Neural-Network-Based Gender Classification
Using Genetic Search for Eigen-Feature Selection

Zehang Sun, Xiaojing Yuan, and George Bebis
Computer Vision Laboratory
Department of Computer Science
University of Nevada, Reno
{zehang,xjyuan,bebis}@cs.unr.edu

Sushil J. Louis
Genetic Adaptive Systems Lab
Department of Computer Science
University of Nevada, Reno
sushil@cs.unr.edu

***Abstract* - We consider the problem of gender classification from frontal facial images using feature selection and neural networks. We argue that feature selection is an important issue in gender classification and we demonstrate that by removing features that do not encode important gender information from the image representation of faces, the error rate can be reduced significantly. Automatic feature subset selection distinguishes the proposed method from previous gender classification approaches. First, Principal Component Analysis (PCA) is used to represent each image as a feature vector (i.e., eigen-features) in a low-dimensional space, spanned by the eigenvectors of the covariance matrix of the training images (i.e., coefficients of the linear expansion). A Genetic Algorithm (GA) is then used to select a subset of features from the low-dimensional representation by removing certain eigenvectors that do not seem to encode important information about gender (e.g., eigenvectors encoding information about glasses). Finally, a Neural Network (NN) is trained to perform gender classification using the selected eigen-feature subset. Experimental results demonstrate a significant improvement in error rate reduction. Using a subset of eigen-features containing only 18% of the features in the complete set, the average NN classification error goes down to 11.3% from an average error rate of 17.7%.**

1. INTRODUCTION

Although several gender classification methods have been reported in the literature, gender classification has attracted less attention compared to other research topics in computer vision. Successful gender classification could be used to boost the performance of face recognition systems. Golomb et. al [4] used a neural network based method to classify gender using facial images. Their database contains 90 images—half male and half female. They referred to their 2-layer fully connected neural network as SEXNET. The accuracy reported in that work was 91.8%. Brunelli et. al [5] developed a gender classifier using Hyper basis function, where two competing RBF networks (HyperBF), one for male and the other for female, were trained to do the gender classification. HyperBF networks used a vector of geometrical features as the input to the network. The accuracy reported in that work was 79%. Burton et. al [6] constructed a discriminant function using 2-D and 3-D face measurement and achieved 85.5% accuracy over their 179 faces using 12 measurements. Gray et al [7] used a simple perceptron and reported an accuracy of 81%. Lately, Moghaddam et al [8] have

investigated gender classification using Support Vector Machines (SVMs). They compared several techniques (LDA, Nearest-Neighbour, and RBF) with SVMs using 1755 "thumbnail" face images from the FERET database. They reported an accuracy of 96.6%.

Many factors affect the success of classification on a given task. Among them, the issue of which features to extract from the data is first and foremost. Feature subset selection is the process of identifying and removing as much irrelevant and redundant information as possible. This may allow learning algorithms to operate faster and more effectively. A number of approaches for feature subset selection have been proposed in the literature. Hall [1] uses a correlation-based method to find and discard redundant features. Koller et al [2] used a greedy algorithm to remove the features that provide the least additional information given the remaining features. Brill et. al [3] have explored randomized population-based heuristic search approaches such as GAs to select feature subsets for NNs.

Almost every gender classification method reported in the literature uses the complete set of features extracted from frontal images or uses the raw image directly. One exception is HyperBF [7], where a set of predetermined geometrical measures is chosen manually. Obviously, frontal images contain lots of information, such as age, race, gender. If the objective is to perform gender classification, then information from unrelated sources might confuse the classifier, especially when available data is limited (due to the *curse of dimensionality*).

Automatic feature subset selection distinguishes our proposed gender classification method from other reported approaches. Facial images are represented in a low-dimensional space, spanned by the eigenvectors of the covariance matrix, computed by using PCA [9]. A GA is used to select gender-related features automatically and improve the performance of the gender classifier that is implemented using a NN. In particular, the GA searches the eigenspace to find eigenvectors encoding mostly gender information. The NN classifier [10] is then used to evaluate the selected feature subset by performing the final gender classification. Experimental results and comparisons demonstrate a significant improvement in the performance of the NN gender classifier by using the selected feature subset instead of the complete set. In particular, the average NN error rate goes down to 11.3%

from an average error rate of 17.7%, while the number of selected eigen-features is only 18% of the complete eigen-feature set. Comparing the NN classifier with the Bayes classifier, the NN classifier shows better performance.

Our work is related to [13]-[15] where NNs were used with PCA features for gender classification. Although the emphasis in those studies was on understanding how humans perform gender classification, an important issue was identifying which eigen-features were most important for gender classification. The proposed genetic search approach for eigen-feature selection can provide us with useful insights regarding this issue and we further discuss this in sections 3 and 7.

The rest of the paper is organized as follows: In Section 2, we present a brief overview of eigenspace representations. Section 3 presents a discussion on feature selection in the context of gender classification. In section 4, we present the genetic search approach for eigen-feature selection. Our datasets and preprocessing are discussed in section 5. Experimental results and comparisons are presented in section 6. Section 7 discusses our results and section 8 concludes the paper and presents possible directions for future work.

2. EIGENSPACE REPRESENTATIONS

Eigenspace representations of images use PCA [9] to linearly project an image in a low-dimensional space. This space is spanned by the principal components (i.e., eigenvectors corresponding to the largest eigenvalues) of the distribution of the training images. After an image has been projected in the eigenspace, a feature vector containing the coefficients of the projection is used to represent the image. We refer to these features as eigen-features. Here, we just summarize the main ideas [9]:

Representing each image $I(x, y)$ as a NxN vector Γ_i, first, the average face Ψ is computed:

$$\Psi = \frac{1}{R} \sum_{i=1}^{R} \Gamma_i$$

where is R is the number of faces in the training set. Next, the difference Φ of each face from the average face is computed: $\Phi_i = \Gamma_i - \Psi$. Then the covariance matrix is estimated by:

$$C = \frac{1}{R} \sum_{i=1}^{R} \Phi_i \Phi_i^T = AA^T ,$$

where, $A = [\Phi_1 \quad \Phi_2 \quad \cdots \quad \Phi_R]$. The eigenspace can then be defined by computing the eigenvectors μ_i of C. Since C is very large ($N^2 x N^2$), computing its eigenvectors will be very expensive. Instead, we can compute v_i, the eigenvectors of $A^T A$, an RxR matrix. Then, μ_i can be computed from v_i as follows:

$$\mu_i = \sum_{j=1}^{R} v_{ij} \Phi_j, \quad j=1 \cdots R$$

Usually, we only need to keep a smaller number of eigenvectors R_k corresponding to the largest eigenvalues. Given a new image, Γ, we subtract the mean ($\Phi = \Gamma - \Psi$) and compute the projection:

$$\tilde{\Phi} = \sum_{i=1}^{R_k} w_i u_i ,$$

where $w_i = u_i^T \Phi$ are the coefficients of the projection. In this paper, $\{w_i\}$ are our eigen-features.

The projection coefficients allow us to represent images as linear combinations of the eigenvectors. It is well known that the projection coefficients define a compact image representation and that a given image can be reconstructed from its projection coefficients and the eigenvectors (i.e., basis). The eigenspace representation of images is very powerful and has been used in various applications such as image compression and face recognition. Here, we use the eigen-features for gender classification.

3. FEATURE SELECTION

Most gender classification methods in the literature follow the same basic strategy: (a) feature extraction is applied on the raw images, (b) a classifier is trained using all the features extracted from the images. The problem with this strategy is that it uses all the extracted features for gender classification. As a result, gender-irrelevant information might be fed to the gender classifier. This might not allow the classifier to generalize nicely, especially when the training set is small.

As we discussed in the previous section, we represent each image in terms of a set of eigen-features. It has been found in several studies that different eigenvectors encode different kind of information [12]-[15]. For example, the first few eigenvectors seem to encode lighting while other eigenvectors seem to encode features such as glasses or moustaches [12]. We have made very similar observations in our case by analyzing the eigenvectors obtained from our training sets. Fig. 1, for example, shows some of the eigenvectors computed from our training data. Obviously, eigenvectors 1-4 encode light variations while eigenvectors 10 and 20 encode information about glasses.

Although many of the eigen-features are very important for face recognition, they might actually confuse the classifier in other applications such as in gender classification. In this study, we consider using GAs to

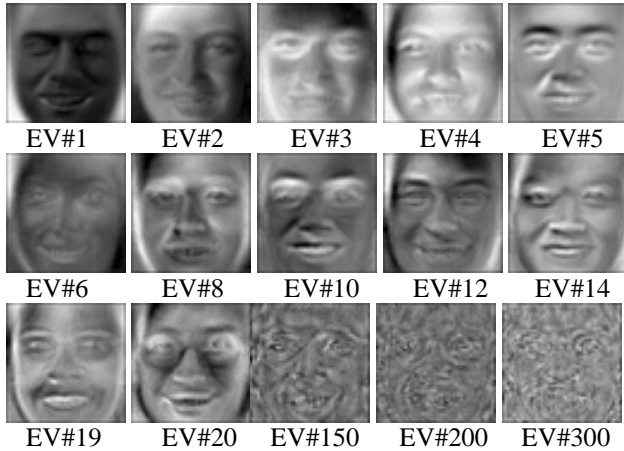select a good subset of eigen-features in order to improve gender classification performance.

Figure 1. Eigenvectors # 1-6, 8, 10, 12, 14, 19, 20, 150, 200 and 300.

4. GENETIC FEATURE SELECTION

4.1. A brief review of GAs
GAs are a class of optimization procedures inspired by the mechanisms of natural selection [16][17]. GAs operate iteratively on a population of structures, each of which represents a candidate solution to the problem, encoded as a string of symbols (chromosome). A randomly generated set of such strings forms the initial population from which the GA starts its search. Three basic genetic operators guide this search: selection, crossover and mutation

4.2. Overview of the Proposed Method
The main steps of the proposed method are as follows:

 (a) Eigen-feature extraction using PCA
 (b) Optimal eigen-feature subset selection using GAs; and training of the NN gender classifier
 (c) Classification of novel images

A binary encoding scheme is used to represent the presence or absence of a particular eigenvector from the linear expansion of the training images. Each individual in a generation represents an eigen-feature subset which is used to train a NN. The performance of the NN classifier is used to provide a measure of fitness used to guide the GA.

4.3. Encoding
Each image is represented as a vector of eigen-features which are the coefficients of the linear expansion of the image in the eigenspace. In our encoding scheme, the chromosome is a bit string whose length is determined by the number of eigenvectors. Each eigenvector, computed using PCA, is associated with one bit in the string. If the ith bit is 1, then the ith eigenvector is selected, otherwise, that

component is ignored. Each chromosome thus represents a different eigen-feature subset.

4.4. Fitness Evaluation
The goal of feature subset selection is to use fewer features to achieve the same or better performance. Therefore, the fitness evaluation contains two terms: (i) accuracy and (ii) number of features used. Only the features in the eigen-feature subset encoded by an individual are used to train the NN classifier. The performance of the NN is estimated using a validation data set (see section 5) and used to guide the GA. Each feature subset contains a certain number of features.

 If two subsets achieve the same performance, while containing different number of features, the subset with fewer features is preferred. Between accuracy and feature subset size, accuracy is our major concern. Combining these two terms, the fitness function is given as:

$$fitness = 10^4 \, Accuracy + 0.4 \times Zeros$$

where *Accuracy* is the accuracy rate that an individual achieves, and *Zeros* is the number of zeros in the chromosome. The accuracy ranges roughly from 0.5 to 1 (i.e., the first term assumes values in the interval 5000 to 10000). The number of zeros ranges from 0 to l where l is the length of the chromosome (i.e., the second term assumes values in the interval 0 to 100 since l=250 here).

 Overall, the higher the accuracy is, the higher the fitness is. Also, the fewer the number of features used the higher the number of zeros and as a result, the higher the fitness. It should be noted that individuals with higher accuracy will outweigh individuals with lower accuracy, no matter how many features they contain.

4.5. Initial Population
In general, the initial population is generated randomly, (e.g., each bit in an individual is set by flipping a coin). In this way, however, we will end up with a population where each individual contains the same number of 1's and 0's on the average. To explore subsets of different numbers of features, the number of 1's for each individual is generated randomly. Then, the 1's are randomly scattered in the chromosome.

4.6. Crossover
In general, we do not know how the eigenfeatures depend on each other. If dependent features are far apart in the chromosome, it is more probable that traditional 1-point crossover, will destroy the schemata. To avoid this problem, uniform crossover is used here.

4.7. Mutation
Mutation is a very low probability operator and just flips a specific bit. It plays the role of restoring lost genetic material. Our selection strategy was cross generational. Assuming a population of size N, the offspring double the size of

the population and we select the best *N* individuals from the combined parent-offspring population [18].

5. DATASET

The dataset used in this paper contains 400 frontal images from 400 distinct people, representing different races, with different facial expressions, and under different lighting conditions. Two hundred of them are male, and the rest are female. To compute the eigenvectors, the images were first registered using a procedure similar to that given in [11]. Specifically, the centers of the eyes, corners of the mouth, tip of the chin, and the top of the forehead were used to warp each face to the same scale, orientation and position. Then, each face was mapped to a 50x50 window. The mapping used was an affine transformation, computed iteratively in a least square fashion. Histogram equalization was also applied to each normalized image to account for different lighting conditions.

For each approach considered in our experiments, the average error rate was recorded using a three-fold cross-validation procedure (i.e., Data Set1, Data Set2, and Data Set3). To do this, we randomly split the dataset three times by keeping 300 images (150 female and 150 male) for training, 50 images for validation (25 female nd 25 male) and 50 images for testing (25 female 25 male). The validation set is strictly used to evaluate the goodness of a given subset of selected eigen-features. As mentioned above, the dataset used in this study contains 400 images from 400 distinct people. This makes this dataset harder than datasets used in other studies, where the same person is represented multiple times in the dataset. In this case, the training and test data sets might not be independent [5][7].

6. EXPERIMENTS AND RESULTS

We have performed a number of experiments and comparisons in order to demonstrate the performance of the proposed gender classification approach. First, we trained a NN classifier using very low-resolution images (20x20). This experiment was motivated by previous studies indicating that low-resolution images are sufficient for gender classification [4][8]. The input to the NN are the raw pixel values, normalized in the interval [0,1] (we will be referring to this approach as *NN-raw*). We used a 2-layer NN with 400 nodes in the input layer and 1 node in the output layer. We performed several runs by varying the number of nodes in the hidden layer in order to evaluate the performance of this approach. The back-propagation rule with momentum was used for training the NNs in all of our NN experiments. Training continues until the error on the validation set becomes less than a small threshold. Table 1 shows the average error obtained using this approach. The overall average error is 20.23%.

Next, we trained a NN using eigen-features (we will be referring to this approach as *NN-PCA*). In each run, the eigenspace was computed by applying PCA on the images of the training set. Each image in the training, validation,

and test sets was then represented by the coefficients of the linear expansion of the image in the eigenspace. Each of the training sets contains 300 images which implies that we can compute up to 300 eigenvectors using the methodology in [9]. Only the first 10 to 250 eigenvectors were used in our experiments (the last 50 eigenvectors seemed to encode mostly noise). A 2-layer network, trained by the back-propagation algorithm, was used in this experiment. We performed several runs again by varying both the number of hidden nodes (from 5 to 40) and the number of eigenvectors (from 10 to 250, ordered based on their corresponding eigenvalues). We used one output node with the number of input nodes being determined by the number of eigenvectors used. Table 2 shows the average error rate obtained for the best network tried. The overall error rate is 17.7%, which is lower that the error rate obtained using the *NN-raw* approach.

Table 1. Error rates for the NN-Raw approach.

# hidden nodes	Data Set1	Data Set2	Data Set3	Average
20	22%	26%	20%	22.7%
40	18%	20%	16%	18%
60	20%	24%	16%	20%

Table 2. Error rates for the NN-PCA approach

# EV(top)	#hidden nodes	Data Set1	Data Set2	Data Set3	Average
10	5	18%	14%	14%	15.3%
20	5	14%	22%	18%	18%
30	10	16%	16%	20%	17.3%
50	15	18%	22%	16%	18.6%
150	40	18%	24%	16%	19.3%

For comparison purposes, we also implemented a Bayes classifier (assuming equal priors) using eigen-features (we will be referring to this approach as *Bayes-PCA*). Both the male and female classes were modeled using a Gaussian distribution whose mean and covariance were estimated using the sample mean and sample covariance from each class. Several runs were performed again by varying the number of eigenvectors (from 10 to 150). The reason we did not use more than 150 eigenvectors in this case is because the covariance matrices of the Gaussians become singular using more eigen-features. Table 3 shows the average error rate obtained in these runs. It is interesting to note that the performance of this approach degrades significantly using more eigen-features. The overall average error rate obtained is 22.38%, which is worst than both the *NN-raw* and *NN-PCA* approaches.

Table 3. Error rates for the Bayes-PCA approach.

# EV(top)	Data Set 1	Data Set2	Data Set3	Average
10	16%	20%	16%	17.3%
20	16%	20%	22%	19.3%
30	14%	18%	16%	16%
50	18%	22%	20%	20%
150	40%	30%	48%	39.3%

In the next set of our experiments, we used GAs to select an optimum subset of eigen-features for gender classification. First, we used the GA approach to select a set of good eigen-features for the Bayes classifier (we will be referring to this approach as *Bayes-PCA+GA*). Then, we used the GA approach to select a set of good eigen-features for the NN classifier (we will be referring to this approach as *NN-PCA+GA*). The GA parameters we used in both cases are as follows: population size: 350, number of generations: 400, crossover rate: 0.66 and mutation rate: 0.04. Figure 2 shows the average error rate obtained in these runs. The *NN-PCA+GA* approach yielded an average error rate of 11.3%, which is much lower than 17.7% yielded by the *NN-PCA* approach. The *Bayes-PCA+GA* approach yielded an average error rate of 13.3% which is slightly higher that the error obtained by the *NN-PCA+GA* approach but much lower compared to the *Bayes-PCA* approach. Overall, using genetic search for eigen-feature selection has improved the performance of both the NN and Bayes classifiers.
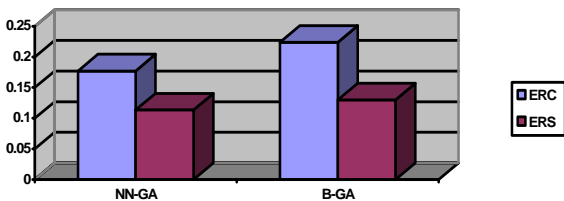

Figure 2. Error rates of various classifiers using features subsets selected manually or by the GA approach ERC: the error rate using the manually selected feature subsets; ERS: error rates using GA selected feature subsets.

The number of eigen-features selected by the *NN-PCA+GA* approach was 17.6% of the complete set of 250 eigenvectors. In terms of information contained in the selected feature subsets, the NN feature subset contains 38% of the information contained in the 250 eigenectors. In the case of the *Bayes-PCA+GA* approach, the number of eigen-features selected was 13.3% of the original set of 150 eigenvectors. This corresponds to 31% of the information in the complete set. Figure 3 shows these results.
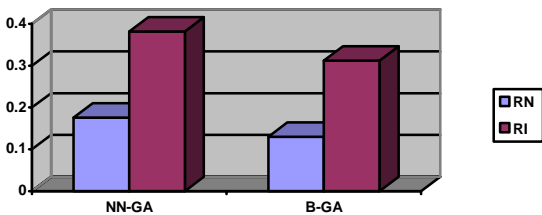

Figure 3. A comparison between the feature subset and the complete feature set: RN: the ratio between the number of features in the feature subset and the complete feature set; RI: the percentage of the information contained in the feature subset.

7. DISCUSSION

To get an idea about the optimal set of eigenvectors selected by the GA for the Bayes and NN classifiers, we have computed two histograms (see Figs. 4 and 5), showing the average distribution of the selected eigenvectors, over the three training sets used in our experiments. The x-axis corresponds to the eigenvectors, ordered by their eigenvalues, and has been divided into intervals of length 10. The y-axis corresponds to the average number of times an eigenvector within some interval has been selected by the GA in the final solution. It should be reminded again that the *Bayes-PCA+GA* approach was run using only the first 150 eigenvectors (see section 6). Figs. 4 and 5 show that both approaches have selected eigenvectors from the whole range of eigenvectors. Most interesting, the distributions are multimodal which is in contrast to the traditional approach.
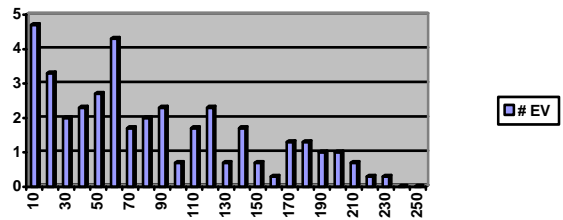

Figure 4. The distribution of eigenvectors selected by the NN-PCA+GA approach.
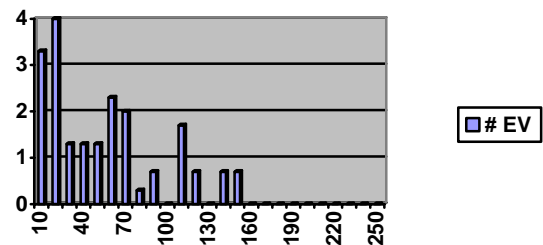

Figure 5. The distribution of eigenvectors selected by the Bayesian-PCA+GA approach. Note that in this experiment, the GA was run using the first 150 eigenvectors only (see text).

As we have discussed in section 3, different eigenvectors seem to encode different kind of information. For visualization purposes, we have reconstructed the facial images using the selected eigenvectors only (Fig. 6). Several interesting comments can be made through observing the reconstructed images. First of all, it is obvious that face recognition can not be performed based on the reconstructed faces using only the eigenvectors selected by the GA —they all look fairly similar to each other. In contrast, the reconstructed faces using the best eigenvectors (i.e., principal components) do reveal identity as can be seen from the images in the second row. The reconstructed images from eigenvectors selected by the GA, however, do disclose strong gender information—the

reconstructed female faces look more "female" than the reconstructed male faces. This implies that the GA did select out eigenvectors that seem to encode gender information. Second, those eigenvectors encoding features unimportant for gender classification seem to have been discarded by the GA. This is obvious from the reconstructed face images corresponding to the first two males shown in Fig. 6. Although both of them wear glasses, the reconstructed faces do not contain glasses which implies that the eigenvectors encoding glasses have not been selected by the GA. Note that the reconstructed images using the first 30 most important eigenvectors (second row) preserve features irrelevant to gender classification (e.g., glasses).
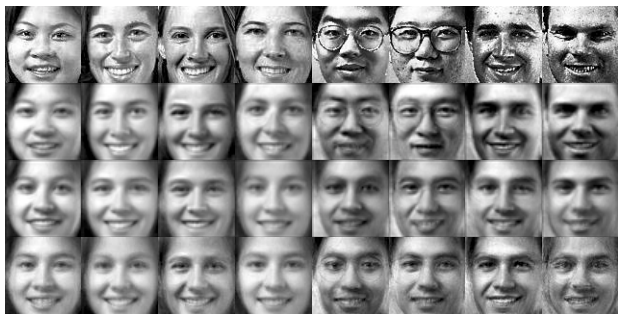

Figure 6. Reconstructed images using the selected feature subsets. The images in the first row are the original images, the images in the second row are the reconstructed images using top 30 eigenvectors, the images in the third row are the reconstructed images using the eigenvectors selected by the Bayes-PCA+GA approach, and the images in the fourth row are the reconstructed images using the eigenvectors selected by the NN-PCA+GA approach.

8. CONCLUSIONS

We have considered the problem of gender classification from frontal facial images. A NN approach with an automatic eigen-feature selection scheme based on GAs was proposed in this paper. By reducing the irrelevant information and using only the selected eigen-feature subset, the NN classifier showed significant improvement. Our method could provide valuable insights into other pattern classification problems—how to extract and use only the relevant features for a particular pattern classification task, especially when the amount of training data is limited. For future work, we plan to further explore the selected feature subsets to better understand the relation among them. Hopefully, this investigation will allow us to get some insights into the distribution of gender-related features in facial images. Moreover, we plan to test the genetic eigen-feature selection scheme using more datasets (e.g., vehicle classification) and more classifiers such as Linear Discriminant Analysis (LDA) [10] and Support Vector Machines (SVMs) [10].

References

[1] M. Hall. *Correlation-based Feature Subset Selection for Machine Learning,* PhD dissertation, Department of Computer Science, University of Waikato, 1999.
[2] D. Koller, and M. Sahami. "Towards optimal feature selection" *ICML-96,* Bari, Italy,87-95, 1996.
[3]F. Brill, D. Brown, and W. Martin. "Fast genetic selection of features for neural network classifers" IEEE *Transaction on Neural Networks*, 3(2), 324-328, 1992.
[4] B. Golomb, D. Lawrence, and T.J. Sejnowski. "SEXNET: A neural network identifies sex from human faces". *Advances in Neural Information Processing Systems (NIPS),* pp 572-577, 1991.
[5] R. Brunelli and T. Poggio. "HyperBF networks for gender classification", *DARPA Image Understanding Workshop*, pp 311-314, 1992.
[6] M. Burton, V. Bruce and N. Dench, "What's the difference between men and women? Evidence from facial measurement". *Perception*, vol. 22, pp. 153-176, 1993.
[7] M. Gray, D. Lawrence, B. Golomb and T. Sejnowski. "A perceptron reveals the face of sex". *Neural Computation*, 7(6), pp. 1160-1164, 1995.
[8] B. Moghaddam and M. Yang. "Gender classification with support vector machines" *4th IEEE Intl. Conf. on Automatic Face and Gesture Recognition*, pp. 306-311, 2000.
[9] M. Turk and A. Pentland. "Eigenfaces for recognition", *Journal of cognitive neuroscience*. vol 3, no.1 pp. 71-86, 1991.
[10] R. Duda, P. Hart, and D. Stork *Pattern Classification*, John-Wiley, 2nd edition, 2001
[11] G. Bebis, S. Uthiram, and M. Georgiopoulos**, "**Face detection and verification using genetic search". *International Journal on Artificial Intelligence* Tools *(IJAIT)***,** vol 9, no 2, pp. 225-246, 2000.
[12] W. Yambor, B. Draper, and R. Beveridge, "Analyzing PCA-based Face Recognition Algorithms: Eigenvector Selection and Distance Measures", *2nd Workshop on Empirical Evaluation in Computer Vision,* 2000.
[13] D. Valentin et al, "Principal Component and Neural Network Analyses of Face Images: What can be Generalized in Gender Classification?", *Journal of Mathematical Psychology,* vol. 41, pp. 398-413, 1997.
[14] H. Abdi et al., "More about the difference between men and women: evidence from linear networks and the principal component approach", *Perception,* vol. 24, pp. 539-562, 1995.
[15] B. Edelman et al., "Sex Classification of Face Areas: How Well can a Linear Neural Network Predict Human Performance?", *Journal of Biological Systems,* vol. 6, no. 3, 1998.
[16] D. Goldberg*, Genetic Algorithms in Search, Optimization, and Machine Learning,* Addison Wesley, 1989.
[17] J. Holland, *Adaptation in Natural and Artificial Systems,* MIT Press, 1992.
[18] L. Eshelman, "The CHC adaptive search algoruthm: how to have safe search when engaging in non-traditional genetic recombination", *Proceedings of the Foundation of Genetic Algorithms Workshop,* pp. 265-283, 1989.