

Research Article

A New Evolutionary-Incremental Framework for Feature Selection

Mohamad-Hoseyn Sigari,^{1,2} Muhammad-Reza Pourshahabi,¹ and Hamid-Reza Pourreza¹

¹ Machine Vision Research Lab., Computer Engineering Department, Ferdowsi University of Mashhad, Mashhad 91779, Iran

² Control and Intelligent Processing Center of Excellence (CIPCE), School of Electrical and Computer Engineering, College of Engineering, University of Tehran, Tehran 14399, Iran

Correspondence should be addressed to Mohamad-Hoseyn Sigari; hoseyn_sigari@yahoo.com

Received 13 May 2014; Accepted 9 November 2014; Published 25 November 2014

Academic Editor: António Dourado Pereira Correia

Copyright © 2014 Mohamad-Hoseyn Sigari et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Feature selection is an NP-hard problem from the viewpoint of algorithm design and it is one of the main open problems in pattern recognition. In this paper, we propose a new evolutionary-incremental framework for feature selection. The proposed framework can be applied on an ordinary evolutionary algorithm (EA) such as genetic algorithm (GA) or invasive weed optimization (IWO). This framework proposes some generic modifications on ordinary EAs to be compatible with the variable length of solutions. In this framework, the solutions related to the primary generations have short length. Then, the length of solutions may be increased through generations gradually. In addition, our evolutionary-incremental framework deploys two new operators called *addition* and *deletion* operators which change the length of solutions randomly. For evaluation of the proposed framework, we use that for feature selection in the application of face recognition. In this regard, we applied our feature selection method on a robust face recognition algorithm which is based on the extraction of Gabor coefficients. Experimental results show that our proposed evolutionary-incremental framework can select a few number of features from existing thousands features efficiently. Comparison result of the proposed methods with the previous methods shows that our framework is comprehensive, robust, and well-defined to apply on many EAs for feature selection.

1. Introduction

In many pattern recognition problems, hundreds or maybe thousands of features are extracted from data to recognize the patterns in those data. Pattern recognition is a search process in the feature space; thus, when the number of extracted features is increased, computational complexity of search process is usually increased exponentially. Therefore, reduction of feature space is one of the main phases in many pattern recognition systems, especially for real-time systems.

One of the approaches for reduction of feature dimensions is PCA (principal component analysis) based methods. PCA tries to remove correlations between features and transfer data to a new uncorrelated space. The new space is built by a linear combination of original feature space and it is orthogonal. Although this approach can reduce dimension of feature spaces, this method does not preserve the essence of original feature spaces in the new space. In other words,

unlike the original feature space, the new feature space is usually not meaningful, because it is a linear combination of original features. In [1], a review and comparison of linear and nonlinear dimension reduction techniques is presented.

Another approach for reduction of dimensionality of feature space is feature selection. In this approach, redundant and not important features are eliminated from feature space; therefore, dimension of feature space is decreased. Feature selection is one of the most important and difficult problems in pattern recognition. The main goal of feature selection is selection of a feature subset with the minimum number of features which results in the maximum rate of accuracy of pattern recognition. The most important reasons for feature selection are as follows.

- (i) Increasing the number of features results in more computational complexity. In fact, pattern recognition is a search process in a feature space; therefore,

dimension of the search space will be increased when the number of features is increased.

- (ii) Learning is modeling of samples in a feature space. According to the minimum description length (MDL) principle [2], a simple model is better than a complex model. One of the methods for development of a simpler model is achieved by reducing the number of features.
- (iii) Some extracted features may be correlated together. Therefore, the computational complexity of the system is increased, but the system cannot extract new information from the correlated features.

Feature selection method is an NP-hard problem. Suppose F is a set with n elements including all possible features. If the goal of feature selection is selection of m features from F , the time complexity of the algorithm will be order of $O(n!/m!(n-m)!)$. n is usually much greater than m ; thus, the time complexity of the algorithm will be approximately equal to the order of $O(n^m/m!)$. Suppose F contains 1000 features and 10 features should be selected from F to achieve the highest accuracy. In this case, more than $2e + 23$ different cases should be evaluated in order to select 10 best features. Thus, finding the optimal feature subset using exhaustive search is very time consuming and it is impossible except for small feature sets. In the following, two main approaches for feature selection are considered: (1) sequential methods and (2) evolutionary methods.

One of the common and simplest methods for feature selection is incremental method. Incremental feature selection is divided into two types: forward selection and backward elimination [3]. Forward selection methods start with an empty set and add features until the maximum objective function is obtained. On the contrary, backward elimination methods start with a complete set of features and remove one feature at each step that removal of removed feature has the lowest effect on the accuracy of classification. In the literatures, incremental feature selection methods are called sequential methods too.

Incremental methods or modified version of such methods are useful for online feature selection on a stream of data [4]. However incremental feature selection methods are simple and useful for some special cases like online data streaming, but they are greedy and may fall into the local optimums. In other words, they assume the current selected features are absolutely optimal till now, and then try to add another feature to maximize objective function. Thus, they cannot take into account the high order dependency between the features completely and falls into local optimums. To overcome this issue, a probabilistic-incremental method was presented for feature selection in [5]. In this method, Las Vegas algorithm was used to select features probabilistically. Based on the reported results, this method can perform feature selection more efficiently. In this method, a consistency criterion was used to select features with the least consistency. The features are iteratively evaluated by the consistency criterion and a feature is added to the selected subset if it has minimum consistency with the selected features. This probabilistic-incremental method considers only some

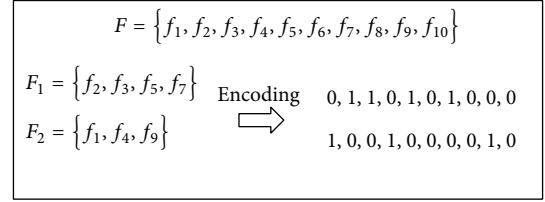


FIGURE 1: Chromosomes encoding in the most common form of GA for feature selection.

random selected features for consistency evaluation in each of the iteration.

Due to the high computational complexity of feature selection process, EAs are suitable approaches for solving such problem. EAs can find suboptimal solution in a limited period of time. One of the most popular and most widely used EAs is GA [6]. In the most common form of using GA for feature selection, chromosomes are defined as a bit string with the same length of F . In this case, existence of each feature is represented by a bit in the bit string. When a bit is 1, the corresponding feature is selected; otherwise, that feature is not selected. Figure 1 shows how a sample problem is encoded by a bit string in GA.

As depicted in Figure 1, possible solutions are displayed as a set of bit strings. Therefore, standard recombination and mutation operators can easily be used. Also, a criterion can be considered for fitness function which determines a higher fitness value for smaller subset of features with higher accuracy. For example, fitness function can be defined by a linear combination of the accuracy and length of the selected subset of features as follows:

$$\text{fitness}(F_i) = \alpha \text{Acc}(F_i) + (1 - \alpha) \text{length}(F_i). \quad (1)$$

In the above equation, F_i is a subset of the features, $\text{Acc}(F_i)$ is the accuracy of system using F_i , and $\text{length}(F_i)$ is the number of features in F_i . α is a number in the range $[0, 1]$. When α approaches to 1, although it may achieve the highest accuracy, the length of the final selected subset will not be optimal. Also, if α approaches to 0, the final selected subset has probably the shortest length, but the accuracy of the selected features is not so good. Therefore, α adjusts the balance between the accuracy and length of the selected features.

Multiobjective genetic algorithm (MOGA) [7] is another method for feature selection to achieve both the highest accuracy and the smallest subset of features. MOGA finds optimal or suboptimal solution in an optimization problem with two or more inconsistent objectives (criteria). Reaching the highest accuracy by the smallest subset of features is a multiobjective problem. Thus, some researches such as [8] tried to solve feature selection problem by MOGA.

Some of evolutionary methods like PSO (particle swarm intelligence) are essentially designed for continuous space. Thus, feature selection by such methods needs a few modifications. Liu et al. [9] used multiswarm PSO for feature selection. They modified ordinary PSO to prepare it for discrete problems; then, they selected best features using

a multiswarm mechanism. In their proposed method, each single swarm acts like a PSO and multiswarms controlled by heuristic rules.

Sigari and Lucas [10] investigated and compared three evolutionary algorithms including PSO, ICA, and IWO for feature selection in the application of face recognition. In this paper, face recognition was performed using coefficients extracted by Gabor wavelets. The authors tried to localize and optimize parameters of Gabor wavelets by evolutionary algorithms and thus the best features were extracted based on the best Gabor wavelets.

In addition to the traditional and standard EAs, many researches were performed for feature selection using modified EAs or combination of them. In [11], a new EA was proposed which is a combination of GA and immune clonal algorithm. In [12], an extended IWO algorithm for feature selection was proposed and used in the application of handwritten digit recognition and gate recognition. Extension of standard EAs or combination of them with other algorithms is usually performed to increase ability of exploration and exploitation or control diversity of population.

According to above brief review, incremental methods solve feature selection problem by greedy strategy. However these methods usually fall into local optimums, but greedy strategy helps us to solve problem in less time. In contrast, EAs may escape local optimums with higher probability, but they need more computing resources. In this paper, we propose an evolutionary-incremental framework for feature selection. The proposed framework combines greedy strategy of incremental methods with high explorative/exploitative abilities of EAs to achieve a good compromise between optimality and computational complexity. For experimental results, we apply our proposed framework on two EAs and we use them for feature selection in face recognition. The remain of the paper is organized as follows. Our proposed evolutionary-incremental framework is introduced in Section 2. Additionally, in Section 2, we explain how our proposed framework can be used in the form of two EAs: GA and IWO. In Section 2, the method of feature extraction for face recognition will be discussed. Experimental results and comparative results are presented in Section 3. Finally, conclusions and future works are presented in Section 4.

2. Materials and Methods

2.1. The Proposed Evolutionary-Incremental Framework for Feature Selection. In this paper, we propose a new framework that introduces some common modifications to change an ordinary EA and make a new extended version of that algorithm which is able to solve feature selection problem more efficiently. In fact, the architecture of our proposed framework is similar to the architecture of EAs, but some parts are modified. Also, new elements of the proposed framework are *addition* and *deletion* operators. Thus, we propose a generic structure for changing ordinary EAs and make them more suitable for feature selection.

In application of ordinary EAs such as GA and IWO for feature selection problem, solutions are represented by binary strings and thus different solutions have the same

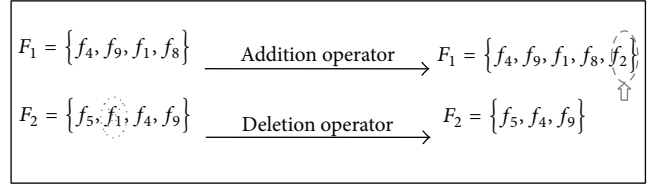


FIGURE 2: An example to show how addition and deletion operators act on a subset of features.

length. In the proposed framework, we initially start with the solutions of length 1 which is similar to the method based on probabilistic-incremental method [5]. Then, the maximum length of solutions is increased incrementally through generations. Thus, the proposed framework uses chromosomes with variable length. Unlike the probabilistic-incremental method, our proposed framework is based on EAs.

In the proposed framework, the first generation includes only solutions of length 1. Thus, in the first generation, the algorithm looks for the best single feature which results in the highest accuracy. After several generations, the solutions of length 2 are also generated using a new evolutionary operator. In the same way, maximum length of solutions is increased after several generations. We expect to find the best set of features according to the maximum length of solutions during several generations. In fact, we expect the EA to find the best set of features more efficiently when the maximum length of solutions is increased gradually.

Because the proposed framework works in the environment of variable length solutions, the selected EA for our framework should also be compatible to work with variable length solutions. Therefore, the proposed approach is an evolutionary-incremental framework which can be implemented using many EAs such as GA.

2.1.1. The New Evolutionary Operators for Variable Length Solutions. As mentioned before, the proposed evolutionary-incremental framework needs two new operators to change length of possible solutions. These operators are called addition and deletion. These operators are unary and act only on one individual of the population.

Addition operator adds a randomly new feature to a given solution (subset of feature) and deletion operator deletes randomly one feature from a given solution. According to the properties of the proposed framework, order of features in selected subset is not important. Therefore, addition operator can add a new feature to the end of the subset. The only point that should be considered in addition operation is that the new added feature must not already exist in the subset of features.

Figure 2 shows an example which explains how addition and deletion operators change a subset of features.

Implementation of the Proposed Framework Using Some EAs. In this section, we implement our proposed evolutionary-incremental framework using two EAs: GA and IWO. In fact, we have changed the structure of the standard algorithms

Input: P_r : recombination probability, P_m : mutation probability

- (i) Initialization: P_0 is a random initial population
- (ii) Evaluation: calculation of fitness value of each solution in the population
- (iii) Parent Selection: randomly selection of parents from the population by a given parent selection method (e.g., roulette wheel)
- (iv) Recombination: applying a recombination operator (e.g., 1-point crossover) on each pair of parents with the probability of P_r to generate new offspring
- (v) Mutation: applying a mutation operator (e.g., uniform mutation) on each offspring generated from the previous step with the probability of P_m
- (vi) Evaluation: calculation of fitness value of offspring
- (vii) Survival Selection: randomly selection of individuals from the current population and generated offspring by a given survival selection method (e.g., fitness based or age based)
- (viii) Termination Condition: if termination condition(s) is satisfied, the algorithm is finished; otherwise go to step (iii).
When the algorithm is finished, the best solution with the highest fitness value is the output of the algorithm.

ALGORITHM 1: Standard genetic algorithm.

Input: P_r : recombination probability, P_m : mutation probability, P_a : addition probability, P_d : deletion probability, L : maximum length of solutions, T : number of generations that L is constant during these generations

- (i) Initialization: $L = 1$, $t = T$ is an auxiliary variable, P_0 is a random initial population with the maximum length of solutions is equal to L
- (ii) Evaluation: calculation of fitness value of each solution in the population
- (iii) $t = t - 1$
- (iv) If $t < 0$ then $t = T$ and $L = L + 1$
- (v) Parent Selection: randomly selection of parents from the population by a given parent selection method (e.g., roulette wheel)
- (vi) Recombination: applying a recombination operator (e.g., 1-point crossover) on each pair of parents with the probability of P_r to generate new offspring
- (vii) Mutation: applying a mutation operator (e.g., uniform mutation) on each offspring generated from the previous step with the probability of P_m
- (viii) *Addition*: applying addition operator on each offspring generated from the previous step with the probability of P_a while the length of solutions must be less than or equal to L
- (ix) *Deletion*: applying deletion operator on each offspring generated from the previous step with the probability of P_d while the length of solutions must be greater than or equal to 1
- (x) Evaluation: calculation of fitness value of offspring
- (xi) Survival Selection: randomly selection of the individuals from the current population and generated offspring by a given survival selection method (e.g., fitness based or age based)
- (xii) Termination Condition: if termination condition(s) is satisfied, the algorithm is finished; otherwise go to step (iii).
When the algorithm is finished, the best solution with the highest fitness value is the output of the algorithm.

ALGORITHM 2: Evolutionary-incremental genetic algorithm.

of GA and IWO to achieve an evolutionary-incremental approach for feature selection.

2.1.2. The Evolutionary-Incremental Framework in the Form of GA. GA is one of the most popular evolutionary algorithms which have many applications in optimization problems. This algorithm was proposed by Holland in 1975 [6]. One of the applications of GA is in feature selection problem which was studied in introduction. In this section, we apply the proposed evolutionary-incremental framework to the standard form of GA.

The standard algorithm of GA is as in Algorithm 1. By applying the proposed evolutionary-incremental framework on GA, the algorithm will be changed as in Algorithm 2.

In some problems, final feasible solutions must have a minimum length. In this case, this condition has to be checked as a termination condition in the final step of the algorithm.

2.1.3. The Evolutionary-Incremental Framework in the Form of IWO Algorithm. One of the recently introduced EAs is IWO. This algorithm was introduced by Mehrabian and Lucas in 2006 [13]. Algorithm 3 presented an evolutionary optimization which is inspired by growth and proliferation of invasive weeds. In this algorithm, each possible solution in the search space is considered as a seed.

By applying the proposed evolutionary-incremental framework on IWO algorithm, the algorithm will be changed as in Algorithm 4.

2.2. Feature Selection for Face Recognition. Face recognition is one of the oldest problems in pattern recognition. In this section, we want to use the proposed evolutionary-incremental framework in the form of GA and IWO algorithms for feature selection in face recognition. The proposed method for face recognition is based on the method presented in [14].

Input: R_{\min} : minimum production rate, R_{\max} : maximum production rate, S_{\min} : minimum number of seeds, S_{\max} : maximum number of seeds

- (i) Initialization: a finite number of seeds are being dispread over the search space randomly, the number of initial seeds have to be fall in the range $[S_{\min}, S_{\max}]$
- (ii) Evaluation: calculation of fitness value of each seed
- (iii) Reproduction: current seeds grow to flowering and produce new seeds. Rate of production of each flower is dependent to the fitness of the corresponding seed. The production rate of seeds is linearly computed based on their fitness in the range $[R_{\min}, R_{\max}]$
- (iv) Spatial Dispersal: the produced seeds are being randomly dispread over the search space and grow
- (v) Competitive Exclusion: if the maximum number of plants is greater than S_{\max} , now only S_{\min} plants with the highest fitness values can survive and produce seeds. Other plants are eliminated
- (vi) Termination Condition: if termination condition(s) is satisfied, the algorithm is finished; otherwise go to step (iii).
When the algorithm is finished, the best solution with the highest fitness value is the output of the algorithm.

ALGORITHM 3: Standard invasive weed optimization.

Input: R_{\min} : minimum production rate, R_{\max} : maximum production rate, S_{\min} : minimum number of seeds, S_{\max} : maximum number of seeds, P_a : addition probability, P_d : deletion probability, L : maximum length of solutions, T : number of generations that L is constant during these generations

- (i) Initialization: a finite number of seeds are being dispread over the search space randomly, the number of initial seeds have to be fall in the range $[S_{\min}, S_{\max}]$
- (ii) Evaluation: calculation of fitness value of each seed
- (iii) $t = t - 1$
- (iv) If $t < 0$ then $t = T$ and $L = L + 1$
- (v) Reproduction: current seeds grow to flowering and produce new seeds. Rate of production of each flower is dependent to the fitness of the corresponding seed. The production rate of seeds is linearly computed based on their fitness in the range $[R_{\min}, R_{\max}]$
- (vi) *Addition*: applying addition operator on each seed produced from the previous step with the probability of P_a while the length of solutions must be less than or equal to L
- (vii) *Deletion*: applying deletion operator on each seed produced from the previous step with the probability of P_d while the length of solutions must be greater than or equal to 1
- (viii) Spatial Dispersal: the produced seeds are being randomly dispread over the search space and grow
- (ix) Competitive Exclusion: if the maximum number of plants is greater than S_{\max} , now only S_{\min} plants with the highest fitness values can survive and produce seeds. Other plants are eliminated
- (x) Termination Condition: if termination condition(s) is satisfied, the algorithm is finished; otherwise go to step (iii).
When the algorithm is finished, the best solution with the highest fitness value is the output of the algorithm.

ALGORITHM 4: Evolutionary-incremental invasive weed optimization.

In [14], a face is represented by elastic labeled graph that each node of graph contains a set of features. The set of features at each node of graph is extracted using 2D Gabor wavelet transforms. The method presented in that paper is called Elastic Bunch Graph Matching (EBGM), because it uses a graph matching method for face recognition. In that method, wavelength and rotation of Gabor wavelets are from a given sets determined by user, but location of each node changes on the face image of different individuals. According to this method, total number of features extracted from each face is 1200.

In [15], the author has tried to modify the method presented in [14]. In this paper, GA was used to determine the best set of wavelength for Gabor wavelets. This method extracts a feature vector of length 672 for each face. Therefore, this method is faster than EBGM.

In this paper, we want to modify EBGM method by applying a feature selection method. In EBGM method,

although the features are extracted from flexible locations, a set of certain Gabor wavelets are used for feature extraction at each node of elastic graph. In the proposed method, we try to find the best locations and the best Gabor wavelets for each node while the locations of nodes are fixed for all faces. In other words, the proposed method extracts only one feature instead of a set of features for each node, but it uses a fixed graph instead of elastic graph for face modeling. Although we used a fixed graph for all faces, we tried to construct the best graph. Thus, we are trying to select the best features at the best location of the face images to reduce the number of extracted features.

2.2.1. Gabor Wavelet. 2D Gabor wavelet is the result of multiplication of a 2D sinusoid function by a 2D Gaussian function. The sinusoid function extracts frequency information corresponding to its frequency and the Gaussian function determines the region of effects of the sinusoid

function. Therefore, Gabor wavelet operates like a local edge detector. A 2D Gabor wavelet is formulated as follows:

$$G(x, y) = \cos\left(\frac{2\pi X}{\lambda}\right) \exp\left(-\frac{(X - \mu_x)^2 + (Y - \mu_y)^2}{2\sigma^2}\right), \quad (2)$$

where

$$\begin{aligned} X &= x \cos \theta - y \sin \theta, \\ Y &= x \sin \theta + y \cos \theta. \end{aligned} \quad (3)$$

According to the above equation, there are five parameters for Gabor wavelet: μ_x , μ_y , θ , λ , and σ . μ_x and μ_y are the center coordinates of the Gabor wavelet. θ is the rotation angle of the wavelet which indicates the direction of the wavelet. λ is the wavelength of the sinusoid function and σ is the radius (standard deviation) of the Gaussian function. In Gabor wavelet, the radius of Gaussian function is usually considered as a coefficient of the wavelength of the sinusoid function. Thus, there are only four independent parameters for Gabor wavelet. By applying the 2D Gabor wavelets on a certain location of image (μ_x, μ_y) , a wavelet coefficient is computed.

2.2.2. Feature Extraction. Our goal of feature selection in face recognition is selection of a limited number of Gabor wavelets to achieve the maximum accuracy. The method used in this paper is similar to EBGm method [14]. The main differences between the proposed method and EBGm method are as follows.

- (i) In EBGm, the locations used to extract Gabor features are flexible for different faces, but we use a set of fixed locations to extract features for all faces.
- (ii) In EBGm, the locations used to extract Gabor features are selected by a matching method for each face, but we select the best locations to extract features using an EA.
- (iii) In EBGm, a certain set of Gabor coefficients are extracted at each location, but we extract only one Gabor coefficient at each location. The parameters of Gabor wavelets which are applied at each location are determined by an EA.

As mentioned in the previous section, a Gabor wavelet has 4 parameters: μ_x , μ_y , θ , and λ . If the facial image has a resolution of 50×50 pixels and the number of possible values for θ and λ are 8 and 10, respectively, 200000 different Gabor wavelets exist to extract 200000 different features from each image. Although some of these features are correlated with each other, feature selection from such a large collection of features is very difficult.

3. Results and Discussion

For experimental results, 200 face classes from FERET database were used. 100 face classes were used as set A for

feature selection and 100 face classes were used as set B for evaluation of the selected features. Each face class includes two frontal face images with 56×48 pixel size. These facial faces are normalized in size, rotation, and eye location.

In the experiments, the accuracy of face recognition using the selected features is considered as the fitness function of each possible solution. Therefore, it is necessary to train and evaluate the face recognition system based on the selected features. For this purpose, we used set A: one image of each class for training and one image for test. Therefore, for a subset of features, the fitness value is equal to the accuracy obtained using those features on set A.

When the EA finishes, the best features are selected according to the best answer in the final generation. To determine the quality of this feature set, it is evaluated on a new dataset (set B). The accuracy of face recognition on set B determines whether the final selected feature is over-fitted or not. In other words, if the final selected features result in higher accuracy on set B, the selected features are nearer to the optimal features.

All experiments have been performed using MATLAB R2008a, on a personal computer with an AMD 5600+ Dual Core CPU with 2 GB of memory.

As mentioned before, the extracted features are based on Gabor wavelet. Each Gabor wavelet has four parameters. In addition to the Gabor wavelet parameters, a parameter called weight coefficient (w) is determined for each selected feature. The weight coefficient is a real number in the range $(0, 1]$ and is used for distance calculation in the nearest neighbor classifier. Thus, there are five parameters for each feature which are listed as follows.

μ_x, μ_y : the center coordinates of the Gabor wavelet that $\mu_x \in [1, 48]$ and $\mu_y \in [1, 56]$.

θ : the rotation angle of the Gabor wavelet that $\theta \in \{0, \pi/8, \pi/4, 3\pi/8, \pi/2, 5\pi/8, 3\pi/4, 7\pi/8\}$.

λ : the wavelength of the Gabor wavelet that $\lambda \in \{1, 2, \dots, 12\}$.

w : the weight coefficient assigned to the Gabor wavelet that $w \in (0, 1]$.

The values which are selected for θ and λ are similar to the suggested values in [14, 15].

According to the above explanations, each feature of our proposed system has 5 parameters. Number of features is determined by the EA. Each generation (epoch) contains several solutions and each solution shows a few features and determines the parameters of the features. According to the features determined by each solution, features are extracted from all images in set A. Then, face recognition is performed using weighted nearest neighbor. In the classification phase which is based on weighted nearest neighbor, one image of each class is used for training and another image is used for test. Thus, the fitness value of a solution is determined as the average accuracy of face recognition based on the features determined by that solution.

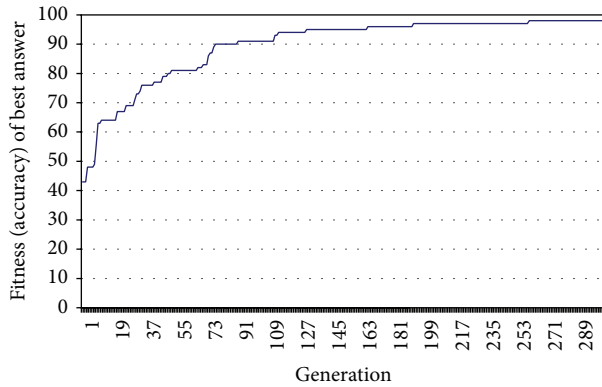


FIGURE 3: Changes of fitness value of the best solution through generations using the proposed evolutionary-incremental GA.

3.1. *Feature Selection Using Evolutionary-Incremental Framework in the Form of GA.* The parameters of the evolutionary-incremental framework and GA are as follows:

- (i) $P_r = 0.2$,
- (ii) $P_m = 0.1$ (for each allele),
- (iii) $P_a = 0.6$,
- (iv) $P_a = 0.4$,
- (v) $T = 5$,
- (vi) $L = 1$,
- (vii) population size = 100,
- (viii) elitism is enabled.

In this algorithm, parent selection is performed using tournament selection (with parameter 2). In our experiments, all of the chromosomes of the current population are replaced by new offspring. In addition, because elitism is enabled in our algorithm, the best solution of the current population is remained in the new generation while the worst solution from the generated offspring will be deleted.

Figure 3, shows the changes of fitness value of the best solution through generations. According to this figure, the increase of the fitness value is very similar to an exponential function in which the slope of the diagram is gradually decreased.

Figure 4 shows number of selected features determined by the best solution in each generation. It is observed that the number of features is increased almost linearly. However, the number of selected features in some generations is less than its previous generation.

In order to evaluate the quality of the final selected features, after finishing the feature selection by the evolutionary-incremental algorithm, the best selected features in the last generation are used for face recognition on set B. In set B, there are two images for each class: one image is for training and another image for test. Experimental results showed that the accuracy of face recognition using the selected features on set B is 92%. This result shows that the features selected by our proposed evolutionary-incremental framework in the form

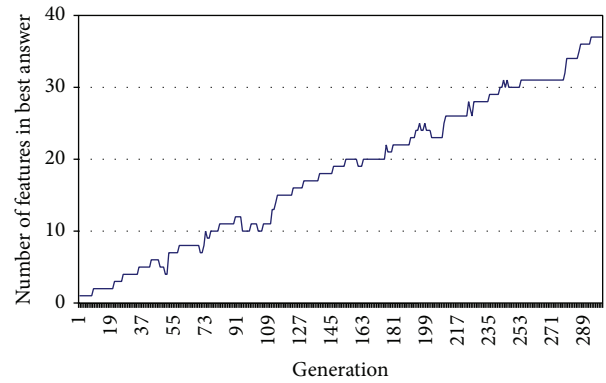


FIGURE 4: Number of selected features determined by the best solution through generations using the proposed evolutionary-incremental GA.

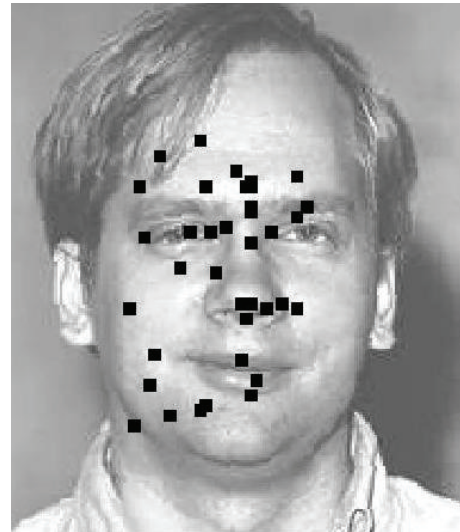


FIGURE 5: Center of the Gabor wavelets on a sample face which are selected by the proposed evolutionary-incremental GA.

TABLE 1: Specification of the best selected features using the proposed evolutionary-incremental GA.

Number of selected features	40
Accuracy of face recognition on set A	98%
Accuracy of face recognition on set B	92%

of GA are efficiently selected and the selected features can be generalized for face recognition on other face databases.

The specification of the final best selected features using our proposed method is shown in Table 1. In this table, the number of selected features and the accuracy of face recognition on sets A and B using the best selected feature are shown. Figure 5 shows the center of the Gabor wavelets on a sample face which are selected by our proposed evolutionary-incremental framework in the form of GA.

This experiment lasted about 30 hours. Due to the short length of the solutions in the primary generations, speed of fitness calculation is high and therefore each generation ends

in a short period. During maturity of solutions in generations, the length of solutions is increased and therefore speed of fitness calculation will decreased gradually. Consequently the time required for ending each of the generation will be longer.

3.2. Feature Selection Using Variable Length GA. One of the methods for representation of a subset of features is based on fixed-length bit string which was described in Section 1. This method may be replaced by variable length chromosomes. This method is similar to the proposed evolutionary-incremental framework except the fact that no restrictions exist for the length of the solutions in each generation. In this method, addition and deletion operators exist to change the length of solutions. This type of GA is called variable length GA.

In another experiment, feature selection for face recognition was performed using variable length GA. In this algorithm, the following parameters were used:

- (i) $P_r = 0.2$,
- (ii) $P_m = 0.1$ (for each allele),
- (iii) $P_a = 0.5$,
- (iv) $P_d = 0.5$,
- (v) max length of solutions = 100,
- (vi) population size = 100,
- (vii) elitism is enabled.

The parameters used in this experiment are almost the same as the parameters used in the previous experiments. In this algorithm, parent selection is performed using tournament selection (with parameter 2). In our experiments, all of the chromosomes of the current population are replaced by new offspring, while the best solution of the last generation is remained (elitism).

Because in variable length GA, fitness calculation of very long solutions is very time consuming, we have to restrict the length of solutions. For this reason, the maximum allowed length of solution is assumed to be 100. In the proposed variable length GA for feature selection, fitness value of each solution is calculated from (1) with $\alpha = 0.99$. The value selected for α allows us to reach the highest accuracy as the main goal while we probably achieve the minimum length of solution as the second goal.

Figure 6 shows the changes of fitness value of the best solution in each generation. The fitness value of the best solution in the first generation is near to 82% which is very good as the best solution in the first generation. Over successive generations, the speed of increasing fitness value is very slow, and after 65 generations, the change of fitness value is zero.

Figure 7 shows the number of selected features in the best solution of each generation. Based on the results depicted in this figure, the number of selected features in almost all of the generations is 77 or more. In the primary generations (generations 1 to 5), the number of selected features was decreased suddenly from 94 to 77. Then, the number of selected features was increased suddenly from 77 to 100

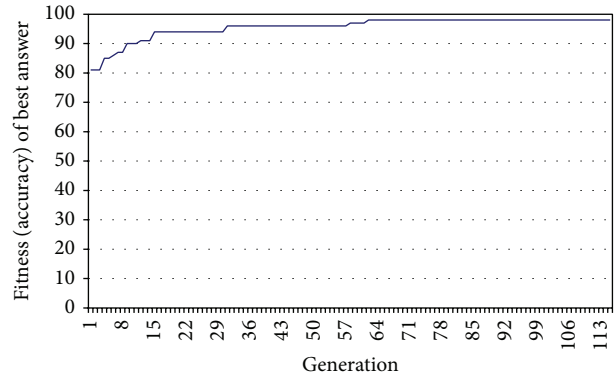


FIGURE 6: Changes of fitness value of the best solution through generations for variable length GA.

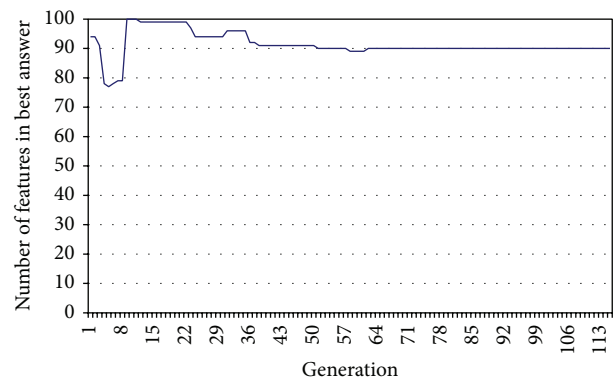


FIGURE 7: Number of selected features determined by the best solution through generations for variable length GA.

TABLE 2: Specification of the best selected features using variable length GA.

Number of selected features	90
Accuracy of face recognition on set A	98%
Accuracy of face recognition on set B	89%

during generations 6 to 10. After generation 11, the number of selected features was decreased gradually.

This experiment shows that because the longer solutions result in higher fitness value (accuracy), the probability of reproducing and remaining of short length solutions in successive generations is very low. Therefore, variable length GA does not permit short solutions to be matured and reproduce offspring while α is near to 1. On the other hand, when α is reduced, the probability of reproducing and remaining of short solutions is increased, but the convergence time of the algorithm will also be increased.

In order to evaluate the quality of feature selection algorithm, the best selected features in the last generation was used for face recognition process on set B. Experiments showed that the accuracy of face recognition using the final selected features on set B is 89%.

The specification of the final best selected features using variable length GA is shown in Table 2. In this table, the number of selected features and the accuracy of face recognition



FIGURE 8: Center of the Gabor wavelets on a sample face which are selected by variable length GA.

on sets A and B using the best selected feature are shown. Figure 8 shows the center of the Gabor wavelets on a sample face which are selected by variable length GA.

This experiment lasted about 43 hours. Although the number of generations is 120 (which is about 40% of the number of generations used in the evolutionary-incremental GA), but due to the existence of long solutions in the populations, each generation takes a lot of time. In other words, the solutions in different generations of the algorithm are usually long; therefore, the number of selected features is more and the time required for calculating the fitness value is longer.

3.3. Feature Selection Using Evolutionary-Incremental Framework in the Form of IWO Algorithm. For selecting features using evolutionary-incremental framework in the form of IWO the following parameters were used:

- (i) $R_{\min} = 0.05$,
- (ii) $R_{\max} = 0.5$,
- (iii) $S_{\min} = 50$,
- (iv) $S_{\max} = 150$,
- (v) $P_a = 0.6$,
- (vi) $P_a = 0.4$,
- (vii) $T = 10$,
- (viii) $L = 1$,
- (ix) elitism is enabled.

The parameters used for evolutionary-incremental IWO are almost the same as the parameters used in the previous experiments. In IWO, we set S_{\min} and S_{\max} to 50 and 150, respectively, while the average of these values is equal to 100 (the population size of each generation in GA-based algorithm).

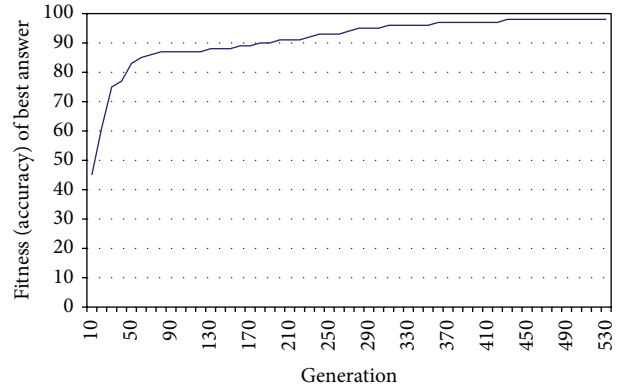


FIGURE 9: Changes of fitness value of the best solution through generations for evolutionary-incremental IWO.

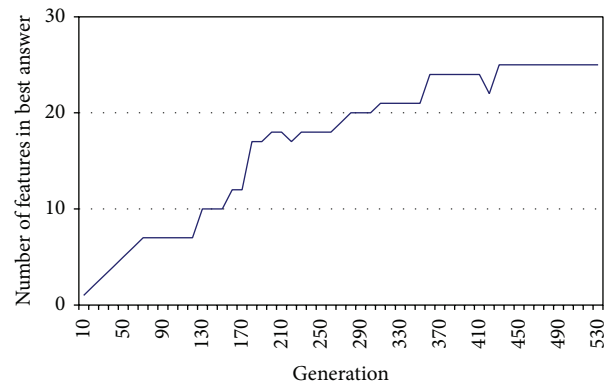


FIGURE 10: Number of selected features determined by the best solution through generations for variable length GA.

Figure 9 shows the changes of fitness value of the best solution in each generation. As can be seen, the changes of fitness value are very similar to an exponential function, in which the slope of the changes is high at first and gradually decreases. The changes of the best fitness value ultimately reaches zero.

Figure 10 shows the number of selected features in the best solution in each generation. The number of selected features changes in a nonuniformly ascending manner.

In order to evaluate the quality of the final selected features, the best selected features in the last generation was used for face recognition process on set B. The experiments showed that the accuracy of face recognition using these selected features is 93% on set B. This result shows that the selected features are efficiently selected. In other words, our proposed evolutionary-incremental IWO can select the best features with minimum overfitting on training data.

The specification of the best solution in the last generation of the proposed evolutionary-incremental IWO-based feature selection process is shown in Table 3. Also, Figure 11 shows the location of the final selected features on a sample face image.

This experiment lasted about 28 hours. In IWO algorithm, size of population differs through different generation;

TABLE 3: Specification of the best selected features using evolutionary-incremental IWO.

Number of selected features	25
Accuracy of face recognition on set A	98%
Accuracy of face recognition on set B	93%

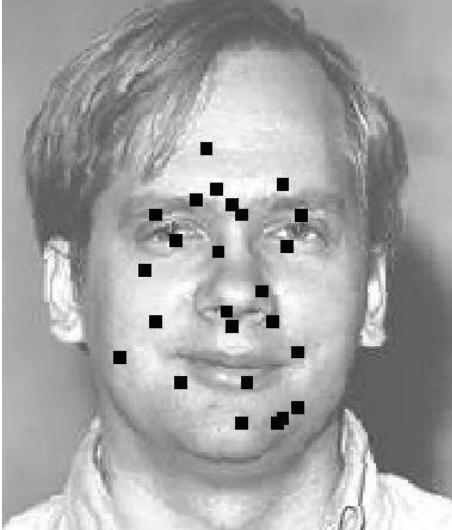


FIGURE 11: Center of the Gabor wavelets on a sample face which are selected by evolutionary-incremental IWO.

therefore, the time spent for a generation is variable. In the primary generations, solutions are short and the time required for calculation of fitness is short. By passing primary generations, longer solutions are reproduced and calculation of fitness value will be increased.

3.4. Comparison Results. In this section, we compare the results of feature selection methods which were introduced in the previous sections. In addition, the results of face recognition using the features selected by feature selection methods are compared with the results of other face recognition methods including Elastic Bunch Graph Matching (EBGM) [14] and its modified version proposed by Sigari [15].

3.4.1. Comparison of the Evolutionary-Based Feature Selection Methods for Face Recognition. In this paper, three methods were presented and evaluated for feature selection based on the proposed framework. According to the experimental results, we can compare these feature selection methods briefly in Table 4.

According to the results shown in Table 4, the evolutionary-incremental IWO algorithm has the best specifications. This algorithm could achieve the highest accuracy on set B while it selected the minimum number of features. Additionally, the time required for evolution (selecting the best features on set A) is the shortest one in comparison with the other methods. The evolutionary-incremental GA could also reach the similar accuracy on set B by more number of features. Also, the time required for evolution (selecting

the best features on set A) is longer than evolutionary-incremental IWO. Among all methods evaluated in this paper, variable length GA is the worst method in both number of selected features and the accuracy of face recognition on set B.

3.4.2. Comparison of the Proposed Methods and the Previous Methods for Face Recognition. Elastic Bunch Graph Matching (EBGM) method was presented by [14]. In this method, at first 30 points (called jet points) on the face image have been determined using an elastic bunch graph. Then, Gabor wavelet coefficients in 5 wavelengths (λ) and 8 different directions (θ) are calculated for each of these points. Thus, $30 \times 5 \times 8 = 1200$ coefficients are extracted for each face image totally. Afterward, face recognition is performed using a new proposed similarity method. This algorithm is evaluated on frontal images of FERET database and the accuracy of 98% has been reported.

Sigari [15] has modified EBGM algorithm and used GA for selection of the best wavelength of Gabor wavelets. In this algorithm, the number of points on the face used for feature selection is decreased to 14 points. In this method, 8 different directions (θ) were predefined for Gabor wavelet, but 6 wavelengths were selected as the best wavelengths (λ) for face recognition. According to this setting, $14 \times 6 \times 8 = 672$ features were extracted for each face. This algorithm is evaluated on frontal images of FERET database and the accuracy of 91% was achieved.

Sigari and Lucas [10] investigated three EAs including PSO, ICA, and IWO for feature selection in the application of face recognition. These EAs were used to look for the best 2D Gabor wavelets those extract best features from facial images. Parameters of Gabor wavelets optimized by EAs are center of the Gabor wavelet in 2D space (μ_x, μ_y), rotation angle (θ), and wavelength (λ). In this research, authors assumed number of the best feature is fixed (40 features) and the objective function was defined as the accuracy of face recognition.

Table 5 compares the specifications of 5 different methods of face recognition. All of these methods have been tested on frontal images of FERET database in equal situations.

According to Table 5, the number of selected features using the proposed EAs is much less than the number of features selected by [10, 14, 15]. Although the basic algorithm of EBGM has the highest accuracy for face recognition, it has to extract 1200 features and therefore it takes much time. It seems that the proposed method which selects the best 25 features by evolutionary-incremental IWO achieves a good balance between number of features (computational complexity) and accuracy. In this case, we recognize faces only by about 2% of total number of features which are used by basic EBGM, while we achieve 93% of the accuracy.

4. Conclusions

In this paper, we proposed a new evolutionary-incremental framework for feature selection which can be used in the form of many EAs such as GA and IWO. Experiments were performed for feature selection in face recognition problem.

TABLE 4: Comparison of the proposed evolutionary based feature selection algorithms for face recognition.

Method	Number of generations	Number of selected features	Accuracy of face recognition on set A (training)	Accuracy of face recognition on set A (test)	Time
Variable length GA	115	90	98%	89%	43 hours
Evolutionary-incremental GA	300	37	98%	92%	30 hours
Evolutionary-incremental IWO	530	25	98%	93%	28 hours

TABLE 5: Comparison of the proposed methods and the previous methods for face recognition.

Method	Number of features	Accuracy of face recognition
Basic EBGM [14]	1200	98%
Combination of EBGM and GA [15]	672	91%
Combination of EBGM and PSO [10]	40	81%
Combination of EBGM and ICA [10]	40	86%
Combination of EBGM and IWO [10]	40	87%
Combination of EBGM and variable length GA	90	89%
Combination of EBGM and evolutionary-incremental GA	37	92%
Combination of EBGM and evolutionary-incremental IWO	25	93%

According to the obtained results, we can use only a few features selected by the proposed method instead of hundreds of features while it causes acceptable and low error rate for face recognition. Experiments showed that the evolutionary-incremental EAs are more efficient than ordinary EAs to select the best features for face recognition.

It seems that the evolutionary-incremental IWO is the most appropriate method for feature selection for face recognition. This algorithm takes shorter time to find the suboptimal subset of feature with respect to other algorithms. In addition, evolutionary-incremental IWO finds better subset of features which leads to higher recognition rate on test dataset (set B).

Evolutionary-incremental GA has also similar accuracy but it selects more number of features. Experimental results show that variable length GA has higher computational complexity for finding the optimal features. Additionally, variable length GA selects a long feature set and the selected features lead to weaker results in face recognition. Thus, it seems that the proposed evolutionary-incremental framework for feature selection is appropriate and efficient. From the viewpoint of computational complexity and the number of selected features, the proposed evolutionary-incremental framework is better than the conventional EAs.

In the experiments, the relative efficiency of the proposed evolutionary-incremental framework for feature selection was evaluated. The proposed framework can also be used for other optimization problems such as clustering which has variable length solutions [16, 17].

Conflict of Interests

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

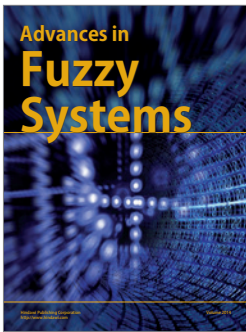
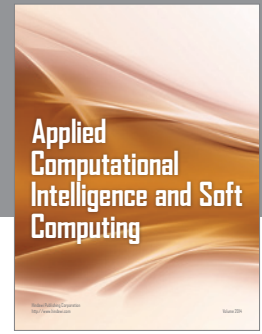
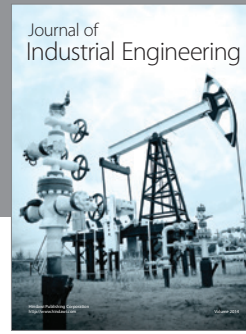
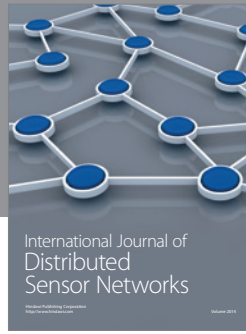
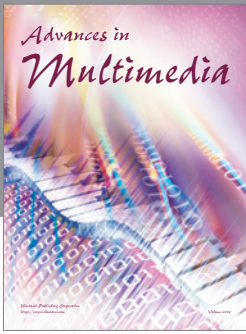
Acknowledgment

The first author would like to thank and remember Professor Caro Lucas (1949–2010), late faculty member of the University of Tehran, who motivated them to start this research.

References

- [1] L. van der Maaten, E. Postma, and J. van den Herik, *Dimensionality Reduction: A Comparative Review*, Tilburg University, 2009.
- [2] J. Rissanen, "A universal prior for integers and estimation by minimum description length," *The Annals of Statistics*, vol. 11, no. 2, pp. 416–431, 1983.
- [3] G. Chandrashekar and F. Sahin, "A survey on feature selection methods," *Computers & Electrical Engineering*, vol. 40, no. 1, pp. 16–28, 2014.
- [4] X. Wu, K. Yu, W. Ding, H. Wang, and X. Zhu, "Online feature selection with streaming features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 5, pp. 1178–1192, 2013.
- [5] H. Liu and R. Setiono, "Incremental feature selection," *Applied Intelligence*, vol. 9, no. 3, pp. 217–230, 1998.
- [6] J. H. Holland, *Adaption in Natural and Artificial Systems*, The University of Michigan Press, Ann Arbor, Mich, USA, 1975.
- [7] T. Murata and H. Ishibuchi, "MOGA: multi-objective genetic algorithms," in *Proceedings of the IEEE International Conference on Evolutionary Computation*, pp. 289–294, Perth, Australia, December 1995.
- [8] M. Morita, R. Sabourin, F. Bortolozzi, and C. Y. Suen, "Unsupervised feature selection using multi-objective genetic algorithms for handwritten word recognition," in *Proceedings of the International Conference on Document Analysis and Recognition*, pp. 666–670, August 2003.
- [9] Y. Liu, G. Wang, H. Chen, H. Dong, X. Zhu, and S. Wang, "An improved particle swarm optimization for feature selection," *Journal of Bionic Engineering*, vol. 8, no. 2, pp. 191–200, 2011.
- [10] M. H. Sigari and C. Lucas, "Near optimal feature selection using evolutionary algorithms for face recognition: PSO (particle

- swarm intelligence), ICA (imperialist competitive algorithm) and IWO (invasive weed optimization),” in *Proceedings of the 19th Iranian Conference on Electrical Engineering*, Tehran, Iran, 2011.
- [11] S. Wu, Y. Hu, W. Wang, X. Feng, and W. Shu, “Application of global optimization methods for feature selection and machine learning,” *Mathematical Problems in Engineering*, vol. 2013, Article ID 241517, 8 pages, 2013.
- [12] H. Josiński, D. Kostrzewa, A. Michalczuk, and A. Świtoński, “The expanded invasive weed optimization metaheuristic for solving continuous and discrete optimization problems,” *The Scientific World Journal*, vol. 2014, Article ID 831691, 14 pages, 2014.
- [13] A. R. Mehrabian and C. Lucas, “A novel numerical optimization algorithm inspired from weed colonization,” *Ecological Informatics*, vol. 1, no. 4, pp. 355–366, 2006.
- [14] L. Wiskott, J. M. Fellous, N. Krüger, and C. von der Malsburg, “Face recognition by elastic bunch graph matching,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 7, pp. 775–779, 1997.
- [15] M. H. Sigari, “Best wavelength selection for gabor wavelet using GA for EBGM algorithm,” in *Proceedings of the International Conference on Machine Vision (ICMV '07)*, pp. 35–39, Islamabad, Pakistan, December 2007.
- [16] K. Venkatesh, S. C. Satapathy, J. V. R. Murthy, and P. V. G. D. Prasad Reddy, “Hybridized improved genetic algorithm with variable length chromosome for image clustering,” *International Journal of Computer Science and Network Security*, vol. 7, no. 11, pp. 121–131, 2007.
- [17] S. Vijendra and S. Laxman, “Subspace clustering of high-dimensional data: an evolutionary approach,” *Applied Computational Intelligence and Soft Computing*, vol. 2013, Article ID 863146, 12 pages, 2013.



Hindawi

Submit your manuscripts at
<http://www.hindawi.com>

