Genetic Feature Subset Selection for Gender Classification: A Comparison Study

Zehang Sun[1],George Bebis[1], Xiaojing Yuan[1], and Sushil J. Louis[2]
[1]Computer Vision Lab. Department of Computer Science, University of Nevada, Reno
[2]Genetic Adaptive Systems Lab, Department of computer science, University of Nevada, Reno
(zehang,bebis,xjyuan,sushil)@cs.unr.edu

Abstract— **We consider the problem of gender classification from frontal facial images using genetic feature subset selection. We argue that feature selection is an important issue in gender classification and demonstrate that Genetic Algorithms (*GA*) can select good subsets of features (i.e., features that encode mostly gender information), reducing the classification error. First, Principal Component Analysis (*PCA*) is used to represent each image as a feature vector (i.e., eigen-features) in a low-dimensional space. Genetic Algorithms (*GAs*) are then employed to select a subset of features from the low-dimensional representation by disregarding certain eigenvectors that do not seem to encode important gender information. Four different classifiers were compared in this study using genetic feature subset selection: a Bayes classifier, a Neural Network (*NN*) classifier, a Support Vector Machine (*SVM*) classifier, and a classifier based on Linear Discriminant Analysis (*LDA*). Our experimental results show a significant error rate reduction in all cases. The best performance was obtained using the SVM classifier. Using only 8.4% of the features in the complete set, the SVM classifier achieved an error rate of 4.7% from an average error rate of 8.9% using manually selected features.**

I. Introduction

A successful gender classification approach can boost the performance of many other applications including face recognition and smart human-computer interfaces. Despite its importance, it has received relatively little attention in the literature. An automatic feature-selection-based gender classification scheme is proposed in this paper. We argue that feature selection is important for gender classification, and demonstrate that, by removing features that do not encode important gender information from the representation of the faces (e.g., eigenvectors encoding information about glasses), the error rate can be reduced significantly.

A. Previous Work

Based on the type of features used, existing gender classification approaches fall into one of two categories: geometry-based and appearance-based.

Geometry-based methods use metric features, e.g., face width, face length, mouth size, eye size, distances, angles and areas among salient feature points (eyes, nose, etc.). In Burton et al. [1], 73 points were extracted from a database containing 179 frontal facial images. Discriminant analysis was then used to classify gender using point-to-point distances. The accuracy reported on the training data was 85%. Fellous et al. [2] computed 22 normalized distances using a database with 109 images. The accuracy reported in that work was 90%. Brunnelli et al [3] used 16 geometrical features as the input to two competing hyper-basis function networks. A database with 168 images was used for training. The reported accuracy was 79% using novel faces.

Appearance-based methods learn the decision boundary between the male and female classes from training imagery without extracting any geometrical features. A representative method belonging to this category is the eigenface approach [4]. Cottrell et al [5] has proposed a face categorization method using a two-stage neural network, one for face compression and one for face classification. The output of the hidden layer of the compression network performs dimensionality reduction similar to the eigenface method. The accuracy reported was 63% on a database containg 64 images. Colomb et al. [6] used a similar method and referred to their gender classification network as *SEXNET*. Using a database containing 90 images, they reported 91.9% accuracy. Yen et al. [7] followed the same scheme using a larger database (i.e., 1400 face images). They reported 90% accuracy.

Abdi et al. [8] compared raw image with PCA-based image representations using Radial Basis Function (*RBF*) and perceptron networks. Using 160 facial images, the best performance of 91.8% was achieved by a perceptron classifier trained with PCA-based features. O'Toole et al. [9], [10] have also reported good performance using PCA and neural networks. Using raw images, Moghaddam et al. [11] investigated gender classification using SVMs on a database with 1755 face images. They reported 96.6% accuracy using RBF kernels.

B. Feature selection

Brunelli et al. [3] suggested that information useful for classifying faces according to their sex can be captured by a very limited number of geometrical measurements. Moreover, several studies [9][10] have shown that some of the eigenvectors are more useful for predicting sex of faces than others. For example, [9] reported 74.3% classification accuracy by using only the first four eigenvectors. A question that arises very naturally from these studies is "which facial features are most suitable for gender classification?"

Most gender classification methods reported in the literature use all the features extracted for classification pur-

poses. As a result, gender-irrelevant information might be fed to the gender classifier which might not allow the classifier to generalize nicely, especially when the training set is small. Exhaustive evaluation of possible feature subsets is usually computationally prohibitive in practice. A number of feature selection approaches have been proposed in the literature (see Siedlecki et al. [12], Jain et al. [13] for comprehensive surveys).

C. Overview of proposed method

Automatic feature subset selection distinguishes the proposed gender classification method from all other reported approaches. In particular, GAs [14] are employed to select features that encode important gender information and improve classification performance. GAs belong to the class of randomized heuristic search techniques, offering an attractive approach to feature subset selection. Although they have been used in various pattern recognition applications, their use in the area of computer vision is rather limited. Siedlecki et al. [15] has presented one of the earliest studies of GA-based feature selection in the context of a K-nearest-neighbor classifiers. Roth et al. [16] have proposed extracting geometric features using GAs. Yang et al. [17] have also proposed using GAs for feature selection. Using several benchmark real-world problems, they reported improved classification performance using NNs. Chitoui et al. [18] investigated the use of GAs for feature selection in a seed discrimination problem.

In our approach, facial images without hair are represented in a low-dimensional space, computed by using PCA [4]. GAs are then used to select gender-related PCA features. Four classifiers (i.e., Bayesian Classifier, LDA, NNs and SVMs), which have been used extensively in previous gender classification studies, are compared in this study. Our experimental results show significant error rate reductions for all the classifiers considered. For visualization purposes, we reconstruct the faces using the selected eigenfeatures. Although the reconstructed images have lost information about identity, they do disclose strong gender information. This implies that GAs can select eigenvectors encoding mostly gender information. Some results using the NN classifier have been reported previously in [19].

The rest of the paper is organized as follows: In Section II, feature selection in the context of gender classification is addressed. Section III presents a brief review on the classifiers used. In section V, we present the genetic search approach for eigen-feature selection. The database and preprocessing are discussed in section VI. Experimental results and comparisons are presented in section VII. Section VIII discusses the results and section IX concludes with possible directions for future work.

II. Eigen-features

Eigenspace representations of images use PCA [4] to linearly project images onto 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. We refer to the projection coefficients of an image on this space as eigen-features. It has been found in several studies that different eigenvectors encode different kind of information [20], [21], [8]. For example, the first few eigenvectors seem to encode lighting while other eigenvectors seem to encode features such as glasses or moustaches [20]. 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 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.

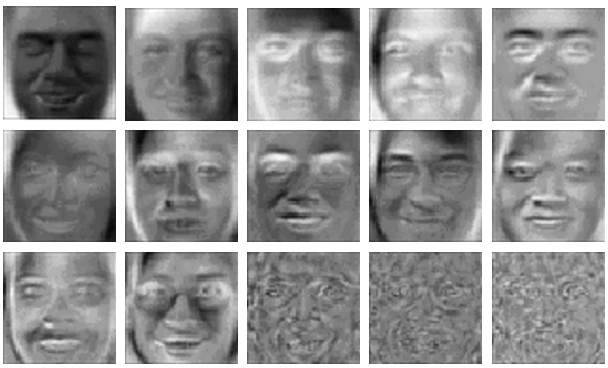

Fig. 1. Eigenvectors (from left to right and from top to bottom): No. 1-6, 8, 10, 12, 14, 19, 20, 150, 200 and 250.

III. classifiers

A. Neural Network

Fully connected, 2-layer networks trained by the backpropagation algorithm are used in this study [22]. Sigmoidal activation units are used in the hidden and output layers. NNs can directly construct highly non-linear decision boundaries, without estimating the probability distribution of the data.

B. Bayesian classifers

Each feature in this case is assumed to be a random variable. Given some features $\mathbf{x}$, classification is performed by computing the posterior probabilities for each class using the Bayes rule:

$$P(\omega_j|\mathbf{x}) = \frac{p(\mathbf{x}|\omega_j)P(\omega_j)}{p(\mathbf{x})} \tag{1}$$

where ω_j corresponds to class j and $P(\mathbf{x}|\omega_j)$ is the likelihood. In this study, the likelihoods are modeled as d dimensional multivariate Gaussians with different covariance matrices:

$$p(\mathbf{x}|\omega_j) = \frac{1}{(2\pi)^{d/2} \mid \Sigma_j \mid^{1/2}} exp[-\frac{1}{2}(\mathbf{x}-\mu_\mathbf{j})^\mathbf{T}\mathbf{\Sigma^{-1}}(\mathbf{x}-\mu_\mathbf{j})] \tag{2}$$

where $\mathbf{x}$ is the d-dimensional vector, μ_j is the d-dimensional mean vector of the j^{th} class, Σ_j is the $d \times d$ covariance matrix of j^{th} class. Maximum likelihood method is used to estimate the mean and covariance matrix for each class.

C. Linear Discriminant Analysis (LDA)

The objective of LDA is to find a projection, $y = W^T x$ (where x is the input image and W the projection matrix), that maximizes the ratio of the between-class scatter and the within-class scatter [22], [11]. To avoid problems with singularities of the within-class scatter matrix (i.e., when the training data is much smaller than the dimension of the data), the original space is projected first onto a smaller, intermediate space using PCA and then onto a final space using LDA. Since there are only two classes here (male/female), the LDA space is one-dimensional. The Bayes classifier is used again with LDA features (the likelihoods are modeled with 1D Gaussians in this case).

D. Support Vector Machine

SVMs are primarily two-class classifiers that have been shown to be an attractive and more systematic approach to learning linear or non-linear decision boundaries [23] [24]. Given a set of points, which belong to either of two classes, *SVM* finds the hyper-plane leaving the largest possible fraction of points of the same class on the same side, while maximizing the distance of either class from the hyper-plane. This is equivalent to performing structural risk minimization to achieve good generalization [23] [24]. Assuming l examples from two classes

$$(x_1, y_1)(x_2, y_2)...(x_l, y_l),\ x_i \in R^N, y_i \in \{-1, +1\} \qquad (3)$$

finding the optimal hyper-plane implies solving a constrained optimization problem using quadratic programming. The optimization criterion is the width of the margin between the classes. The discriminate hyper-plane is defined as:

$$f(x) = \sum_{i=1}^{l} y_i a_i k(x, x_i) + b \qquad (4)$$

where $k(x, x_i)$ is a kernel function and the sign of $f(x)$ indicates the membership of x. Constructing the optimal hyper-plane is equivalent to finding all the nonzero a_i. Any data point x_i corresponding to a nonzero a_i is a support vector of the optimal hyper-plane. The Gaussian kernel is used in this study (i.e., our experiments have shown that the Gaussian kernel outperforms other kernels in the context of our application).

IV. Background on Genetic Algorithms

Goldberg [14] provides a nice introduction to GAs and the reader is referred to this source as well as the survey paper of Srinivas and Patnaik [25] for further information. GAs are a class of optimization procedures inspired by the biological mechanisms of reproduction. In the past, they have been used in various applications including target recognition [26], face detection and verification [27], image registration [28], and object recognition [29].

GAs operate iteratively on a population of structures, each one of which represents a candidate solution to the problem at hand, properly encoded as a string of symbols (e.g., binary). 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. The genetic search process is iterative: evaluating, selecting, and recombining strings in the population during each iteration (generation) until reaching some termination condition.

Evaluation of each string is based on a fitness function that is problem-dependent. It determines which of the candidate solutions are better. Selection of a string, which represents a point in the search space, depends on the string's fitness relative to that of other strings in the population. It probabilistically removes from the population those points that have relatively low fitness. Mutation, as in natural systems, is a very low probability operator and just flips a specific bit. Mutation plays the role of restoring lost genetic material. Crossover in contrast is applied with high probability. It is a randomized yet structured operator that allows information exchange between points. Its goal is to preserve the fittest individuals without introducing any new value.

Selection probabilistically filters out solutions that perform poorly, choosing high performance solutions to concentrate on or *exploit*. Crossover and mutation, through string operations, generate new solutions for *exploration*. Given an initial population of elements, GAs use the feedback from the evaluation process to select fitter solutions, generating new solutions through recombination of parts of selected solutions, eventually converging to a population of high performance solutions.

V. genetic feature selection

A. 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 is associated with one bit in the string. If the i^{th} bit is 1, then the i^{th} eigenvector is selected, otherwise, that component is ignored. Each chromosome thus represents a different eigen-feature subset.

B. 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.

C. 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 from the validation data and (ii) number of features used. Only the features in the eigen-feature subset encoded by an individual are used to train a classifier. The performance of the classifier is estimated using a validation data set 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:

$$fitness = 10^4 \times Accuracy + 0.4 \times Zeros \qquad (5)$$

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

D. Crossover and Mutation

In general, we do not know how the eigen-features depend upon 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. 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 is 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 [30].

VI. Dataset

The dataset used contained 400 frontal images from 400 distinct people, representing different races, with different facial expressions, and under different lighting conditions. The 400 images were equally divided between males and females. To compute the eigenvectors, the images were first registered using the procedure given in [27]. 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 database three times by keeping 300 images (150 females and 150 males) for training, 50 images for validation (25 females and 25 males) and 50 images for testing (25 females, 25 males). The validation sets were used to terminate the training of the NN classifier while the test sets were strictly used to evaluate the suitability of a given subset of selected eigen-features. As mentioned above, the database used in this study contains 400 images from 400 distinct people. This results in a more difficult database than those used in other studies, where the same person appears multiple times in the dataset [3].

VII. 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, each classifier was tested using manually selected eigen-features. The NN classifier used was a 2-layer network trained by the back-propagation algorithm (we will be referring to this approach as *NN-PCA*). Several runs were performed varying both the number of hidden nodes (from 5 to 40) and the number of eigenvectors (from 10 to 150). We used one output node with the number of input nodes being determined by the number of eigenvectors used. In the case of the Bayes classifier, both the male and female classes were modeled using Gaussian densities in the eigenspace, assuming equal priors (we will be referring to this approach as *Bayes-PCA*). We run several experiments varying the number of eigenvectors from 10 to 150 (i.e., using more than 150 eigenvectors yields singular covariance matrices due to the relatively small number of training data). The LDA was tested using PCA as a preprocessing step (we will be referring to this approach as LDA-PCA). Several experiments were performed varying the number of eigenvectors kept in the intermediate eigenspace (from 10 to 150). Finally, the SVM classifier was trained using a Gaussian kernel (we will be referring to this approach as *SVM-PCA*). Several runs were performed again by varying the width of the Gaussian kernel and the number of eigenvectors (from 10 to 150). Table I summarizes the results obtained (see also Figure 2). The best performance, 8.9%, was obtained using the SVM classifier.

TABLE I

Error rate(ER) using manually selected features. In the 3^{rd} column, the numbers in parentheses indicate the number of hidden nodes

EV	Bayes	NN	LDA	SVM
10	17.3%	15.3%(5)	13.3%	7%
20	19.3%	18%(5)	14.7%	8.7%
30	16%	17.3%(10)	14%	8%
50	20%	18.7%(15)	14%	8.7%
150	39.3%	19.3%(40)	15%	12%
Average	22.38%	17.7%	14.2%	8.9%

In the next set of our experiments, we used GAs to select optimum subsets of eigenvectors for gender classification. The GA parameters we used in all experiments are as fol-

lows: population size: 350, number of generations: 400, crossover rate: 0.66 and mutation rate: 0.04. It should be noted that in every case, the GA converged to the final solution much earlier (i.e., after 150 generations). Fig. 2 shows the average error rate obtained in these runs. The results illustrate clearly that the feature subsets selected by the GA have reduced the error rate of all the classifiers significantly: from 22.4% to 13.3% using the Bayes classifier, from 17.7% to 11.3% using NNs, from 14.2% to 9% using LDA, and from 8.9% to 4.7% using SVMs. The best classification performance was achieved by the SVM classifier. In these experiments, GAs searched the space of the first 250 eigenvectors (except in the case of the Bayes-PCA+GA approach, where they searched the space of the first 150 eigenvectors only). 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 eigenvectors. 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. The number of eigen-features selected by LDA-PCA+GA approach was 36.4% of the original set, which contained 61.2% information. The smallest number of eigen-features were selected by the SVM-PCA+GA approach − only 8.4% of the features in the original set were selected, containing 32.4% information. Fig. 2 shows these results.

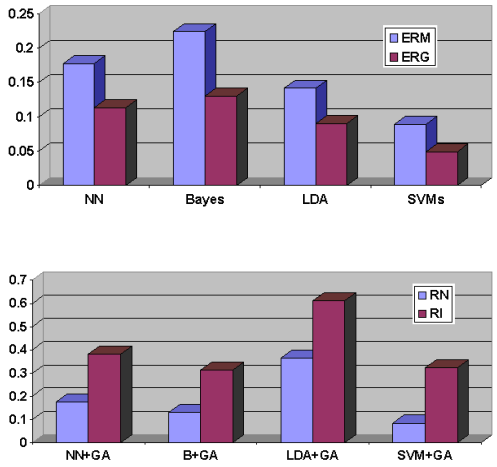

Fig. 2. *(Top)*: Error rates of various classifiers using features subsets selected manually or by GAs. ERM: the error rate using the manually selected feature subsets; ERG: error rate using GA selected feature subsets. *(Bottom)*: A comparison between the automatically selected feature subsets and the complete feature set. RN: the ratio between the number of features in the GA-selected feature subsets and the complete feature set; RI: the percentage of the information contained in the GA-selected feature subset.

VIII. Discussion

To get an idea regarding the optimal set of eigenvectors selected by GAs for each classifier, we computed histograms (see Fig.3), showing the average distribution of the selected eigenvectors (i.e, over the three training sets). 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 GAs in the final solution. Note that the Bayes-PCA+GA and Bayes-PCA+SFBS approach was run using only the first 150 eigenvectors. Fig.3 illustrates that the GA has selected eigenvectors from the entire range of eigenvectors for all the approaches. It can be easily noted that the selected eigenvectors have some overlap, for example, they all contain the first principal component.

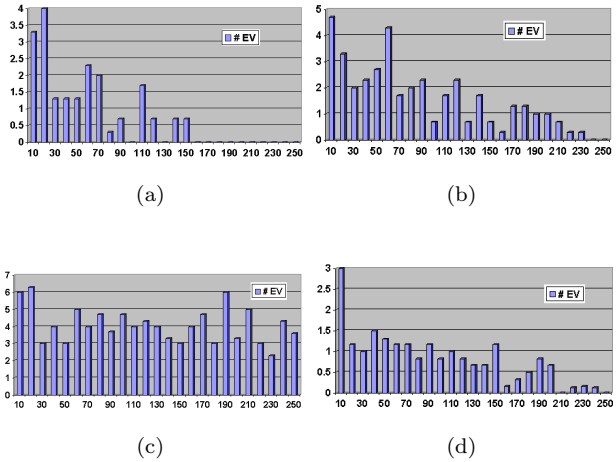

Fig. 3. The distribution of eigenvectors selected by (a) Bayes-PCA+GA, (b) NN-PCA+GA (c) LDA-PCA+GA. (d) SVM-PCA+GA.

Fig. 4. Reconstructed images using the selected feature subsets. First row: original images; Second row: using top 30 eigenvectors; Third row: using the eigenvectors selected by Bayes-PCA+GA; Fourth row: using the eigenvectors selected by NN-PCA+GA; Fifth row: using the eigenvectors selected by LDA-PCA+GA; Sixth row: using the eigenvectors selected by SVM-PCA+GA.

As we have discussed in Section II, different eigenvectors seem to encode different kinds of information. For visualization purposes, we have reconstructed the facial images using the selected eigenvectors only (Fig.4). Several interesting comments can be made through observing the reconstructed images using feature subsets selected by GAs.

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.4. 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).

IX. Conclusions

We have considered the problem of gender classification from frontal facial images using feature subset selection. An automatic eigen-feature selection scheme based on GAs was proposed in this paper. By reducing irrelevant information and using only selected eigen-feature subsets, the four classifiers (Bayes, NN, LDA and SVM) showed significant performance improvements. 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 number of training examples is limited. For future work, we plan to further explore the selected feature subsets to better understand the relationships among them. Hopefully, this investigation will allow us to get some insights about 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) as well as features extracted by various methods (e.g., Gabor features).

Acknowledgements: This research was supported by NSF under CRCD grant No. 00880861 and grant No. 9624130.

References

[1] A. Burton, V. Bruce and N. Dench, "What's the difference between men and women? evidence from facial measurements," *Perception*, vol. 22, pp. 153–176, 1993.

[2] J. Fellous, "Gender discrimination and predication on the basis of facial metric information," *Vision Res.*, vol. 37, pp. 1961–1973, 1997.

[3] R.Brunelli and T. Poggio, "Hyperbf networks for gender classification," *DARPA Image Understanding Workshop*, pp. 311–314, 1992.

[4] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, pp. 71–86, 1991.

[5] G. Cottrell and M. Fleming, "Face recognition using unsupervised feature extraction," *International Conference on Neural Network*, pp. 322–325, 1990.

[6] B. Golomb, D. Lawrence and T. Sejnowski, "Sexnet: a neural network identifies sex from human faces," *Advances in Neural Information Processing Systems*, pp. 572–577, 1991.

[7] S. Yen, P. Sajda and L. Finkel, "Comparison of gender recognition by pdp and radial basis function networks," *The Neurobiology of Computation*, pp. 433–438, 1994.

[8] H. Abdi, D. Valentin, B. Edelman and A. O'Toole, "More about the difference between men and women: evidence from linear neural networks and the principal component approach," *Perception*, vol. 24, pp. 539–562, 1995.

[9] A. O'Toole, H. Adbi, K. Deffenbacher and J. Bartlett, "Classifying faces by race and sex using an autoassociative memory trained for recognition," *13th Annual Conference of the Cognitive Science Society*, pp. 847–851, 1991.

[10] A. O'Toole, H. Adbi, K. Deffenbacher and D. Valentin, "A low-dimensional representation of faces in the higher dimensions of space," *Journal of the Optical Society of America*, vol. 10, pp. 405–411, 1993.

[11] 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.

[12] W. Siedlecki and J. Sklansky, "On automatic feature selection," *International Journal of Pattern Recognition and Artificial Intelligence*, vol. 2, no. 2, pp. 197–220, 1988.

[13] A. Jain D. Zongker, "Feature selection: evaluation, application, and small sample performance," *IEEE Transactions on Pattern Analysis and Machine Interlligence*, vol. 19, pp. 153–158, 1997.

[14] D. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine Learning*. Addison Wesley, 1989.

[15] W. Siedlecki and J. Sklansky, "A note on genetic algorithm for large-scale feature selection," *Pattern Recognition Letter*, vol. 10, pp. 335–347, 1989.

[16] G. Roth and M. Levine, "Geometric primitive extraction using a genetic algorithm," *IEEE Transactions on Pattern Analysis and Machine Interlligence*, vol. 16, pp. 901–905, 1994.

[17] J. Yang and V. Honavar, "Feature subset selection using a genetic algorithm," *A Data Mining Perspective (H. Motoda and H. Liu eds) Chapt 8, Kluwer*, 1998.

[18] Y. Chtioui, D. Bertrand and D. Barba, "Feature selection by a genetic algorithm. application to seed discrimination by artificial vision," *J. Sci. Food Agric.*, vol. 76, pp. 77–86, 1998.

[19] Z. Sun, X. Yuan, G. Bebis, and S. Louis, "Neural-network-based gender classification using genetic search for eigen-feature selection," *IEEE International Joint Conference on Neural Networks*, May, 2002.

[20] 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.

[21] D. Valentin, H. Abdi, B. Edelman and A. O'Toole, "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.

[22] R. Duda, P. Hart, and D. Stork, *Pattern Classification*. Jon-Wiley, 2001.

[23] V. Vapnik, *The Nature of Statistical Learning Theory*. Springer Verlag, 1995.

[24] C. Burges, "Tutorial on support vector machines for pattern recognition," *Data Mining and Knowledge Discovery*, vol. 2, no. 2, pp. 955–974, 1998.

[25] M. Srinivas and L. Patnaik, "Genetic algorithms: A survey," *IEEE Computer*, vol. 27, pp. 17–26, 1994.

[26] A. Katz and P. Thrift, "Generating image filters for target recognition by genetic learning," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 16, 1994.

[27] G. Bebis, S. Uthiram, and M. Georgiopoulos, "Face detection and verification using genetic search," *International Journal on Artificial Intelligence Tools*, vol. 9, no. 2, pp. 225–246, 2000.

[28] J. Fitzpatrick, J. Grefenstette, and D. Gucht, "Image registration by genetic search," *IEEE Southeastcon*, pp. 460–464, 1984.

[29] G. Bebis, S. Louis, Y. Varol, and A. Yfantis, "Genetic object recognition using combinations of views," *IEEE Transactions on Evolutionary Computation*, vol. 6, pp. 132–146, 2002.

[30] L. Eshelman, "The chc adaptive search algoruthm: how to have safe search when engaging in non-traditional genetic recombination," *The Foundation of Genetic Algorithms Workshop*, pp. 265–283, 1989.