

# Face Identification based Bio-Inspired Algorithms

Sanaa Ghouzali<sup>1</sup> and Souad Larabi<sup>2</sup>

<sup>1</sup>Department of Information Technology, King Saud University, Saudi Arabia

<sup>2</sup>Computer Science Department, Prince Sultan University, Saudi Arabia

**Abstract:** Most biometric identification applications suffer from the curse of dimensionality as the database size becomes very large, which could negatively affect both the identification performance and speed. In this paper, we use Projection Pursuit (PP) methods to determine clusters of individuals. Support Vector Machine (SVM) classifiers are then applied on each cluster of users separately. PP clustering is conducted using Friedman and Kurtosis projection indices optimized by Genetic Algorithm and Particle Swarm Optimization methods. Experimental results obtained using YALE face database showed improvement in the performance and speed of face identification system.

**Keywords:** Support vector machine, projection pursuit, particle swarm optimization, genetic algorithms, Kurtosis index, Friedman index.

Received February 23, 2017; accepted June 12, 2017

<https://doi.org/10.34028/iajit/17/1/14>

## 1. Introduction

Recently biometric authentication based security and access control applications have been mainly investigated in the computer community. Biometric based authentication systems consist of human identification that depends on some physiological or behavioral characteristics (e.g., face, gait ... etc.). These systems can be categorized as verification or identification. The verification type consists of either admitting or refusing a user's requested identity whereas the identification type is performed, without any request of user identity, by operating matches against the system database of previously stored biometric templates in order to detect the existence of a user biometric record or not. The last type is costly, time consuming and results in low performance because most real-world biometric identification applications (e.g., border control, criminal investigation, missing children identification, corpse identification ... etc.) have a large scale database that can be estimated to numerous thousands.

Support Vector Machines (SVM) based machine learning technique has been used to solve several classification problems because of its ability to offer a good compromise between complexity of implementation and time of execution contrary to other techniques such as Neural Network, Bayesian classification ... etc., More details about SVM can be found in [47]. However in the case of very large datasets, training SVM cannot be solved [7]. Yet, the SVM training problem deserves to be investigated.

Several research studies have been performed in this area and many have suggested to split the large scale problem into small sub problems in order to reduce the complexity of training SVMs. Following the same

principle, in this paper the users' database will be first divided into clusters owing significant similarities, and then SVM classifier will be employed on each cluster separately.

Exploratory data analysis methods are judiciously employed to detect relevant and hidden information from large volumes of multidimensional data. This information can be interesting structures such as clusters, outliers ...etc., Principle Component Analysis (PCA) [2] is one common exploratory data analysis method that seeks to maximize the dispersion of the observations. However, PCA is not efficient due to its inability to extract potential interesting structures. Another type of exploratory data analysis methods is Projection Pursuit (PP). PP involves determining interesting low dimensional projections of high-dimensional multivariate data by optimizing an index related to each projection space [21, 24]

The main objective in this paper is to investigate the use of PP to reduce the complexity of SVM related to the increasing number of users in real biometrics based identification applications. PP is applied in this research work to determine clusters of users possessing important resemblances and then SVM classifier is applied on each cluster of users independently. Therefore, this allowed increasing the performance accuracy rate and speed of classification when huge number of users is used.

The remainder of the paper is as follows. In section 2 an overview of related work is given. A detailed description of the proposed method is presented in section 3. Experimentation results and discussion are provided in section 4, section 5 draws conclusion and future work.

## 2. Related Work

SVMs have gained popularity in the field of pattern classification due to its promising empirical performance, moderate computation complexity and its strong mathematical foundation. However the training complexity of SVMs is highly dependent on the size of the dataset. Standard SVM training has  $O(m^3)$  time and  $O(m^2)$  space complexities, where  $m$  is the training set size. It is thus computationally infeasible on very large datasets as it requires to solve a very large-scale Quadratic Programming (QP) optimization problem. Several attempts have been made to solve this problem but mainly in the case of a two-class SVM. Multi-classification SVM is more complex for large datasets than binary classification, even in the case that both have the same amount of training data [20].

To solve SVM training problem, Sequential Minimization Optimization (SMO) technique has been proposed in [39]. Using SMO, the large QP problem is broken into series of smallest possible QP problems which are analytically solved with Lagrange Multipliers while satisfying the Karush-Kuhn-Tucker (KKT) conditions. This method consumes lots of memory space to save all the sub problems and requires the KKT conditions to be checked for each sub problem. Inspired from the core set-based approximate Minimum Enclosing Ball algorithms in computational geometry, Tsang *et al.* [46] have developed an approximation algorithm for SVM training called Core Vector Machines. Generally speaking, in an optimization problem, a core set is a subset of input points such that a good approximation to the original input can be found by solving the optimization problem directly on the core set.

Another solution consists in scaling down the training data before inputting it to the SVM classifiers. The simplest method would be just to sample from the original dataset and use the sample to train SVM. Pavlov *et al.* [37] used boosting to combine a large number of SVMs, each is trained on only a small data subsample. While this method is not optimal in general, it allows for very fast training of SVMs, has substantially lower memory cost and yields performance close to that of the full SMO. Alternatively, Collobert *et al.* [10] proposed a parallel SVM training and classification algorithm that each subset of a dataset is trained with SVM and then these small SVM classifiers are combined into a final single classifier using a neural-network-based method. Lee and Mangasarian [29] proposed the reduced SVM, which uses a random rectangular subset of the kernel matrix to greatly reduce the size of the quadratic program to be solved and simplify the characterization of the nonlinear separating surface.

Instead of random sampling, one can intelligently sample a small number of training data for SVM training. Authors in [42, 45] have proposed to use

active learning to select training samples that are expected to minimize the estimated future error. This heuristic has shown that SVM trained on a well-chosen subset of the available offers better performance at less time and lower cost than the naive approach of training on all available data. However these methods are not immune to selecting outliers. In [36], authors used squashing that allows to scale the data down while preserving its statistical properties. Training SVMs on squashed datasets has been fast, decreased memory requirement and produced classification accuracy that is closed to that of the full data. In [5], authors used editing by selectively removing samples from the training set using probabilistic estimates. The removal procedure aimed at creating a separable distribution of training examples without modifying the location of the decision boundary. Boley and Cao [4] used clustering to partition the training data into several pair-wise disjoint clusters by exploiting the distributional properties of the training data. Then, the representatives of these clusters are used to train an initial support vector machine, based on which are approximately identified the support vectors and non-support vectors. After replacing the cluster containing only non-support vectors with its representative, the number of training data can be significantly reduced, thereby speeding up the training process. Similarly, Yu *et al.* [49] applied a hierarchical micro-clustering algorithm that scans the entire dataset only once to provide an SVM with high quality samples that carry the statistical summaries of the data such that the summaries maximize the benefit of learning the SVM. Their method has shown to be scalable in terms of the training efficiency while maximizing the performance of SVMs.

PP has effectively been developed in several domains such as clustering [38], regression analysis [41], classification and others. In classification PP has been employed to find low-dimensional projections that expose dissimilarities between categorized groups. The study of combining classification with PP has been started since 1969 [48], and then more attempts have been developed [6, 18, 22]. In the following the related recent studies are addressed. Note that these studies focused on solving the curse of dimensionality in large dimension small sample size data. In this paper, we are concerned about solving large sample size classification problems.

In [26], a new projection pursuit index is proposed based on the Linear Discriminant Analysis (LDA) for solving supervised classification problems. This index is a generalization of the one-dimensional LDA projection indices developed in [11, 13, 23]. It is dedicated to reduce the number of variables and obtain a better comprehension of the class structure in the data space. It has been noticed that this index is robust in detecting outliers. The authors encouraged the use of PP methods together with multivariate tree methods

and several projection pursuit indices.

In [17], the authors combined PP with Neural Network method (called Projection Pursuit networks) that reduces the dimensionality and uses clustering. They suggested a new (one, two)-dimensional PP index based on the quality of projected clusters. The index is dedicated to identify interesting linear projections by localizing clusters that contain different classes. The authors suggested further studies to improve PP network.

In [27], PP is applied to circumvent the curse of dimensionality where a new PP index based on Penalized LDA (called PDA) is proposed for classification. The results show that the performance of PDA index depends on the value assigned to the parameter  $\lambda$ . The PDA index is mainly recommended to be used when the number of features is large compared to the number of samples. In [28], a new supervised classification method called PPTree is addressed. It consists of combining PP with Tree classification method using visualization representation. A recursive binary partition method along with PP are applied. The performance of PPTree highly depends on the characteristics of the datasets and the choice of PP index in each node of the tree. In [12], a framework to circumvent the curse of dimensionality in classification is developed. It includes two phases: the compaction phase to reduce the dimension of the data and PP phase to find the optimal projections utilizing the compacted illustration as input. The results are encouraging. However, the choice of the compaction method and the PP index depends on the dataset. In [9], the paper depicts visual statistical inference technique to solve the curse of dimensionality problem. PP is applied to reduce the dimension of the data using LDA [26] and PDA [27] indices. Then, PP optimization procedure in 'tourr' package is performed. The authors recommended PDA index in high-dimensional, low sample size data.

### 3. Proposed Method

SVM has been applied successfully to solve different classification problems, but large-scale problems such as biometrics based user identification are computationally difficult to solve using traditional SVMs. In this paper, we propose to decompose the complicated large-scale biometrics database into several relatively simpler sub-clusters using Projection Pursuit methods, and for each sub-cluster an SVM is used for classification to determine the user identity. In this section we will briefly describe the proposed approach.

#### 3.1. PP Approach

PP is a family of data analysis methods looks for interesting low-dimensional projections of high-dimensional multivariate data [24, 44]. Low-dimension

is expected to be (one-, two-, and three-) dimension based on human perception. PP has successfully been employed in different domains (biology, bioinformatics, image processing, biometry, etc.). PP emphasizes on the designation and the optimization of a function called projection index related to each projection space.

A projection index  $I(a)$  defines and determines "interesting" structures depending on a standardized projection vector  $a$  by finding the degree of nonlinear structure present in the distribution of the projected data. Let denote by  $X:N \times P$  the data set matrix of  $N$  observations representing the users' images and  $P$  variables.  $X_i$  is the  $i$ -th column vector in  $R^P$  associated with the  $i$ -th observation. The current study is performed on one-dimensional projection. The projection vector is defined from  $R^P$  to  $R$  such as  $Z=Xa$ ,  $a$ , is a  $P$ -dimensional vector defining the linear transformation, and  $Z$  is a  $N$ -dimensional vector corresponding to the coordinates of the projected observations. So, the problem leads to optimize a selected projection index in order to find the projection  $a$ .

The optimization of the projection index was done by means of exact optimization methods (e.g., Newton, Steepest ascent, etc.) but these methods met some shortcomings [15, 34, 44]. In this paper, Genetic Algorithm (GA) and Particle Swarm Optimization (PSO) are used. GA and PSO have been successfully applied to solve different complex problems (e.g., [1, 50, 32]). The performance of these selected methods combined with PP is published in [3, 31].

According to the literature review, two types of PP indices are used based on class information or dedicated to the unsupervised tasks. The first type helps find the important variables that best separate classes. In this study, we are interested on one-dimensional indices that are dedicated in revealing clusters (unsupervised tasks) and we used the Friedman [15] and Kurtosis [38] indices. There exist other useful projection indices such as the discriminant index [3] among others.

#### 3.1.1. The Projection Indices

- *The Friedman index*: belongs to the polynomial-based (Legendre, Hermite and Natural Hermite) family indices. It is based on an estimation of a criterion measuring the departure from the Gaussian distribution using expansions constructed on orthogonal polynomials. The Friedman index is to be maximized. This index can be written in the form:

$$\int_R (P_Y(y) - \varphi(y))^2 g(y) dy \quad (1)$$

Where  $P$  is the density of the projected data,  $\varphi$  is the univariate standard normal density and  $g$  is a weight function.

- *The kurtosis index*: is a one-dimensional projection pursuit index based on the fourth moment of the distribution of the projected data (knowing that the data is standardized). The maximization of this coefficient results in the detection of outlying observations in the projections, while its minimization leads to the determination of groups [38]. In this study, we focus on the minimization of the kurtosis coefficient of the projected data to determine clusters. It is defined as:

$$I_K(a) = \sum_{i=1}^N (a^T X_i)^4 \quad (2)$$

This index is known by its fast computation however it usually determines bimodal projections.

### 3.1.2. The Optimization Methods

To find out interesting structures, the projection index  $I(a)$  should be optimized. As stated above, GA and PSO are applied. In the following, exhaustive descriptions of both algorithms are given along with the pseudo codes.

- *Genetic Algorithm*: is an evolutionary algorithm inspired by Darwinian evolution biology [19]. It is composed of a population of individuals randomly generated in the search space, represented by a specific encoding (real encoding in this work) and stands for the projection vector defined above. At each iteration, each individual is evaluated, and/or selected to produce a new population to be used for the next iteration. The evaluation of every individual is performed by the projection index to be optimized. The reproduction of new individuals is done as follows. A tournament selection of  $n$  participants ( $n=3$ ) and a  $k$ -point crossover ( $k=2$ ) with a probability equal to 0.65 applied to the entire population. Once the selection and the crossover stages are achieved, a different (new) population of individuals is generated either immediately copied or created by crossover. Finally, a mutation operator is employed to each individual using a probability equal to 0.05 by choosing arbitrarily one gene to be substituted by a random real value. The GA used here is deduced from [14, 19, 40, 43].
- *Particle Swarm Optimization*: is a population based stochastic optimization method inspired by social behavior of bird flocking or fish schooling [25]. In PSO, a population called “swarm” is initialized by random particles covering the search space. The particle represents the projection vector. Every particle is assessed by means of the projection index to be optimized. The position of each particle is updated at each iteration according to its best position achieved so far (called  $pbest$ ), the best position of the best particle of the swarm (called global best and denoted by  $gbest$ ) and the velocity which is randomly initialized. Moreover, the

particle position may be concerned by its neighbors, especially the best neighbor particle (called the local best and denoted by  $lbest$ ). In this case, the particles do not investigate the entire search space but move in near vicinity to the local best. This work used the original version introduced and modified by [25, 35]. In this implementation,  $V_{max}$  parameter is applied to avoid a rapid moving of particles from one area to another in the search space and prevent explosion of the particles search.

$$V_{max} = \frac{(Projection_{max} - Projection_{min})}{2} \quad (3)$$

$$V_{min} = -V_{max} \quad (4)$$

In order to examine concurrently several areas of the search space to get first local optima and then a global optimum, a Cosine neighborhood notion is proposed. It consists of splitting the swarm into several groups such that the cosine angle between each projection vector (particle) in the same group should not exceed 30 degree (for more detail see [25]). In each group, the particle travels pursuant its best position ( $pbest$ ) and the best position of its group ( $lbest$ ). As soon as the positions of all the particles are updated, this configuration is updated by the formation of novel groups. The number of iterations and the number of individuals/particles are chosen so that the obtained optimum is difficult to be enhanced (using numerous tests).

The proposed GA algorithm (Algorithm 1) and PSO algorithm (Algorithm 2) are detailed in [30] and summarized as follows. Note that, the data set and the projection index can affect the parameters setting.

*Algorithm 1: Genetic Algorithm (GA)*

```
#population size is N
#Max number of iterations is max_iterations
While (iteration number ≤ max_iterations)
{
  Evaluate the projection_index
  for each individual {
    Perform a tournament selection of 3 participants
    #Perform genetic operators:
    Crossover using probability = 0.65
    Mutation using probability = 0.05
  }
}
Select the best_projection.
```

*Algorithm 2: Particle Swarm Optimization (PSO)*

```
#Swarm size is N
#Max number of iterations is max_iterations
#getBestNeighbor: a function to get a particle's best
neighbor (lbest) among the number of neighbours = 3
#rand( ): is a random number uniformly distributed in [0,1];
CI=0.7
while (iteration number ≤ max_iterations)
{
  while (Number of particles ≤ N)
```

```

{
    Update position: position = position + Velocity
    Calculate fitness (the projection_index value)
    If (fitness > fitness(pbest))
    {
        fitness(pbest) = fitness
        # update the best position
        pbest = position (newly_updated)
    }
    # update the best neighbor
    lbest = getBestNeighbor
    # update Velocity
    Velocity = C1 * Velocity + rand() * (pbest - position) + rand() * (lbest - position)
}
}
Determine the best projection (position)

```

### 3.2. PP-SVM Approach

The aim of the proposed PP-SVM approach is to improve the classification rate obtained by SVM and minimize the required time. SVM combined with PP have not yet been applied for biometric identification task which is the motivation of the present work. The combination of PP-SVM is explained as follows. PP is used first to divide the training data set into two distinct groups and subgroups (clustering step) and then SVM is applied to these obtained groups/subgroups (sub-classification step). PP may be applied several times to get different groups levels until the classification rate is satisfactory as shown in Figure 1.

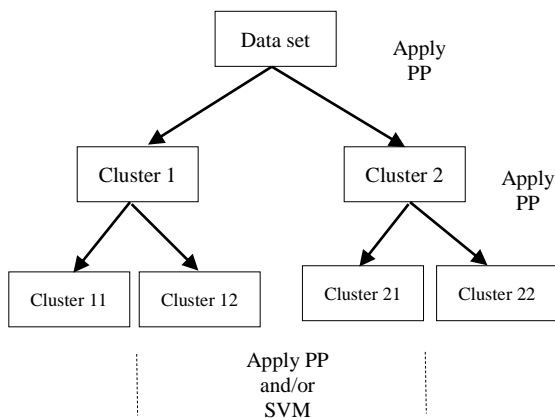


Figure 1. PP-SVM hybrid approach.

## 4. Experimentation and Discussion

### 4.1. Experimental Setting

The proposed approach is validated in this paper using a face based identification system. Experimentations are conducted using YALE [2] face database which consists of 15 individuals, where for each individual, there are 11 face images containing variations in illumination and facial expression. 5 images per user are used for training and the remaining images are used for testing. In order to remove the images' unnecessary

information and preserve only the internal structures of the face image contributing maximum in face recognition (such as eyebrows, eyes, nose, lips and surrounding area), the samples were hence cropped.

To extract face features, we used in this experimentation Laplacian Smoothing Transform (LST) followed by Discriminant Analysis via Support Vector (SVDA) as described in [34]. Since feature extraction process does not fall in the scope of this paper, thereby, it is not described here.

We considered Friedman and Kurtosis projection indices optimized by GA and PSO methods. The number of chromosomes (resp. particles) for GA (resp. PSO) was set to 100 (resp. 50) and the number of iterations was set to 50 for GA and 100 for PSO after performing several runs and examining the projection index convergence to local optima. Each experiment is run 100 times to make sure that the structures detected are unchanged for diverse values of the parameters set above.

For SVM classification, we used the library Libsvm based on SMO [8] implemented in Matlab. The SVM classifier is trained in these experiments using 'quadratic polynomial' that provided better performance compared to the linear and radial basis functions in our simulations. C, a constant that controls the trade-off between the training error and the margin, was set to '100'.

### 4.2. PP Based Classification

The first experiment consisted of applying PP to YALE training dataset in order to detect the maximum number of classes. The GA and PSO optimization algorithms were run 100 times on YALE training dataset. As we used one-dimensional PP, it was difficult to detect all the classes. It is thus recommended to use several projections in PP method because no single projection might hold the information of the multivariate data. Accordingly, diverse interesting projections associated with the global and local optima for the Friedman index obtained by GA (resp. PSO) have discovered 14 classes (resp. 15) out of 15. To check the accuracy of the above result, PP has also been applied to YALE testing dataset, which revealed the same classes for both methods. In order to confirm that the result found using YALE training dataset is similar to that obtained using YALE testing dataset, we compared their corresponding projections associated with the global optimum for the Friedman index using both datasets (Train and Test). This comparison is performed by means of a well-known statistical plot named scatter plot, which is employed to find out the relationships between variables in a dataset. Figure 2 displays the relation among variable 'V1' that represents the projection associated with the global optimum for the Friedman index and GA using YALE training dataset

and variable 'V2 that stands for the projection associated with the global optimum for the same index and method using YALE testing dataset. This plot shows that the projections are similar and there is a positive relation, which means the existence of at least two groups. The result using PSO is the same to that provided by GA.

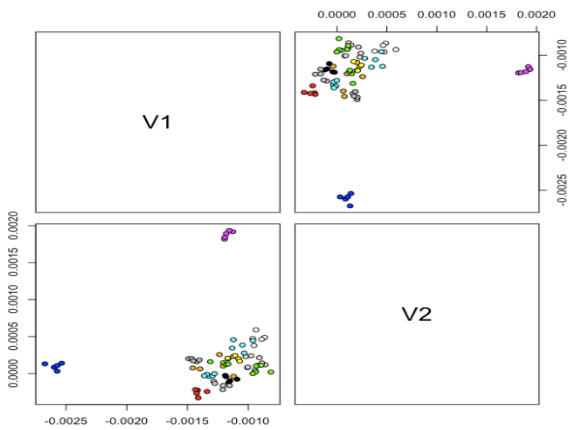


Figure 2. Scatter plot matrix for the Yale data (V1 =the projection associated with the global optimum using YALE training, V2 = the projection corresponding to the global optimum using YALE testing).

The Kurtosis index does not yield the same results for both PSO and GA optimization methods using YALE database. In fact this index divides the dataset into two clusters including different classes. This result can be expected because the Kurtosis index aims to maximize the bimodality of the projections.

### 4.3. PP-SVM Classifier

Although PP based classification approach detected most of the classes as seen in the first experiment, it is not suitable for user identification in the case of large datasets as the process is computationally intensive and time consuming.

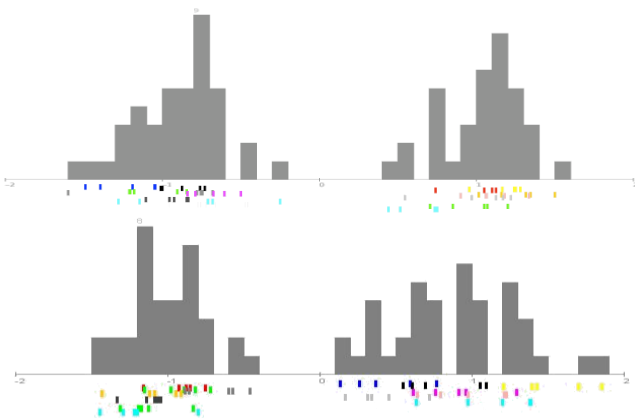


Figure 3. Histograms of the distribution of the projected data of one interesting projection corresponding to one local optimum for the Friedman index using GA (top) and PSO (bottom). Each color specifies one user (class).

Hence, the second experiment consisted of applying PP to YALE training dataset in order to detect different clusters of

users/classes. By employing the Friedman index using GA and PSO, two separate clusters containing several classes of users are found as displayed in Figure 3.

The Kurtosis index also gives good results in terms of interesting structures. Figure 4 shows the interesting projection associated to the global optimum for the Kurtosis index with GA (at top) and PSO (at bottom).

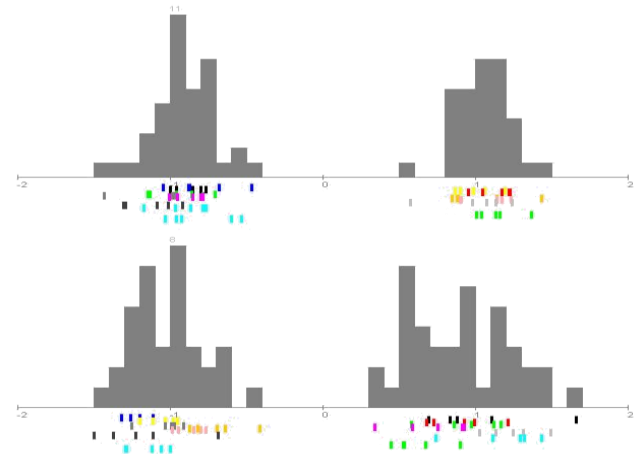


Figure 4. Histograms of the distribution of the projected data of one interesting projection corresponding to one local optimum for the kurtosis index using GA (top) and PSO (bottom). Each color specifies one user (class).

For each cluster of individuals (as determined by PP), we have separately trained an SVM classifier to build a model which is later used to predict the target values of the testing dataset. We have obtained the results displayed in Table 1.

Giving the classification accuracy rate of 87.78% after applying only SVM on the entire database (without clustering), we have obtained a gain of 1.95% when either using Friedman or Kurtosis with GA. The results using GA and Friedman have been published in [16]. For PSO, the gain is 3.58% with Friedman and 3.14% with Kurtosis.

Table 1. SVM Classification rate (%) for the projections obtained by the Kurtosis and Friedman indices using PSO and GA.

	Projection obtained by GA		Projection obtained by PSO	
	Cluster 1	Cluster 2	Cluster 1	Cluster 2
<b>Friedman</b>	93.75	85.71	92.86	89.58
<b>Kurtosis</b>	93.75	85.71	88.1	93.75

### 4.4. Hierarchical PP-SVM Classifier

Several studies have suggested the iterative use of PP (for example PPTree [28]). In this experiment, we have applied PP on each of the clusters previously obtained in the previous section in order to detect different sub-clusters of users/classes. When employing the Friedman index using GA and PSO, we found two separate sub-clusters for each method in each of the previously detected clusters as shown in Figures 5 and 6.

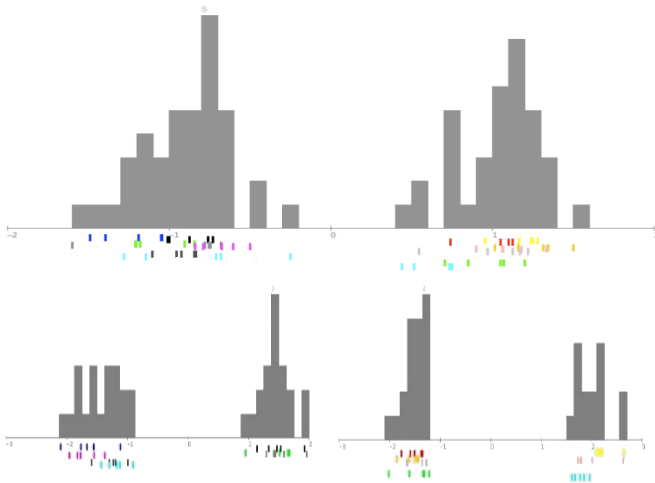


Figure 5. Hierarchical PP: interesting projections associated with one local optimum (top) and the global optimum (bottom left and right) for the Friedman index after applying PP on the entire Yale dataset (top), on the cluster 1 (bottom left) and on the cluster2 (bottom right) using GA.

When applying SVM on the nodes (sub-clusters in level 2) of the hierarchical clustering tree obtained by GA (resp. PSO), we have obtained a gain of 5.28% (resp. 3.89%) compared to the classification rate of the entire database and a gain of 3.32% (resp. 0.31%) compared to the classification of the clusters in level 1 as shown in Table 2.

Table 2. SVM Classification rate for the projections obtained by the Friedman indices using PSO and GA.

	SVM performance for the projection obtained by GA in %				SVM performance for the projection obtained by PSO in %			
Level 1	93.75	85.71	92.86	89.58				
Level 2	100	100	83.33	88.89	100	87.5	95.83	83.33

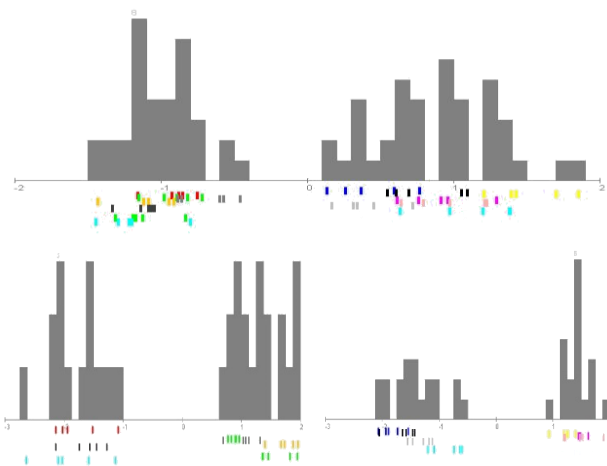


Figure 6. Hierarchical PP: interesting projections associated with one local optimum (top) and the global optimum (bottom left and right) for the Friedman index after applying PP on the entire Yale dataset (top), on the cluster 1 (bottom left) and on the cluster2 (bottom right) using PSO.

Kurtosis index is known for its simple complexity as it represents the fourth moment, henceforth it is unable to determine well separated sub-clusters from those obtained using GA and/or PSO due to the

complexity of the data set. To circumvent this limitation, we proposed to apply Friedman index to determine sub-clusters using both clusters obtained by the kurtosis index from the whole data set. Figure 7 (resp. Figure 8) displays interesting projection corresponding to the global optimum for the kurtosis index with GA (resp. PSO) using the whole data set (top) and interesting projections corresponding to the global optimum for the Friedman index with GA (resp. PSO) using only the observations of cluster 1 (bottom left) and cluster 2 (bottom right) respectively.

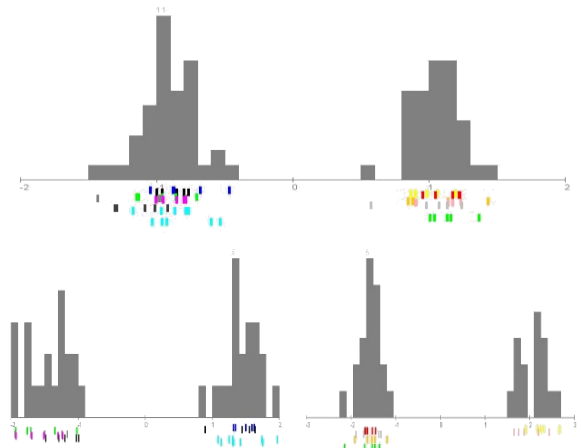


Figure 7. Hierarchical PP: interesting projection corresponding to the global optimum for the kurtosis index after applying PP on the entire Yale dataset (top), and interesting projections associated with the global optimum for the Friedman index when applying PP on the cluster 1 (left bottom) and on the cluster 2 (right bottom) using GA.

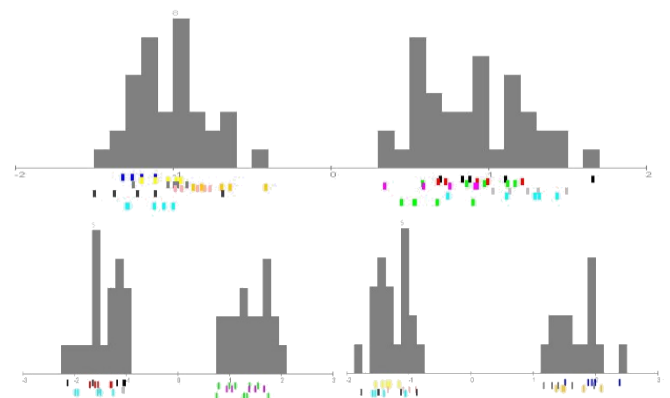


Figure 8. Hierarchical PP: interesting projection corresponding to the global optimum for the kurtosis index after applying PP on the entire Yale dataset (top), and interesting projections associated with the global optimum for the Friedman index when applying PP on the cluster 1 (bottom left) and on the cluster 2 (bottom right) using PSO.

When applying SVM on the nodes (sub-clusters in level 2) of the hierarchical clustering tree obtained by GA (resp. PSO), we have obtained a gain of 3.19% (resp. 6.66%) compared to the classification rate of the entire database and a gain of 1.24% (resp. 3.52%) compared to the classification of the clusters in level 1 as shown in Table 3.

Table 3. SVM Classification rate for the projections obtained by the Kurtosis and Friedman indices using PSO and GA.

	SVM performance for the projections obtained by GA in %				SVM performance for the projections obtained by PSO in %			
	Kurtosis		Friedman		Kurtosis		Friedman	
Level 1	93.75	85.71	88.1	93.75	88.1	93.75	85.71	93.75
Level 2	100	91.67	83.33	88.89	91.67	94.44	95.83	95.83

#### 4.5. Discussion

Different experimentations have been performed, the first experiment shows the ability of PP to find different and several classes (clusters) without using a classification technique. However, this strategy is not always efficient specially when the dataset is huge and the number of classes is large. The second experiment (PP-SVM classifier) consists of applying PP once followed by SVM on the obtained clusters. The result is promising compared to that obtained using only SVM. In the last experiment (Hierarchical PP-SVM classifier) PP and SVM are applied several times until reaching satisfying classification rate. This last experiment gives an excellent result.

According to the computation time, the kurtosis is the fastest index compared to the Friedman index. The Friedman index is slightly longer than the kurtosis index but it is still very acceptable. Table 4 displays the average time per run for each index with GA and PSO.

Table 4. Computation time for both indices using PSO and GA.

	Friedman Index	Kurtosis Index
GA	5s 23ms	325ms
PSO	5s 737ms	6s 538ms

PP is fast and does not take time to find clusters. As illustrated, GA is faster than PSO. In addition, the computation time for SVM is highly proportional to the size of the dataset, so applying SVM on part of the dataset is highly efficient in terms of time and results.

#### 5. Conclusions

Biometric based identification systems suffer from the curse of dimensionality especially in large databases of users. This paper presented the proposed PP-SVM approach to bypass this problem. The proposed approach will help to accurately establish biometric identification systems for huge and enormous datasets. The preliminary results conducted on YALE face database showed that the proposed approach is efficient in enhancing the recognition rate of users by diminishing the complexity of SVM. However, more experiments have to be conducted on large datasets to confirm the obtained results.

The proposed approach can be improved further by employing another PP index based on class information such as within-between variance using the Linear Discriminant Analysis (LDA) index [24, 28]. Moreover, the performance results can be improved

using hierarchical SVM-PP approach if additional interesting structure providing more than two clusters are found.

#### Acknowledgment

The authors would like to express their gratitude and appreciation to King Abdulaziz City for Science and Technology (KACST) for providing the research grant ARGP-35-107.

#### References

- [1] Awad M., "Forecasting of Chaotic Time Series Using RBF Neural Networks Optimized by Genetic Algorithms," *The International Arab Journal of Information Technology*, vol. 14, no. 6, pp. 826-834, 2017.
- [2] Belhumeur P., Hespanha J., and Kriegman D., "Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection," *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 711-720, 1997.
- [3] Berro A., Marie-Sainte S., and Ruiz-Gazen A., "Genetic Algorithms and Particle Swarm Optimization for Exploratory Projection Pursuit," *Annals of Mathematics and Artificial Intelligence*, vol. 60, no. 1-2, pp. 153-178, 2010.
- [4] Boley D. and Cao D., "Training Support Vector Machine Using Adaptive Clustering," in *Proceedings of the 4<sup>th</sup> SIAM International Conference on Data Mining*, Lake Buena Vista, pp. 126-137, 2004.
- [5] Bakir G., Bottou L., and Weston J., "Breaking SVM Complexity with Cross-Training," in *Proceedings of the 17<sup>th</sup> International Conference on Neural Information Processing Systems*, Vancouver, pp. 81-88, 2004.
- [6] Bridle J., Heading A., and MacKay D., "Unsupervised Classifiers, Mutual Information and Phantom Targets," in *Proceedings of the 4<sup>th</sup> International Conference on Neural Information Processing Systems*, Denver, pp. 1096-1101, 1991.
- [7] Burges J., "A Tutorial on Support Vector Machines for Pattern Recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 121-167, 1998.
- [8] Chang C. and Lin C., "LIBSVM: a Library for Support Vector Machines," *ACM Transactions on Intelligent Systems and Technology*, vol. 2, no. 3, pp. 1-27, 2011.
- [9] Chowdhury N., Cook D., Hofmann H., Majumder M., Lee E., and Toth A., "Using Visual Statistical Inference to Better Understand Random Class Separations In High Dimension, Low Sample Size Data," *Computation Statistique*, vol. 30, no. 2, pp. 293-316, 2015.



- [10] Collobert R., Bengio S., and Bengio Y., "A Parallel Mixture of Svms for Very Large Scale Problems," *Neural Computation*, vol. 14, no. 5, pp. 1105-1114, 2002.
- [11] Duda R., Hart P., and Stork D., *Pattern Classification*, Wiley, 2001.
- [12] Espezua S., Villanueva E., Maciel C., and Carvalho A., "A Projection Pursuit Framework For Supervised Dimension Reduction of High Dimensional Small Sample Datasets," *Neuro Computing Journal*, vol. 149, pp. 767-776, 2015.
- [13] Fisher R., "The Statistical Utilization of Multiple Measurements," *Annals of Eugenics*, vol. 8, no. 4, pp. 376-386, 1938.
- [14] Fogel L., Owens A., and Walsh M., *Artificial Intelligence through Simulated Evolution*, Wiley, 1966.
- [15] Friedman J., "Exploratory Projection Pursuit," *Journal of the American Statistical Association*, vol. 82, no. 1, pp. 249-266, 1987.
- [16] Ghouzali S. and Marie-Sainte S., "Support Vector Machine Based Exploratory Projection Pursuit Optimization for User Face Identification," in *Proceedings of International Conference on Information and Communication Technology Research*, Abu Dhabi, pp. 322-325, 2015.
- [17] Grochowski M. and Duch W., "Projection Pursuit Constructive Neural Networks Based on Quality of Projected Clusters," in *Proceedings of International Conference on Artificial Neural Networks*, Czech Republic, pp. 754-762, 2008.
- [18] Gutfinger D. and Sklansky J., "Robust Classifiers by Mixed Adaptation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 13, no. 6, pp. 552-567, 1991.
- [19] Holland J., *Adaptation in Natural and Artificial Systems*, University of Michigan Press, 1975.
- [20] Hsu C. and Lin C., "A Comparison of Methods for Multi-Class Support Vector Machines," *IEEE Transactions on Neural Networks*, vol. 13, no. 2, pp. 415-425, 2002.
- [21] Huber P., "Projection Pursuit," *Annals of Statistics*, vol. 13, no. 2, pp. 435-475, 1985.
- [22] Intrator N., "Combining Exploratory Projection Pursuit and Projection Pursuit Regression with Application to Neural Networks," *Journal of Neural Computation Archive*, vol. 5, no. 3, pp. 443-455, 1993.
- [23] Johnson R. and Wichern D., *Applied Multivariate Statistical Analysis*, Prentice-Hall, 2002.
- [24] Jones M. and Sibson R., "What is Projection Pursuit?," *Journal of the Royal Statistical Society*, vol. 150, no.1, pp. 1-37, 1987.
- [25] Kennedy J., Eberhart R., and Shi Y., *Swarm Intelligence*, Morgan Kaufman, 2001.
- [26] Lee E., Cook D., Klinke S., and Lumley T., "Projection Pursuit for Exploratory Supervised Classification," *Journal of Computational and Graphical Statistics*, vol. 14, no. 4, pp. 831-846, 2005.
- [27] Lee E. and Cook D., "A Projection Pursuit Index for Large P Small N Data," *Statistical Computing*, vol. 20, no. 3, pp. 318-392, 2010.
- [28] Lee Y., Cook D., Park J., and Lee E., "PPtree: Projection Pursuit Classification Tree," *Electronic Journal of Statistics*, vol. 7, pp. 1369-1386, 2013.
- [29] Lee Y. and Mangasarian O., "RSVM: Reduced Support Vector Machines," in *Proceedings of the 1<sup>st</sup> SIAM International Conference on Data Mining*, Chicago, pp. 1-17, 2001.
- [30] Marie-Sainte S., "Detection and Visualization of Non-Linear Structures in Large Datasets Using Exploratory Projection Pursuit Laboratory (EPP-Lab) Software," *Journal of King Saud University -Computer and Information Sciences*, vol. 29, no. 1, pp. 2-18, 2017.
- [31] Marie-Sainte S., Berro A., and Ruiz-Gazen A., "An Efficient Optimization Method for Revealing Local Optima of Projection Pursuit Indices," in *Proceedings of International Conference on Swarm Intelligence*, Brussels, pp. 60-71, 2010.
- [32] Mekhmoukh A. and Mokrani K., "MR Brain Image Segmentation Using an Improved Kernel Fuzzy Local Information C-Means Based Wavelet, Particle Swarm Optimization (PSO) Initialization and Outlier Rejection with Level Set Methods," *The International Arab Journal of Information Technology*, vol. 15, no. 4, pp. 683-692, 2018.
- [33] Morton S., "Interpretable Projection Pursuit," Ph.D. Dissertation, Stanford University, Stanford, 1989.
- [34] Moujahdi C., Ghouzali S., Mikram M., Wadood A., and RzizalM., "Inter-Communication Classification for Multi-View Face Recognition," *The International Arab Journal of Information Technology*, vol. 11, no. 4, pp. 387-395, 2014.
- [35] Parsopoulos K., *Particle Swarm Methods*, Springer International Publishing, 2015.
- [36] Pavlov D., Chudova D., and Smyth P., "Towards Scalable Support Vector Machines Using Squashing," in *Proceedings of the 6<sup>th</sup> ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Boston, pp. 295-299, 2000.
- [37] Pavlov D., Mao J., and Dom B., "Scaling-up Support Vector Machines Using Boosting Algorithm," in *Proceedings 15<sup>th</sup> International Conference on Pattern Recognition*, Barcelona, pp. 2219-2222, 2000.
- [38] Peña D. and Prieto F., "Cluster Identification Using Projections," *Journal of the American*

- Statistical Association*, vol. 96, no. 456, pp. 1433-1445, 2001.
- [39] Platt J., "Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines," Technical Report MSR-TR-98-14 Microsoft Research, MSR-TR-98-14, 1998.
- [40] Rechenberg I., *Evolutionsstrategie: Optimierung Technischer Systeme Nach Prinzipien Der Biologischen Evolution*, Frommann-Holzboog, Stuttgart, 1973.
- [41] Ren Y., Liua H., Yao X., and Liu M., "Prediction of Ozone Tropospheric Degradation Rate Constants by Projection Pursuit Regression," *Analytica Chimica Acta*, vol. 589, no. 1, pp. 150-158, 2007.
- [42] Schohn G. and Cohn D., "Less is more: Active Learning with Support Vector Machines," in *Proceedings of the 7<sup>th</sup> International Conference on Machine Learning*, San Francisco, pp. 839-846, 2000.
- [43] Schwefel H., *Numerical Optimization of Computer Models*, John Wiley and Sons, 1981.
- [44] Sun J., "Some Practical Aspects of Exploratory Projection Pursuit," *SIAM Journal on Scientific Computing*, vol. 14, no. 1, pp. 68-80, 1993.
- [45] Tong S. and Koller D., "Support Vector Machine Active Learning with Applications to Text Classification," in *Proceedings of the 17<sup>th</sup> International Conference on Machine Learning*, San Francisco, pp. 999-1006, 2000.
- [46] Tsang W., Kwok J., and Cheung P., "Very Large SVM Training Using Core Vector Machines," in *Proceedings of the 10<sup>th</sup> International Workshop on Artificial Intelligence and Statistics*, Barbados, 2005.
- [47] Vapnik V., *The Nature of Statistical Learning Theory*, Springer-Verlag, 1999.
- [48] Yamac M., "Can We Do Better By Combining Supervised and Non-Supervised Machine Learning for Pattern Analysis," Ph.D. Dissertation, Brown University, 1969.
- [49] Yu H., Yang J., and Han J., "Classifying Large Data Sets Using SVM with Hierarchical Clusters," in *Proceedings of the 9<sup>th</sup> ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Washington DC, pp. 306-315, 2003.
- [50] Zhu J., Li W., Li H., Wu Q., and Zhang L., "A Novel Swarm Intelligence Algorithm for the Evacuation Routing Optimization Problem," *The International Arab Journal of Information Technology*, vol. 14, no. 6, pp. 880-889, 2017.



**Sanaa Ghouzali** received both the Master's and the Ph.D. degrees in Computer Sciences from University Mohamed V-Agdal, Rabat, Morocco, in 2004 and 2009, respectively. In 2005 she has received a Fulbright grant as a visiting student at the Visual and Communication Laboratory of Cornell University, Ithaca, NY, USA. Between 2009 and 2011, she was an Assistant Professor at ENSA (the National school of Applied Sciences) with in Abdelmalek Essaadi University. Starting 2012, she joined King Saud University in the College of Computer and Information Sciences. Her research interests include statistical pattern detection and recognition, Biometrics, Biometric Security and Protection.



**Souad Larabi** is an assistant professor at the department of Computer Science at Prince Sultan University. She was an assistant professor at Information Technology department at King Saud University from 2012 to 2016. She worked as an associate researcher in the department of Computer Science at Toulouse1 Capitole University, France. She earned her PhD from same university in the area of bio-inspired algorithms in June 2011. She holds an M.S.c degree in Mathematics, Computing, Decision and Organization at Sorbonne Paris1 University, France. Souad is graduated from USTHB University in Algeria in the area of Operational Research. Her research interests include Combinatorial Optimization, Meta-heuristics, Artificial Intelligent, Bioinformatics, Data Mining and Biometric identification.