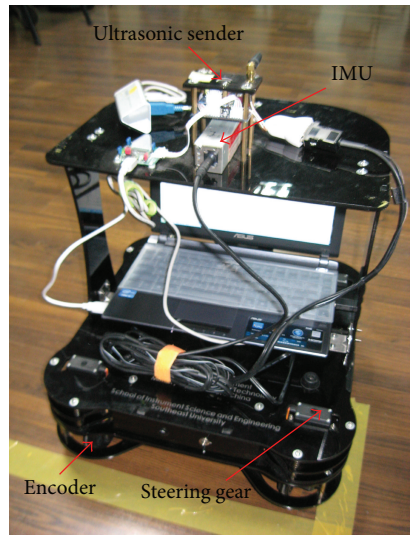


Figure 3: The prototype of the robot.

4.2. The Performance of the Proposed Method. In this section, the performance of the proposed method is discussed. The position errors for the INS only, WSN, EKF, IEKF, and the proposed method are shown in Figure 5.

The east and north position errors of five estimation strategies in the first trajectory are shown in Figures 5(a) and 5(b), respectively. From these figures, it can be seen easily that the INS position error is accumulated. WSN is able to maintain the accuracy of position. It is evident that both the EKF and the IEKF are effective in reducing the position error compared with WSN. The errors for the proposed method are smaller than the ones for the IEKF. Figures 5(c)

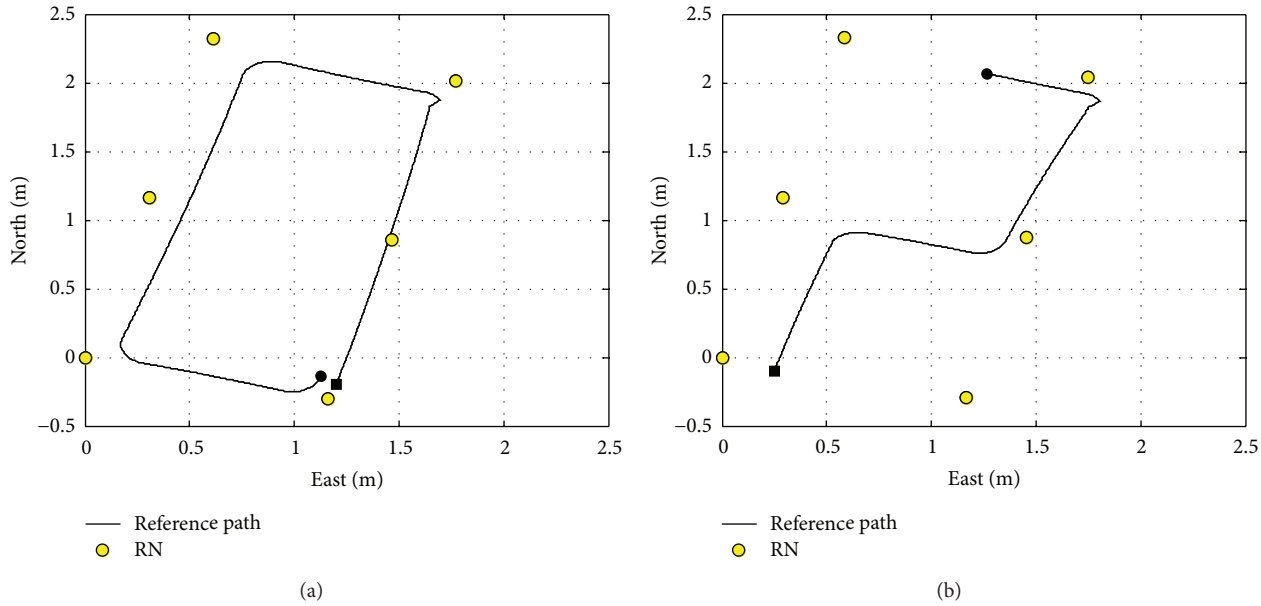


Figure 4: The trajectory of the real test.

Table 1: Comparison of five estimation strategies in terms of position error.

Method	RMSE (m)				Mean
	The first trajectory		The second trajectory		
	East	North	East	North	
INS only	0.5912	0.4590	0.2108	0.3179	0.3947
WSN	0.1132	0.0787	0.1065	0.0697	0.0920
EKF	0.0721	0.0639	0.0736	0.0582	0.0670
IEKF	0.0433	0.0433	0.0462	0.0360	0.0422
The proposed method	0.0333	0.0290	0.0309	0.0249	0.0295

Table 2: Comparison of five estimation strategies in terms of velocity error.

Method	RMSE (m/s)				Mean
	The first trajectory		The second trajectory		
	East	North	East	North	
INS only	0.1391	0.1682	0.1400	0.0957	0.1358
WSN	0.0595	0.0854	0.0650	0.0794	0.0723
EKF	0.0441	0.0539	0.0424	0.0437	0.0460
IEKF	0.0425	0.0556	0.0412	0.0482	0.0469
The proposed method	0.0445	0.0546	0.0420	0.0462	0.0468

and 5(d) display the east and north position errors of five estimation strategies in the second trajectory. Similar to the first trajectory, it is evident that the proposed method has the smallest error.

The comparison of five estimation strategies in terms of position error is shown in Table 1. The results show that the proposed method has the lowest error in east and north direction respectively. The mean root-mean-square error (RMSE) of position for the proposed method is 0.0295 m. It reduces the mean RMSE of position by about 92.53%, 67.93%, 55.97%, and 30.09% compared with the INS only, WSN, EKF, and IEKF.

Table 2 shows the comparison of five estimation strategies in terms of velocity error. It can be seen that the EKF, IEKF, and the proposed method are able to reduce the velocity error compared with the IN S and the WSN, respectively. The result shows that the mean RMSE of velocity for the proposed method is 0.0468 m/s. However, the velocity estimation accuracy for the EKF, IEKF, and the proposed method is close.

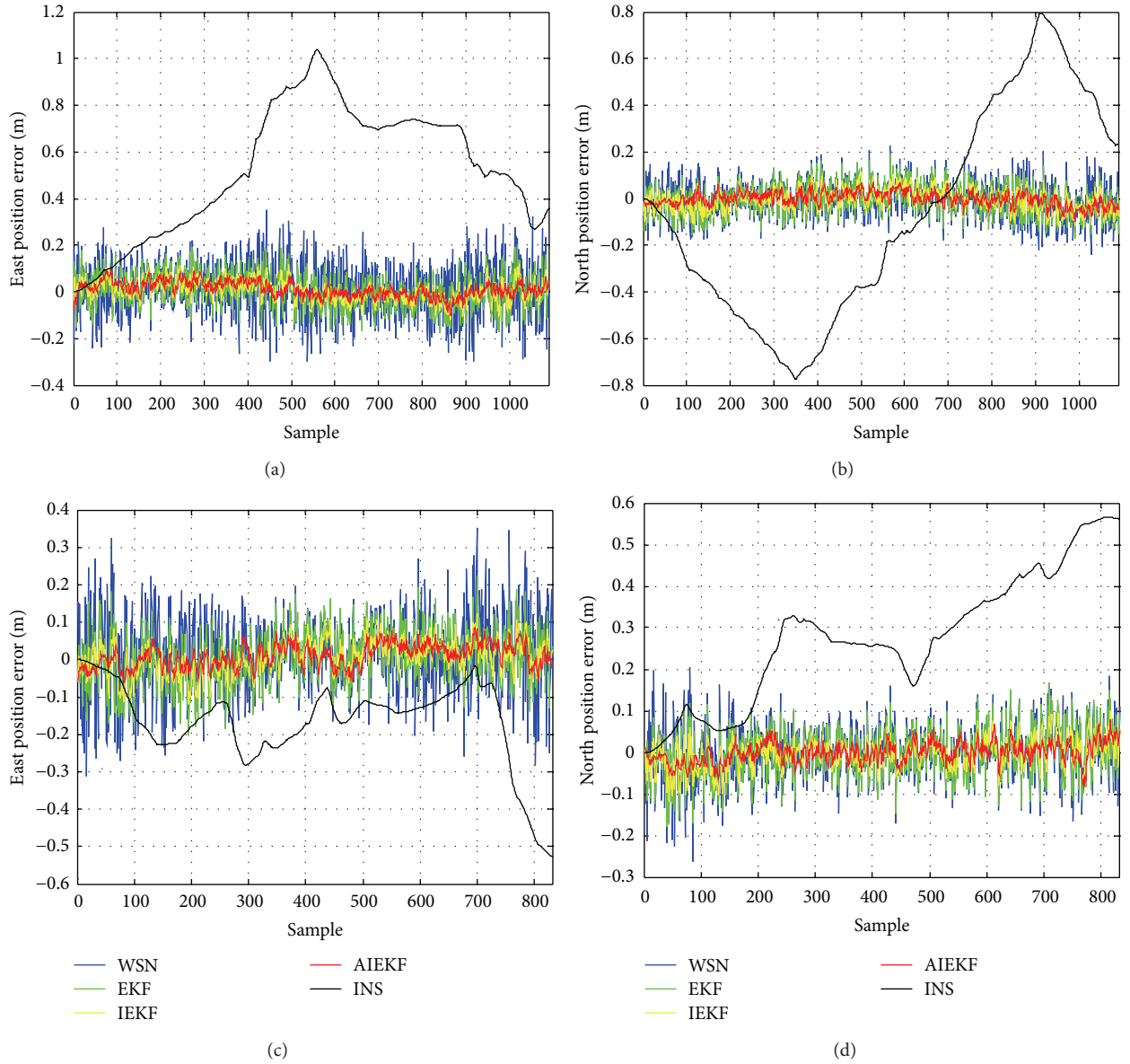


Figure 5: The position errors for the INS only, WSN, EKF, IEKF, and the proposed method. (a) and (b) The first trajectory. (c) and (d) The second trajectory.

5. Conclusions

In this work, the noise statistics estimator is employed into the IEKF to combine the advantages of the AEKF and the IEKF. Then, the AIEKF is used in INS/WSN integrated system. The experimental results show that the proposed method is effective in reducing the position error compared with the INS only, WSN, EKF, and IEKF; however, the velocity estimation accuracy for the EKF, IEKF, and the proposed method is close.

Conflict of Interests

The authors of the paper do not have a direct financial relation that might lead to a conflict of interests for any of the authors.

Acknowledgments

This work was supported in part by the National Natural Science Foundation of China (nos. 51375087, 41204025, and 50975049), Ocean Special Funds for Scientific Research on Public Causes (no. 201205035-09), Specialized Research Fund for the Doctoral Program of Higher Education (no. 20110092110039), the 52 and China Postdoctoral Science Foundation (no. 2012M520967), and the Program Sponsored for Scientific Innovation Research of College Graduate in Jiangsu Province, China (no. CXLX_0101).

References

[1] A. G. Quinchia, G. Falco, E. Falletti, F. Dovis, and C. Ferrer, "A comparison between different error modeling of MEMS applied

to GPS/INS integrated systems," *Sensors*, vol. 13, no. 8, pp. 9549–9588, 2013.

[2] D.-J. Jwo, C.-F. Yang, C.-H. Chuang, and C. H. Lee, "Performance enhancement for ultra-tight GPS/INS integration using a fuzzy adaptive strong tracking unscented Kalman filter," *Nonlinear Dynamics*, vol. 73, no. 1-2, pp. 377–395, 2013.

[3] X. Chen, C. Shen, and Y. Zhao, "Study on GPS/INS system using novel filtering methods for vessel attitude determination," *Mathematical Problems in Engineering*, vol. 2013, Article ID 678943, 8 pages, 2013.

[4] D. J. Jwo, C. W. Hu, and C. H. Tseng, "Nonlinear filtering with IMM algorithm for ultra-tight GPS/INS integration: regular paper," *International Journal of Advanced Robotic Systems*, vol. 10, article 222, 2013.

[5] Y. Zhang, P. Agarwal, V. Bhatnaga, S. Balochian, and J. Yan, "Swarm intelligence and its applications," *The Scientific World Journal*, vol. 2013, Article ID 528069, 3 pages, 2013.

[6] C. M. Bishop, *Neural Networks For Pattern Recognition*, Oxford University Press, New York, NY, USA, 1995.

[7] S. Haykin, *Neural Networks—A Comprehensive Foundation*, IEEE Press, New York, NY, USA, 1994.

[8] A. Noureldin, A. El-Shafie, and M. Bayoumi, "GPS/INS integration utilizing dynamic neural networks for vehicular navigation," *Information Fusion*, vol. 12, no. 1, pp. 48–57, 2011.

[9] Z.-K. Xu, Y. Li, C. Rizos, and X. Xu, "Novel hybrid of LS-SVM and kalman filter for GPS/INS integration," *Journal of Navigation*, vol. 63, no. 2, pp. 289–299, 2010.

[10] Y. Zhang and L. Wu, "Classification of fruits using computer vision and a multiclass support vector machine," *Sensors*, vol. 2012, no. 9, pp. 12489–12505, 2012.

[11] Y. Zhang, S. Wang, G. Ji, and Z. Dong, "An MR brain images classifier system via particle swarm optimization and kernel support vector machine," *The Scientific World Journal*, vol. 2013, Article ID 130134, 9 pages, 2013.

[12] S. J. Kim and B. K. Kim, "Accurate hybrid global self-localization algorithm for indoor mobile robots with two-dimensional isotropic ultrasonic receivers," *IEEE Transactions on Instrumentation and Measurement*, vol. 60, no. 10, pp. 3391–3404, 2011.

[13] A. R. J. Ruiz, F. S. Granja, J. C. P. Honorato, and J. I. G. Rosas, "Accurate pedestrian indoor navigation by tightly coupling foot-mounted IMU and RFID measurements," *IEEE Transactions on Instrumentation and Measurement*, vol. 61, no. 1, pp. 178–189, 2012.

[14] F. Chen and M. W. Dunnigan, "Comparative study of a sliding-mode observer and Kalman filters for full state estimation in an induction machine," *IEE Proceedings: Electric Power Applications*, vol. 149, no. 1, pp. 53–64, 2002.

[15] Z. Chen, R. Rodrigo, V. Parsa, and J. Samarabandu, "Using ultrasonic and vision sensors within extended kalman filter for robot navigation," *Canadian Acoustics*, vol. 33, no. 3, pp. 28–29, 2005.

[16] H. Shao, D. Kim, and K. You, "TDOA/FDOA geolocation with adaptive extended Kalman filter," *Communications in Computer and Information Science*, vol. 121, pp. 226–235, 2010.

[17] A. Gelb, *Applied Optimal Estimation*, The MIT Press, Cambridge, Mass, USA, 1974.

[18] Y. Xu, X. Y. Chen, and Q. H. Li, "Unbiased tightly-coupled INS/WSN integrated navigation based on extended Kalman filter," *Journal of Chinese Inertial Technology*, vol. 20, no. 3, pp. 292–299, 2012.