GA-Fisher: A New LDA-Based Face Recognition Algorithm With Selection of Principal Components

Wei-Shi Zheng, Jian-Huang Lai, and Pong C. Yuen

***Abstract*—This paper addresses the dimension reduction problem in Fisherface for face recognition. When the number of training samples is less than the dimension of the resolution of the image dimension (total number of pixels), the within-class scattered matrix (Sw) in Linear Discriminant Analysis (LDA) is singular, and Principal Component Analysis (PCA) is suggested to employ in Fisherface for dimension reduction of Sw so that it becomes nonsingular. The popular method is to select the largest nonzero eigenvalues and the corresponding eigenvectors for LDA. To attenuate the illumination effect, some researchers suggested removing the three eigenvectors with the largest eigenvalues and the performance is improved. However, as far as we know, there is no systematic way to determine which eigenvalues should be used. Along this line, this paper first proposes a theorem to interpret why PCA can be used in LDA and proposes an automatic and systematic method to select the eigenvectors to be used in LDA using a Genetic Algorithm (GA). A GA-PCA is then developed. It is found that some small eigenvectors should also be used as part of the basis for dimension reduction. Using the GA-PCA to reduce the dimension, a GA-Fisher method is designed and developed. Comparing with the traditional Fisherface method, the proposed GA-Fisher offers two additional advantages. First, optimal bases for dimensionality reduction are derived from GA-PCA. Second, the computational efficiency of LDA is improved by adding a whitening procedure after dimension reduction. The Face Recognition Technology (FERET) and Carnegie Mellon University Pose, Illumination, and Expression (PIE) databases are used for evaluation. Experimental results show that almost 5% improvement compared with Fisherface can be obtained, and the results are encouraging.**

***Index Terms*—Dimension reduction, face recognition, GA-PCA, genetic algorithms, LDA, PCA.**

I. Introduction

LINEAR DISCRIMINANT analysis (LDA) [1], [16] has been one of the popular techniques employed in the face recognition. The basic idea of the Fisher Linear Discriminant is to calculate the Fisher optimal discriminant vectors so that the ratio of the between-class scatter and the within-class scatter (Fisher Index) is maximized. In addition to maximizing the Fisher index, some restrictions, when finding the optimal vectors, are added to reduce the error rate in face recognition. In 1975, Foley and Sammon presented Foley–Sammon optimal discriminant vectors [5], and in 2001, Jin *et al.* [4] presented uncorrelated optimal discriminant vectors.

A number of LDA-based recognition algorithms/systems have been developed in the last few years. The most well known technique is the Fisherface, which combines the techniques of Principal Component Analysis (PCA) with the LDA. The superior performance of LDA on face recognition has been reported in many literatures and Face Recognition Technology (FERET) evaluation test [9]. Despite the advantages of using LDA on pattern recognition applications, LDA suffers from a small sample size (S3) problem. The problem happens when the number of training samples is less than total number of pixels in an image. Under this situation, the within-class scatter matrix will be singular. In turn the inverse of within-class scatter matrix cannot be calculated. S3 problem always occur in face recognition applications.

Basically, there are at least four approaches proposed to overcome the S3 problem. The most well-known technique is the Fisherface [1], which combines the techniques of Principal Component Analysis (PCA) with the LDA (it is also known as PCA+LDA). It performs the dimension reduction by PCA [6], [7] before LDA. The basic idea of PCA for dimension reduction is to keep the (top n) largest nonzero eigenvalues and the corresponding eigenvectors for LDA. The idea of this approach is correct from image compression point of view; keeping the largest nonzero principal components means that we keep most of the energy (information) of that image by projecting into lower dimension subspace. However, from the pattern classification point of view, this argument may not be true. The main reason is that, in pattern classification, we would like to find a set of projection vectors that can provide the highest discrimination between different classes. Therefore, choosing the largest principal components as the bases for dimensionality reduction may not be optimal. Along this line, Zhao *et al.* [16] and Pentland *et al.* [21] proposed to remove the largest three eigenvalues for representation. A lot of experiments have been performed to support this argument, but the theoretical justifications are not enough. The second approach is adding a small perturbation to the within-class scatter matrix [22], so that it becomes nonsingular. The idea is nice, but physical meaning and further analysis have not been given. The third approach is utilizing the pseudo-inverse of the within-class scatter matrix to solve the S3 problem [23]. The computation of this approach can be processed by QR decomposition [23], and it has been proved that the solution to the eigenproblem on LDA\QR is equivalent to the one on generalized LDA. The

Manuscript received May 6, 2004; revised November 26, 2004. This work was supported in part by the NSFC under Grant 60144001, the NSF of Guang-Dong, China under Grant 021766, the RGC Earmarked Research Grant HKBU-2119/03E, and the Key (Key grant) Project of Chinese Ministry of Education under Grant 105134. The Associate Editor recommending this paper was Vittorro Marino.

W.-S. Zheng and J.-H. Lai are with the Mathematics Department, Sun Yat-Sen University (ZhongShan University), Guangzhou 510275, China (e-mail: SunnyWeiShi@163.com; stsljh@zsu.edu.cn).

P. C. Yuen is with the Department of Computer Science, Hong Kong Baptist University, Hong Kong (e-mail: pcyuen@comp.hkbu.edu.hk).

Digital Object Identifier 10.1109/TSMCB.2005.850175

nullspace method [3] is another approach. The basic idea of this approach is to reduce the searching space in a particular subspace. Usually null space of the within-class scatter matrix or range of between-class scatter matrix is used.

It is known that dimension reduction in LDA is important. A poor strategy may lead to lost information and a poor recognition may emerge. This paper focuses on the first approach as mentioned. In particular, we would like to address the following questions.

1) PCA is typically used for dimension reduction. Why it can be used in LDA?
2) The popular methods select the largest principal components for dimension reduction, but how about the smallest ones? Can they be used? How to use?

This paper develops a framework of dimension reduction for LDA, and the contributions include the following.

1) A rigorous mathematical theory, which proves the feasibility of the PCA, used as a process of dimensional reduction for LDA, is developed.
2) We found that some smallest component principals are important, and they sometime greatly contribute to the accuracy of recognition. Along this line, a new principal component selection method named *GA-PCA* is developed.
3) Integrating the GA-PCA in LDA method, a new LDA-based algorithm, called *GA-Fisher*, is developed.
4) Many experiments with two widely accepted measurements, namely *Cumulative Match Characteristic* (CMC) and *Receiver Operating Characteristic* (ROC), are used for evaluation. The experimental results support the above-named arguments.

The outline of this paper is as follows. Section II provides the general background on *LDA* and *Fisherface*. In Section III, we perform an analysis on dimension reduction and develop a new PCA Dimension Reduction Theorem, called *PCA-DRT*. Based on PCA-DRT, Section IV proposes a new model of dimension reduction technique, namely *GA-PCA*, and its application to the LDA-based face recognition, namely *GA-Fisher*, is introduced in Section V. An interpretation on *Fisherface* will be discussed in Section VI. Experimental results are reported in Section VII. Finally, conclusions are given in Section VIII.

II. Background of LDA and Fisherface

Let us consider a set of N samples in the n-dimensional image space R^n, and assume that each image belongs to one of the K classes $\{C_1, C_2, \ldots, C_K\}$. Let us also consider N_j is the number of samples in class $C_j, u_j = (1/N_j)\sum_{x\in C_j} x$ is the mean image of class $C_j, u = (1/N)\sum_{j=1}^{K}\sum_{x\in C_j} x$ is the mean image of all samples.

Then, the within-class scatter matrix is defined as

$$S_w = \frac{1}{N}\sum_{j=1}^{K}\sum_{x\in C_j}(x-u_j)(x-u_j)^T = \Phi_w\Phi_w^T \qquad (1)$$

the between-class scatter matrix is defined as

$$S_b = \frac{1}{N}\sum_{j=1}^{K} N_j(u_j-u)(u_j-u)^T = \Phi_b\Phi_b^T \qquad (2)$$

and total-class scatter matrix is defined as

$$S_t = \frac{1}{N}\sum_{j=1}^{K}\sum_{x\in C_j}(x-u)(x-u)^T = \Phi_t\Phi_t^T = S_w + S_b. \qquad (3)$$

If S_w is not singular, LDA (also known as Fisher Linear Discriminant [15]) tries to find a projection $W_{\text{opt}} = (w_1, w_2, \ldots, w_L)$ that satisfies the Fisher criterion

$$W_{\text{opt}} = \arg\max_W \frac{|W^T S_b W|}{|W^T S_w W|} \qquad (4)$$

where $w_1, w_2, \ldots, w_L$ are the eigenvectors of $S_w^{-1}S_b$ corresponding to $L\ (\le K-1)$ largest eigenvalues $\lambda_1, \lambda_2, \ldots, \lambda_L$.

However, if S_w is singular, the inverse of S_w does not exist. In turn, *Fisherface* is usually adopted. Fisherface makes use of PCA to project the image set to a lower dimensional space so that within-class scatter matrix $\widehat{S}_w$ is nonsingular by (8) following (Section III, Remark 2 declares that it may be experiential) and then applies the standard LDA.

The optimal projection W_{opt} of Fisherface is normally given by

$$W_{\text{opt}}^T = W_{\text{fld}}^T W_{\text{pca}}^T \qquad (5)$$

$$W_{\text{pca}} = \arg\max_W |W^T S_t W|,$$

$$W_{\text{pca}} = \left(w_{\text{pca}_1}, \ldots, w_{\text{pca}_p}\right) \qquad (6)$$

$$W_{\text{fld}} = \arg\max_W \frac{|W^T\ \widehat{S}_b\ W|}{|W^T\ \widehat{S}_w\ W|} \qquad (7)$$

$$\widehat{S}_w = W_{\text{pca}}^T S_w W_{\text{pca}}, \quad \widehat{S}_b = W_{\text{pca}}^T S_b W_{\text{pca}} \qquad (8)$$

where $\{w_{\text{pca}_i}\}_{i=1}^{p}$ are the eigenvectors corresponding to $p\ (\le N-K)$ largest positive eigenvalues $\{\lambda_{\text{pca}_i}\}_{i=1}^{p}$ of S_t.

III. Analysis on Dimension Reduction and PCA Dimension Reduction Theorem (PCA-DRT)

This section is divided into two parts. First, we do some analysis on using PCA for dimension reduction. In the second part, we develop a PCA dimension reduction theorem.

A. Analysis

The use of PCA for dimension reduction in the Fisherface (PCA+LDA) has two purposes. First, when small sample size problem occurs, S_w becomes singular. PCA is used to reduce the dimension such that S_w becomes nonsingular. The second purpose is that even though there is no S3 problem, the use of PCA for dimension reduction in the Fisherface (PCA+LDA) is to reduce the computational complexity. This paper focuses on the S3 problem. In fact, our proposed method is generic can be served as second purpose as well.

PCA is a standard decorrelation technique and derives a set of orthogonal bases. In recent years, it has been suggested that

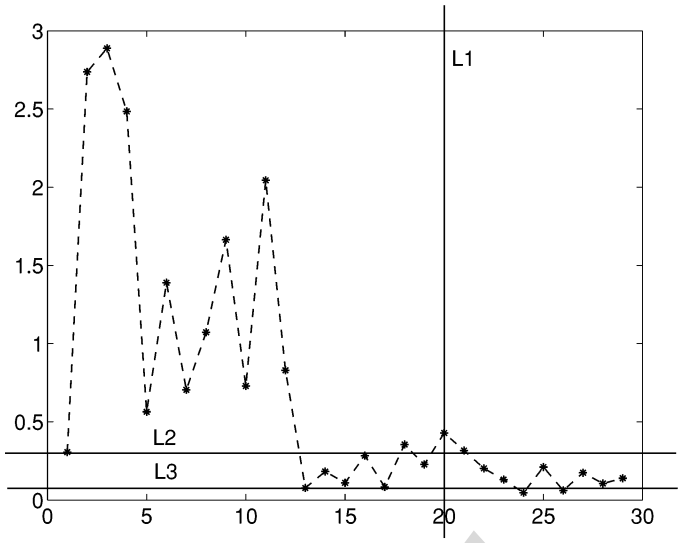


Fig. 1. Values of $(w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i})/(w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i})$ (y-axis) vs. index $i = 1, 2, \ldots, 29$ of eigenvalues (x-axis).

the largest p (=rank(S_w)) principal components are selected, and the corresponding eigenvectors have been selected as the bases. This would satisfy the *minimum mean squared error* criterion. However, from pattern classification criterion, are there any contributions from the smaller principal components?

From (3) and (6), it can be determined that

$$w_{\mathrm{pca}_i}^T S_t w_{\mathrm{pca}_i} = \lambda_{\mathrm{pca}_i} = w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i} + w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i}$$

where $w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i}$ can be viewed as the distance of within-class to which w_{pca_i} contributes, and $w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i}$ is the distance of between-class to which w_{pca_i} contributes. When λ_{pca_i} closes to zero, both $w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i}$ and $w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i}$ tend to zero.

The FERET database with ten persons and three training images/person is used to perform an experiment as an example. Fig. 1 plots a graph of index i of principal component w_{pca_i} (x-axis) against values of $(w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i})/(w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i})$ (y-axis). The smaller the index i represents the principal component with larger eigenvalue.

Although PCA has been employed in face recognition technology for more than 15 years, to the best of our knowledge, there is no rigid algorithm for determining which principal component(s) should be used for face recognition. The most popular guideline is to select the top n largest principal components for LDA. As an example, one would select the largest 20 principal components as shown in line L1. However, Fig. 1 shows that even i is larger than 20, there exists a value of $(w_{\mathrm{pca}_i}^T S_b w_{\mathrm{pca}_i})/(w_{\mathrm{pca}_i}^T S_w w_{\mathrm{pca}_i})$ (e.g., 21, 22, 23, 25, 27), which is larger than that in the range of 15 and 20. This intuitive observation indicates that some of the smaller principal components have better balance on maximizing the between-class distance while minimizing the within-class distance than the selected large principal components. Therefore, a strategy for selecting some smaller principal components as the bases is required.

To find the strategy, the first important point is to guarantee that after PCA projection, $\widehat{S}_w$ (=$W_{\mathrm{pca}}^T S_w W_{\mathrm{pca}}$) will be nonsingular. A simple mathematical inequality shows that $\mathrm{rank}(W_{\mathrm{pca}}^T S_w W_{\mathrm{pca}}) \leq \min\{\mathrm{rank}(S_w), \mathrm{rank}(W_{\mathrm{pca}})\}$. Therefore, there is a possibility that $\widehat{S}_w$ is still singular after PCA dimensionality reduction. A simple example to illustrate this practical problem is shown in Appendix C. This example shows that not all combinations of the principal components guarantee that $\widehat{S}_w$ is nonsingular. Because of this, we develop a PCA Dimension Reduction Theorem to ensure that $\widehat{S}_w$ could be nonsingular with selected principal components so that PCA can be used for dimension reduction in LDA.

B. PCA Dimension Reduction Theorem (PCA-DRT)

Theorem PCA-DRT: Let $m = \mathrm{rank}(S_t), l = \mathrm{rank}(S_w)$, using singular value decomposition, we get $S_t = W_t \Lambda_t W_t^T$, where $W_t = (w_1, w_2, \ldots, w_m), \Lambda_t = \mathrm{diag}(\lambda_1, \lambda_2, \ldots, \lambda_m), \lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_m > 0$. If $m \geq l$, then there exists $\hat{W}_t = (w_{i_1}, \ldots, w_{i_l})$ such that $S_w^{\#} = \hat{W}_t^T S_w \hat{W}_t$ is nonsingular, and $\mathrm{rank}(S_w^{\#}) = \mathrm{rank}(S_w)$, where i_j is a positive integer, $i_j \leq m, j = 1, 2, \ldots, l$ and $i_h \neq i_j$ for any $h \neq j$.

Detailed proof of Theorem PCA-DRT is shown in Appendix A.

Remark 1: The requirement $m \geq l$ in the PCA-DRT theorem can be removed. By Lemma 2 mentioned in Appendix A, it is easy to obtain $\mathrm{rank}(N(S_t)) \leq \mathrm{rank}(N(S_w))$; therefore, $m = \mathrm{rank}(S_t) \geq l = \mathrm{rank}(S_w)$ has no doubt.

Remark 2: This theorem tells us that selecting the l largest eigenvectors of S_t may not always guarantee that the within-class scatter matrix is nonsingular after reducing the dimension. Appendix D gives an example to show that it is not always true from the mathematical point of view. In turn, we will provide a more sensible interpretation in Section VI.

Remark 3: It is easy to conclude that one can select l_1 ($<l$) principal components as bases so that the dimensionality of within-class matrix can be reduced to l_1, and the within-class matrix is nonsingular in the new space.

This theorem tells us that if we want to use PCA to reduce the dimensionality, some eigenvectors corresponding to the smaller eigenvalues of S_t may be selected.

IV. GA-PCA: Selection of Principal Components Using Genetic Algorithm

As discussed in the previous section, we need to find a subset of principal components to reduce the dimension. Although Remark 3 shows that we can reduce the within-class scatter matrix to any dimension smaller than l (=rank(S_w)), this paper would like to reduce its dimensionality to l because we believe that less dimensionality may lose some discriminant information. The most popular method is to select the largest l principal components (also analyzed in Remark 2). However, PCA-DRT Theorem shows that the selection may be not unique. We can add some smaller principal components while removing some larger principal components. Small principal components may contain useful information for recognition.

The Genetic Algorithm (GA) has been widely used in the pattern recognition, feature selection [11], [12] and face recognition [14]. A survey of genetic algorithm has been published in [8]. In this section, we propose a methodology to use the Genetic Algorithm (GA) to select l out of m (=rank(S_t)) principal components of S_t as the bases for dimension reduction based on PCA-DRT theorem, called *GA-PCA*. GA-PCA is driven by a fitness function in term of generalization ability, performance accuracy, and the penalty item. Details are discussed as follows.

A. Chromosome Representation

We use one bit to indicate whether the principal component is selected. Let $a_1a_2\ldots a_m$ be the chromosome representation and $a_i, i = 1, 2, \ldots, m$, taking value of 0 or 1. If $a_i = 1$, the ith principal component is selected; otherwise, it is not. Thus, the length of the chromosome is m.

B. Genetic Operators

GA finds the solution via some operators driven by a fitness function. The operators are selection, crossover, and mutation. In this paper, we use the following.

1) *Mixture selection–Building up the mating set by a two-step selection:* We first retain some best chromosomes from the parent group, and then, a proportionate selection process is implemented to select the others.
2) *Two-point crossover:* Pair the chromosomes in the mating set stochastically; then, randomly select two crossover points and implement the exchange procedure between the crossover points. However, the exchange procedure is not simply exchanging their genetic information between the crossover points. Based on the idea similar to [17], a process of random selection between the crossover points is implemented, while guaranteeing that the number of selections between the crossover points of each chromosome retained no change. The specific model is designed as follows.

Suppose $a_1a_2a_3\ldots a_m$ and $b_1b_2b_3\ldots b_m$ are two chromosomes. Let a_{g_1} and a_{g_y} are two crossover points for b_{g_1} and b_{g_y}, respectively. We can classify the pair (a_i, b_i), where $g_1 \le i \le g_y$, into 2 sets, i.e., $\{(a_i, b_i)|g_1 \le i \le g_y\} = \Delta_1 \cup \Delta_2$, where

$$\begin{aligned}\Delta_1 &= \{(a_i, b_i)\,|\, a_i = b_i, g_1 \le i \le g_y\}\\ &= \{(a_{s_1}, b_{s_1}), \ldots, (a_{s_r}, b_{s_r})\}\\ \Delta_2 &= \{(a_i, b_i)|a_i \ne b_i, g_1 \le i \le g_y\}\\ &= \{(a_{sz_1}, b_{sz_1}), \ldots, (a_{sz_c}, b_{sz_c})\}.\end{aligned}$$

Without loss of generality, we assume that $sz_1 = g_1, sz_c = g_y$. Fig. 2(a) shows the parts of a pair chromosomes that are located between the crossover points before the crossover operation.

Let $n_a = \sum_i a_{sz_i}, n_b = \sum_i b_{sz_i}$. It is obvious that $a_{sz_1} + b_{sz_1} = 1, \ldots, a_{sz_c} + b_{sz_c} = 1$ and $n_a + n_b = c$.

Fig. 2(b) shows the parts of a pair of chromosome corresponding to (a) after the crossover operation and (b) is obtained as follows:

We first retain the pair bits in set Δ_1 in the same place respectively, i.e., $\bar{a}_{s_i} = a_{s_i}, \bar{b}_{s_i} = b_{s_i}, i = 1, 2, \ldots, r$, and let the other bits be 0, i.e., $\bar{a}_{sz_i} = \bar{b}_{sz_i} = 0, i = 1, 2, \ldots, c$.

Second, randomly select a subset $\{sz_{j_1}, \ldots, sz_{j_{n_a}}\}$ from the set $\{sz_1, \ldots, sz_c\}$, and let $\{sz_{j_{(n_a+1)}}, \ldots, sz_{j_c}\} = \{sz_1, \ldots, sz_c\} - \{sz_{j_1}, \ldots, sz_{j_{n_a}}\}$, where $j_i \ne j_k, \forall i \ne k$.

Finally, let $\bar{a}_{sz_{j_1}} = 1, \bar{a}_{sz_{j_2}} = 1, \ldots, \bar{a}_{sz_{j_{n_a}}} = 1, \bar{b}_{sz_{j_{(n_a+1)}}} = 1, \bar{b}_{sz_{j_{(n_a+2)}}} = 1, \ldots, \bar{b}_{sz_{j_c}} = 1$.

It is clear that selection operator operates on the set Δ_2 only.

The key idea is that we want to retain the number of selected principal components in each section of the chromosome between the crossover points so that the crossover operator does not change the number of selected principal components in any chromosome.

3) *Adaptive probability mutation:* Each bit of the chromosome would be changed by the decision on the mutation rate, where the mutation rate is adaptive. We change the bit by changing from 1 to 0 or vice versa. Once we remove or add one principal component, we randomly add or remove a different one so that the total number of selections remains unchanged.

C. Fitness Function

Given a set of principal components $w_{i_1}, \ldots, w_{i_l}$, denote

$$P = (w_{i_1}, \ldots, w_{i_l}) \tag{9}$$

and $\lambda_{i_1}, \ldots, \lambda_{i_l}$ as the eigenvalues corresponding to $w_{i_1}, \ldots, w_{i_l}$. Denote

$$\Lambda_P = \text{diag}(\lambda_{i_1}, \ldots, \lambda_{i_l}) \tag{10}$$

Fitness function plays a crucial role in choosing offspring for the next generation from the current generation. It evaluates the fitness values of chromosomes, which are important for GA to select the better ones at each generation. Note that the PCA-DRT theorem is a fundamental theorem of the principal component selection. Based on PCA-DRT, the fitness function for principal component selection is defined as (11), shown at the bottom of the page, where $F_g(P)$ is the generalization term, $F_a(P)$ is the performance accuracy term, and $F_c(P)$ is the penalty term. $F_g(P)$ serves as the scatter measurement

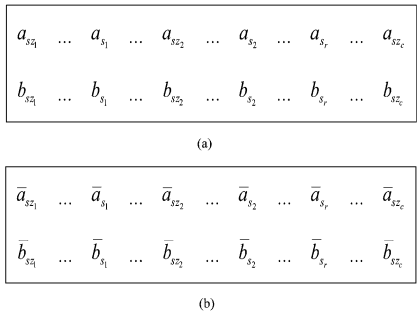


Fig. 2. Parts of a pair chromosome that locate between the crossover points. (a) Before crossover. (b) After crossover.

of different classes; $F_a(P)$ serves as performance indicator of the recognition system, and $F_c(P)$ is a penalty function taken as a constraint on the selection. Generally speaking, $F_g(P)$ aims to select the principal components that have better generalization at the dimension reduction step. For the sake of the fact that the subjects outside the training set are unknown in the training procedure, the value of $F_a(P)$ is based on the training set and provides an evaluation of the output of the specific system. Therefore, $F(P)$ is a trade-off function among $F_g(P)$, $F_a(P)$ and $F_c(P)$. In our experiment, the specific forms of $F_g(P)$, $F_a(P)$, and $F_c(P)$ are designed below, provided that P satisfies the PCA-DRT theorem.

As PCA is a decorrelation technique, the mutual relationship between each principal component is statistically uncorrelated. Hence, to evaluate $F_g(P)$, we consider combinations of different principal components on the generalization.

Let x be a face image. After dimension reduction, x becomes $\bar{x} = P^T x$; the mean image of class C_j becomes $\bar{u}_j = P^T u_j$; the mean image of all samples becomes $\bar{u} = P^T u$, where we still use $\{C_j\}_{j=1}^K$ as the symbol of the set of classes after dimension reduction. Let

$$\Sigma = E\{(\bar{x}-\bar{u})(\bar{x}-\bar{u})^T\} = \frac{1}{N}\sum_{j=1}^{K}\sum_{\bar{x}\in C_j}(\bar{x}-\bar{u})(\bar{x}-\bar{u})^T. \quad (12)$$

It is pointed out that Mahalanobis distance performs better in the standard L_2 norm [10]. To measure the generalization ability, the distance between $\bar{u}_j$ and $\bar{u}$ is defined by

$$d_j = \sqrt{(\bar{u}_j - \bar{u})^T \Sigma^{-1} (\bar{u}_j - \bar{u})}. \quad (13)$$

Taking the Mahalanobis distance, $F_g(P)$ is designed as

$$F_g(P) = \min_{j=1,2,\ldots,K}\{d_j\}. \quad (14)$$

However, selecting too many smallest ones may be risky in that $P^T S_w P$ is still singular after dimension reduction. This scenario is discussed in Appendix C. Therefore, if $F_g(P_1)$ and $F_g(P_2)$ are the same, we would like to choose the one that contains more larger principal components, where P_1, P_2 are two candidates of selected principal components.

In this case, a penalty function is needed. Let

$$\mathrm{SE}(P) = \left(\Sigma_{k=1}^{l}\lambda_{i_k}\right) / \left(\Sigma_{j=1}^{m}\lambda_j\right) \quad (15)$$

Then, we define the penalty function $F_c(P)$ as

$$F_c(P) = e^{a\times(\mathrm{SE}(P)-b)}, \quad a \geq 0, \quad 0 \leq b \leq 1 \quad (16)$$

where b is the threshold of $\mathrm{SE}(P)$, and a is the parameter that may affect the convergence of the GA and the weight of $F_c(P)$. If a is too large, the GA will be premature and may not get the optimal result; it will also increase the weight of $F_c(P)$ in (11). If a is too small, it results in a longer computational time to get the optimal result, and $F_c(P)$ will play a minor role.

The rest is to design $F_a(P)$. Generally speaking, from (11), the value of $F_a(P)$ evaluates the output of complete system, based on the training set. In this paper, GA-PCA is used as a dimension reduction procedure for LDA (see a detailed description in Section V). Therefore, $F_a(P)$ can be interpreted as an evaluation function of the LDA performance. In this paper, we found that the LDA-based recognition system is well trained, which means that for both databases (FERET and CMU PIE), the recognition performance for images inside training set is almost 100%. In order to reduce the computational burden and simplify the procedure, we set $F_a(P) = 1$ for all experiments in this paper. However, it must be pointed out that $F_a(P)$ may be a variable for other applications that employ GA-PCA. Finally, the optimal principal components selection evolves from

$$P_{\mathrm{opt}} = \arg\max_P\{F(P)\}. \quad (17)$$

V. GA-Fisher

A. What Is GA-Fisher?

By virtue of the PCA-DRT Theorem and the GA-PCA algorithm, we have selected a set of principal components for dimension reduction. In this section, we combine the selected eigenvalues/eigenvectors with a whitening procedure to develop a GA-Fisher method for face recognition. Moreover, we have investigated that results of the GA-PCA algorithm would facilitate a fast calculation for LDA after dimension reduction. Details are discussed as follows.

The GA-Fisher algorithm is described as follows.

1) Calculate the eigenvectors corresponding to m largest eigenvalues of S_t, where $m = \mathrm{rank}(S_t)$. Suppose the eigenvectors are $w_1, w_2, \ldots, w_m$ and their corresponding eigenvalues are $\lambda_1, \lambda_2, \ldots, \lambda_m$.
2) Use GA-PCA discussed in Section IV to select l (=rank(S_w)) principal components. Suppose $w_{t_1}, w_{t_2}, \ldots, w_{t_l}$ are finally selected and

$$F(P) = \begin{cases} F_g(P) \times F_a(P) \times F_c(P), & \text{if } P \text{ satisfies theorem PCA-DRT} \\ 0, & \text{otherwise} \end{cases} \quad (11)$$

$\lambda_{t_1}, \lambda_{t_2}, \ldots, \lambda_{t_l}$ are eigenvalues corresponding to them. Let $W_{\mathrm{ga-pca}} = (w_{t_1}, w_{t_2}, \ldots, w_{t_l}), \Lambda_{\mathrm{ga-pca}}^{-1/2} = \mathrm{diag}(\lambda_{t_1}^{-1/2}, \lambda_{t_2}^{-1/2}, \ldots, \lambda_{t_l}^{-1/2})$.

3) Process a whitening procedure, i.e.,

$$W_{\mathrm{ga-pca-whiten}} = W_{\mathrm{ga-pca}} \Lambda_{\mathrm{ga-pca}}^{\frac{-1}{2}}.$$

4) Compute

$$W_{\mathrm{lda}} = \arg\max_W \frac{\left|W^T W_{\mathrm{ga-pca-whiten}}^T S_b W_{\mathrm{ga-pca-whiten}} W\right|}{\left|W^T W_{\mathrm{ga-pca-whiten}}^T S_w W_{\mathrm{ga-pca-whiten}} W\right|}.$$

5) Let $W_{\mathrm{opt}}^T = W_{\mathrm{lda}}^T W_{\mathrm{ga-pca-whiten}}^T$. Then, W_{opt} is the optimal projection for GA-Fisher.

B. Whitening Procedure

1) Why Is a Whitening Procedure Added?: From the Mahalanobis distance between $\bar{u}_j$ and $\bar{u}$ defined in Section IV, we have

$$\begin{aligned}
\Sigma &= E\{(\bar{x}-\bar{u})(\bar{x}-\bar{u})^T\} = E\{P^T(x-u)(x-u)^T P\} \\
&= P^T S_t P = \Lambda_P \\
d_j^2 &= (\bar{u}_j - \bar{u})^T \Sigma^{-1} (\bar{u}_j - \bar{u}) \\
&= ((u_j-u)^T P) \Sigma^{-1} (P^T(u_j-u)) \\
&= ((u_j-u)^T \left(P\Lambda_p^{\frac{-1}{2}}\right)\left(\Lambda_p^{\frac{-1}{2}} P^T\right)(u_j-u)) \\
&= \left(\left(P\Lambda_p^{\frac{-1}{2}}\right)^T (u_j-u)\right)^T \left(\left(P\Lambda_p^{\frac{-1}{2}}\right)^T (u_j-u)\right) \\
&= (\tilde{P}^T u_j - \tilde{P}^T u)^T (\tilde{P}^T u_j - \tilde{P}^T u)
\end{aligned}$$

where $\tilde{P} = P\Lambda_p^{-1/2}$.

Hence, the whitening procedure $\tilde{P} = P\Lambda_p^{-1/2}$ is automatically included if Mahalanobis distance is employed.

2) Fast Computation on LDA Procedure: This section reveals that GA-PCA can facilitate a fast algorithm for GA-Fisher in calculating the W_{lda} in step 4 in Section V with the whitening procedure.

For convenience, we redefine $W_{\mathrm{pca}} = W_{\mathrm{ga-pca}}$, and let $\Lambda_{\mathrm{pca}}^{-1/2} = \Lambda_{\mathrm{ga-pca}}^{-1/2}, S_w^* = \Lambda_{\mathrm{pca}}^{-1/2} W_{\mathrm{pca}}^T S_w W_{\mathrm{pca}} \Lambda_{\mathrm{pca}}^{-1/2}$, and $S_b^* = \Lambda_{\mathrm{pca}}^{-1/2} W_{\mathrm{pca}}^T S_b W_{\mathrm{pca}} \Lambda_{\mathrm{pca}}^{-1/2}$.

These notations are also used in the following part and in Appendix B.

It is obvious that S_w^* is nonsingular by the PCA-DRT Theorem. In the traditional LDA procedure, we need to calculate the LDA transformation matrix $W_{\mathrm{lda}} = (w_{w_1}, w_{w_2}, \ldots, w_{w_q})$ via $W_{\mathrm{lda}} = \arg\max_W(|W^T S_b^* W|)/(|W^T S_w^* W|)$, i.e., $S_w^{*-1} S_b^* W_{\mathrm{lda}} = W_{\mathrm{lda}} \bar{\Lambda}_{\mathrm{lda}}$, where $\bar{\Lambda}_{\mathrm{lda}} = \mathrm{diag}(\lambda_{\mathrm{lda}_1}, \lambda_{\mathrm{lda}_2}, \ldots, \lambda_{\mathrm{lda}_q}), \lambda_{\mathrm{lda}_i} \neq 0, i = 1, 2, \ldots, q, q \leq K-1$.

However, we propose to do it in another way using the following theorem.

Theorem Fast Computation for LDA: Suppose w is the eigenvector of $S_w^{*-1} S_b^*$ corresponding to the eigenvalue λ, then it is a sufficient and necessary condition that w is the eigenvector of S_w^* corresponding to the eigenvalue $1/(\lambda+1)$, and if $\lambda \neq 0$ then $1/(\lambda+1)$ takes value in (0, 1).

To proof the Theorem, we need the following propositions.

Proposition 1:

1) If any w is the eigenvector of $S_w^{*-1} S_b^*$ corresponding to the eigenvalue λ, then w is exactly the eigenvector of S_w^{*-1} corresponding to the eigenvalue $\lambda+1$.
2) If any w is the eigenvector of S_w^{*-1} corresponding to the eigenvalue λ, then w is exactly the eigenvector of $S_w^{*-1} S_b^*$ corresponding to the eigenvalue $\lambda-1$.

The proof of the Proposition 1 is given in Appendix B.

Now let us prove Fast Computation for LDA Theorem. By Lemma 4 in Appendix B, any eigenvalue of $S_w^{*-1} S_b^*$ is not less than zero. Therefore by using Proposition 1, we can conclude that any eigenvector of $S_w^{*-1} S_b^*$ corresponding to the eigenvalue being larger than 0 is the same eigenvector of S_w^{*-1} corresponding to the eigenvalue being larger than 1.

Since S_w^* is nonsingular, S_w^* can be decomposed into $S_w^* = U_w^* \Lambda_w^* U_w^{*T}$ with U_w^* as a positive definite matrix, and $|\Lambda_w^*| > 0$. Hence, we get $S_w^{*-1} = U_w^* \Lambda_w^{*-1} U_w^{*T}$.

Comparing $S_w^* U_w^* = U_w^* \Lambda_w^*$ with $S_w^{*-1} U_w^* = U_w^* \Lambda_w^{*-1}$, we get the following conclusion.

Proposition 2:

1) If any w is the eigenvector of S_w^{*-1} corresponding to the eigenvalue λ, then w is exactly the eigenvector of S_w^* corresponding to the eigenvalue $1/\lambda$.
2) If any w is the eigenvector of S_w^* corresponding to the eigenvalue λ, then w is exactly the eigenvector of S_w^{*-1} corresponding to the eigenvalue $1/\lambda$.

Combining Propositions 1 and 2, the Fast Computation for the LDA Theorem is obvious.

From now on, we can solve

$$W_{\mathrm{lda}} = \arg\max_W \frac{\left|W^T W_{\mathrm{ga-pca-whiten}}^T S_b W_{\mathrm{ga-pca-whiten}} W\right|}{\left|W^T W_{\mathrm{ga-pca-whiten}}^T S_w W_{\mathrm{ga-pca-whiten}} W\right|}$$

by calculating the eigenvectors of S_w^* corresponding to the eigenvalues in (0, 1).

C. Face Recognition System Using GA-Fisher

Using the GA-Fisher method described above, a complete system is developed. The block diagram of the GA-Fisher face recognition system is shown in Fig. 3. First of all, given a set of training images, PCA is employed, and then, we get a set of principal components. Applying the GA-PCA algorithm reported in Section IV, a selected set of principal components and the corresponding eigenvectors are obtained. Then, we can perform the dimensionality reduction with the whitening transformation. Finally, based on the description in Section V, we can determine the optimal projection matrix of GA-Fisher for face recognition.

VI. Further Interpretation on Fisherface

In Section III, Remark 2 of the PCA-DRT Theorem points out that we need experience or extra information in selecting

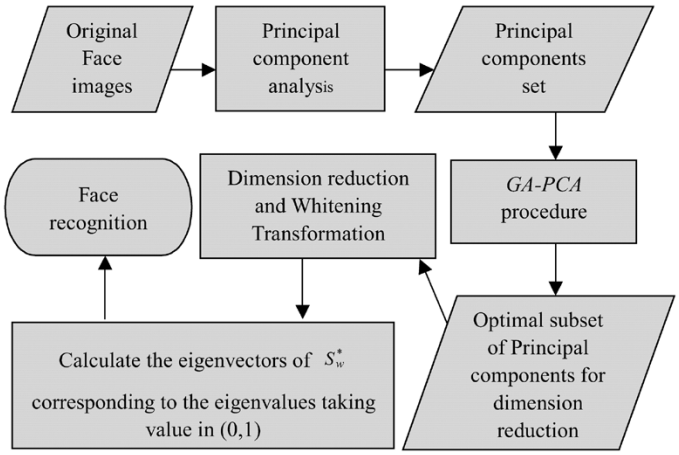


Fig. 3. Face recognition using *GA-Fisher*.

the l largest eigenvectors of S_t for dimensionality reduction in Fisherface. Below, an additional interpretation is provided.

From the analysis in Section IV, as we set $F_a(P) = 1$ in our experiment, the fitness function of GA-PCA for LDA becomes $F_g(P) \times F_c(P)$, if P satisfies the PCA-DRT Theorem, and $F_c(P) = e^{a\times(\mathrm{SE}(P)-b)}$. $F_g(P)$ is bounded. From Section V, we get

$$\begin{aligned} d_j^2 &= (\bar{u}_j - \bar{u})^T \Sigma^{-1} (\bar{u}_j - \bar{u}) \\ &= (\tilde{P}^T u_j - \tilde{P}^T u)^T (\tilde{P}^T u_j - \tilde{P}^T u) \end{aligned}$$

where $\tilde{P} = P\Lambda_p^{-1/2}$, and

$$\tilde{P}^T S_t \tilde{P} = \tilde{P}^T S_w \tilde{P} + \tilde{P}^T S_b \tilde{P} \Rightarrow I_{l\times l} = \tilde{P}^T S_w \tilde{P} + \tilde{P}^T S_b \tilde{P}.$$

Using (2), we get

$$\tilde{P}^T S_b \tilde{P} = \frac{1}{N} \sum_{j=1}^{K} N_j (\tilde{P}^T u_j - \tilde{P}^T u)(\tilde{P}^T u_j - \tilde{P}^T u)^T.$$

Therefore

$$\begin{aligned} (F_g(P))^2 &\le \sum_j d_j^2 \le \max_j (N/N_j) \times \mathrm{trace}(I_{l\times l}) \\ &= l \times \max_j (N/N_j). \end{aligned}$$

Therefore, if a is large enough and b tends to zero, the value $F_c(P)$ dominates in the fitness function. In that case, GA−PCA+LDA becomes Fisherface if P satisfies Theorem PCA-DRT. Therefore, Fisherface is a special case of GA−PCA+LDA. This further explains why the Fisherface selects the largest principal components of S_t as a basis for dimension reduction.

VII. Experimental Results

The results presented in this section are divided into three parts. First, we will use FERET database with smaller dataset to demonstrate the procedures and details of the proposed GA-PCA and GA-Fisher algorithms. We also show the probability curve showing that selecting smaller principal components always happens in GA-PCA. In the second part, we will evaluate the performance of GA-Fisher using FERET database with larger dataset. Finally, we will use CMU PIE database to evaluate the GA-Fisher performance for pose and illumination variations.

Two widely accepted measurements, namely *Cumulative Match Characteristic* (CMC) and *Receiver Operating Characteristic* (ROC), are used for evaluation.

A. Database and Parameter Setting

1) FERET Database: The FERET database will be used in the first two experiments. In the first experiment, in order to illustrate the procedure of our method, we select a smaller dataset, which includes 72 people and six images for each individual. In the second experiment, we have a larger dataset with 255 individuals, and each person has four frontal images with different illuminations, face expressions, ages, and they are with or without glasses [9]. All images are extracted from four different sets, namely, Fa, Fb, Fc, and duplicate [9]. All images are aligned by the centers of eyes and mouth, which are given in the database and then normalized with the resolution 92×112. Some images of the larger subset of the FERET database are shown in Fig. 4.

2) CMU PIE Database: The CMU Pose, Illumination, and Expression (PIE) [18], [24] database consists of 41 368 images of 68 people. Each person has images captured under 13 different poses and 43 different illumination conditions and with four different expressions. In this paper, we classify the images into five subsets based on their poses, namely, fontal view, half right profile, half left profile, full right profile, and full left profile, with background (neutral illumination) turned off or on, as shown in Table I. The images of these subsets come from cameras that have flashed, except the one with neutral illumination. The images are normalized to the resolution of 92 by 112. Fig. 5 shows some of the images in each subset.

3) Parameters Setting: The parameters of the fitness function and G.A. algorithm used in this paper are as follows.

1) $F_c(P) = e^{20\times(\mathrm{SE}(P)-0.99)}$, i.e., $a = 20, b = 0.99$. The values of a and b are determined experimentally.
2) Let the crossover rate be 80% and the mutation rate be 23% at the beginning; then, they would decrease gradually and finally reach at 70% and 3%, respectively.

Fig. 4. Some images from larger subset of FERET database.

Fig. 5. Some Images of each subset from CMU PIE database. (a) Subset-1. (b) Subset-2. (c) Subset-3. (d) Subset-4. (e) Subset-5.

3) In order to obtain the stable performance of GA, the population of the GA algorithm and its generation are 200 and 400, respectively.

B. Part I: Experiment on FERET Database With Small Dataset

This section mainly aims to demonstrate the procedures and details of the proposed GA-PCA. We will present the probability of principal components selection. Two methods, namely, Fisherface and GA-Fisher, are selected for evaluation and comparison. The experiment is performed as follows. Three images per person are randomly selected for training, whereas the rest of images are used for testing. This experiment is repeated with ten runs. Due to the limit of length, we plot ROC curves for the first run of the test. The average accuracy with five ranks and an ROC curve are shown in Fig. 6. Unless otherwise stated, in this paper, the false accept rate axis of the ROC is on a logarithmic scale [19].

By comparing the accuracy with *Fisherface (PCA+LDA)*, the effectiveness of GA-PCA can be seen. Employing the GA-PCA algorithm for selecting principal components, GA-Fisher can increase the accuracy from 87.08% to 89.63% at rank 1, and the ROC curve also shows the superior performance of GA-PCA. This results show that the proposed GA-PCA is an effective principal component selection algorithm for dimensionality reduction in LDA.

Moreover, we would like to report a statistic on the selection of principal components by GA-PCA. They are recorded from the ten runs with different training samples above. Fig. 7 shows the percentage of principal components' selected. The observations are as follows.

TABLE I
CONSTRUCTION OF SUBSETS OF CMU PIE

	Subset-1	Subset-2	Subset-3	Subset-4	Subset-5
Pose type	Frontal view	Half Right profile	Half Left profile	Full Right profile	Full Left profile
Background light	on and off	off	off	off	off
Number of images per person	43	21	21	21	21
Amount of Individuals	68	68	68	68	68

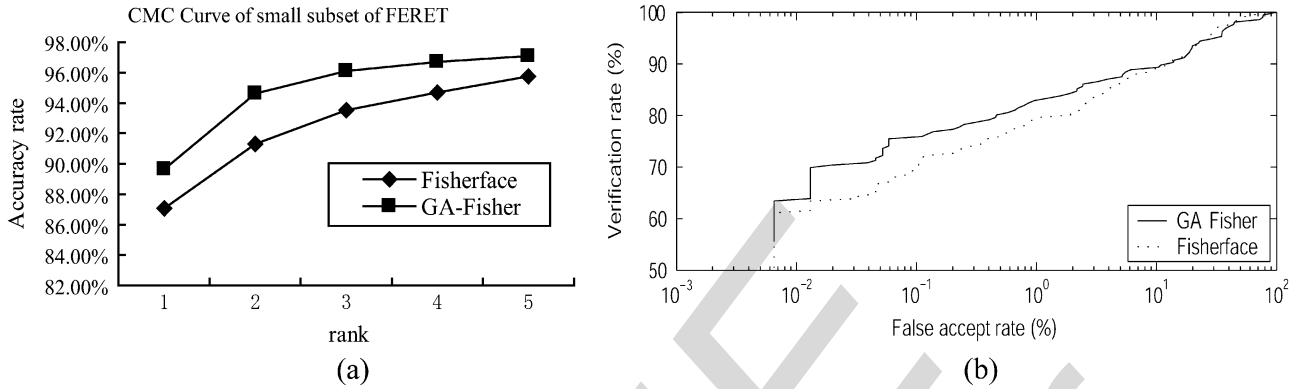


Fig. 6. Performance on small subset of FERET database. (a) Identification performance of GA-Fisher versus Fisherface. (b) Verification performance of GA-Fisher versus Fisherface. The false accept rate axis of ROC is on a logarithmic scale.

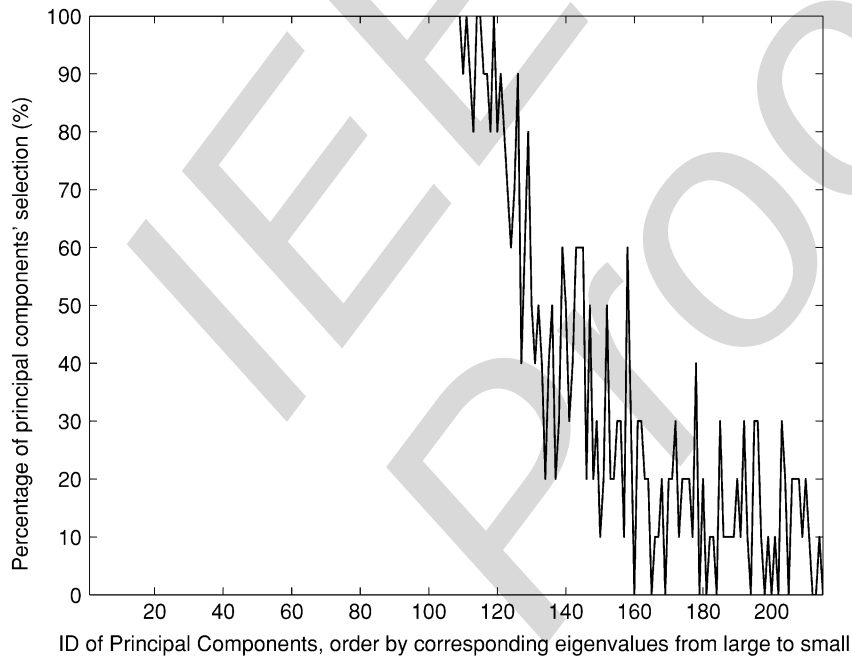


Fig. 7. Probability of principal components' selection during the experiment on small FERET database.

1) The largest principal components are always retained (please note that this dataset does not contain images with large illumination variations).
2) Around 20% of the cases select the smallest principal components. This concludes that we should not hard-code to remove all small principal components.

TABLE II
CMC SCORES OF IDENTIFICATION PERFORMANCE

Methods	Rank 1	Rank 2	Rank 3	Rank 4	Rank 5
GA-Fisher	81.49%	85.84%	87.29%	88.51%	89.37%
Fisherface	76.20%	80.55%	82.20%	83.69%	84.98%
Fisherface w/o 3	75.80%	79.45%	81.61%	82.98%	84.55%

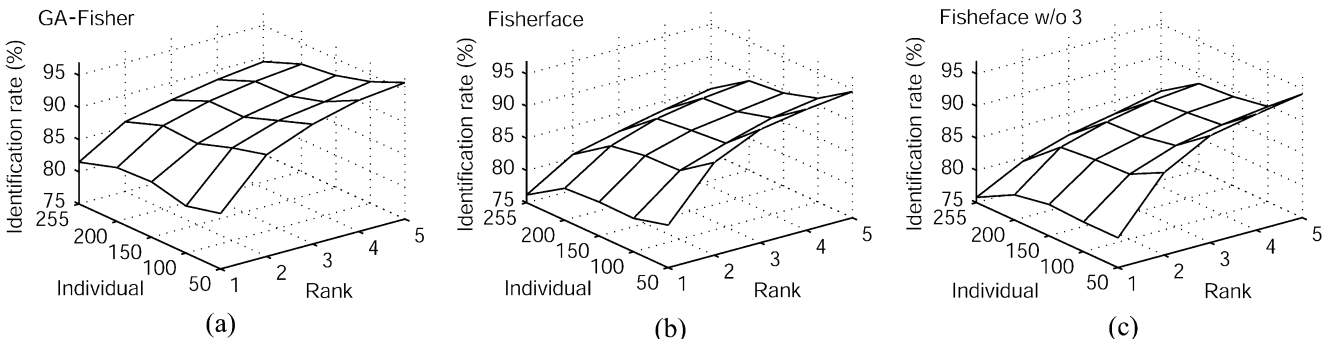


Fig. 8. Rank 1–5 identification performance as a function of database size. (a) GA-Fisher. (b) Fisherface. (c) Fisherface w/o 3 eigenvectors with the largest eigenvalues.

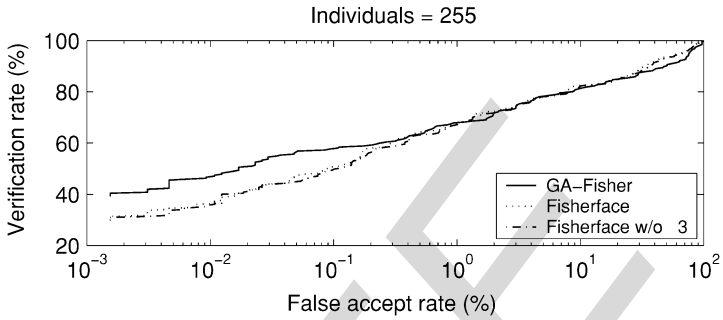


Fig. 9. ROC curves of three methods on the FERET database.

TABLE III
CMC (RANK ACCURACY) FOR SUBSET 1 TO SUBSET 5 ON CMU PIE DATABASE

Method	Subset-1		Subset-2		Subset-3		Subset-4		Subset-5	
	Rank 1	Rank 3	Rank 1	Rank 3	Rank 1	Rank 3	Rank 1	Rank 3	Rank 1	Rank 3
GA-Fisher	87.87%	92.19%	92.12%	95.38%	91.95%	94.40%	92.17%	94.93%	94.51%	96.81%
Fisherface	83.74%	88.54%	89.59%	92.12%	88.26%	91.74%	89.90%	92.89%	92.90%	95.63%
Fisherface w/o 3	84.70%	89.17%	90.29%	92.94%	89.24%	92.40%	90.82%	93.76%	93.65%	96.23%

C. Part II: Experiment on FERET Database With Larger Dataset

The FERET database with larger dataset (255 individuals, four images per person) is selected for evaluation of the proposed algorithm. In this experiment, we randomly select three images of each person for training, and the rest are used for testing. It is experimentally known that eliminating the first three largest principal components for eigenface (PCA w/o 3) is suggested to reduce the affect from illumination [7], [16]. In order to compare the methodology for dimension reduction in LDA, three methods, namely Fisherface, Fisherface w/o 3 (PCA w/o 3 + LDA), and GA-Fisher are selected for comparison. The experiment is repeated for ten runs, and the average rank accuracy is recorded and reported in Table II.

Table II shows that CMC Scores as the rank increases, and the same conclusion as Part I can be drawn. The proposed GA-Fisher algorithm gives the best result. GA-PCA performs better than that of the one using PCA w/o 3 as the methodology for dimensionality reduction.

Next, we would like to see the performance when the size of the database changes from 50 to 255. The experiment setting is the same as before. The results are shown in Fig. 8. It can be seen that both Fisherface and Fisherface w/o 3 have a more obviously descendent trend than GA-Fisher with respect to database size (or individual number). This concludes that GA-PCA selection processing is less sensitive to the number of individuals. Therefore, GA-PCA is an effective method for dimension reduction. In turn, the proposed GA-Fisher algorithm gives the best results.

Finally, we would like to evaluate the GA-Fisher algorithm using the Receiver Operating Characteristic (ROC). The results are plotted in Fig. 9.

Note that a lower false accept rate (FAR) may lead to a more secure system. From Fig. 9, GA-Fisher outperforms the other two methods when the FAR is smaller than 0.1%. For larger FAR, the performance of the proposed GA-Fisher gives an equally good performance than the other two methods. This may be due to the fact that the images variations in the FERET database are not very large.

D. Part III: Experiment on CMU PIE Database

The CMU PIE database is a challenging face database that consists of the pose and illumination variations. To perform the face recognition across different poses, one aspect is to determine what kind of the pose gesture first and then to do the recognition work based on the fixed pose [20]. In fact, many

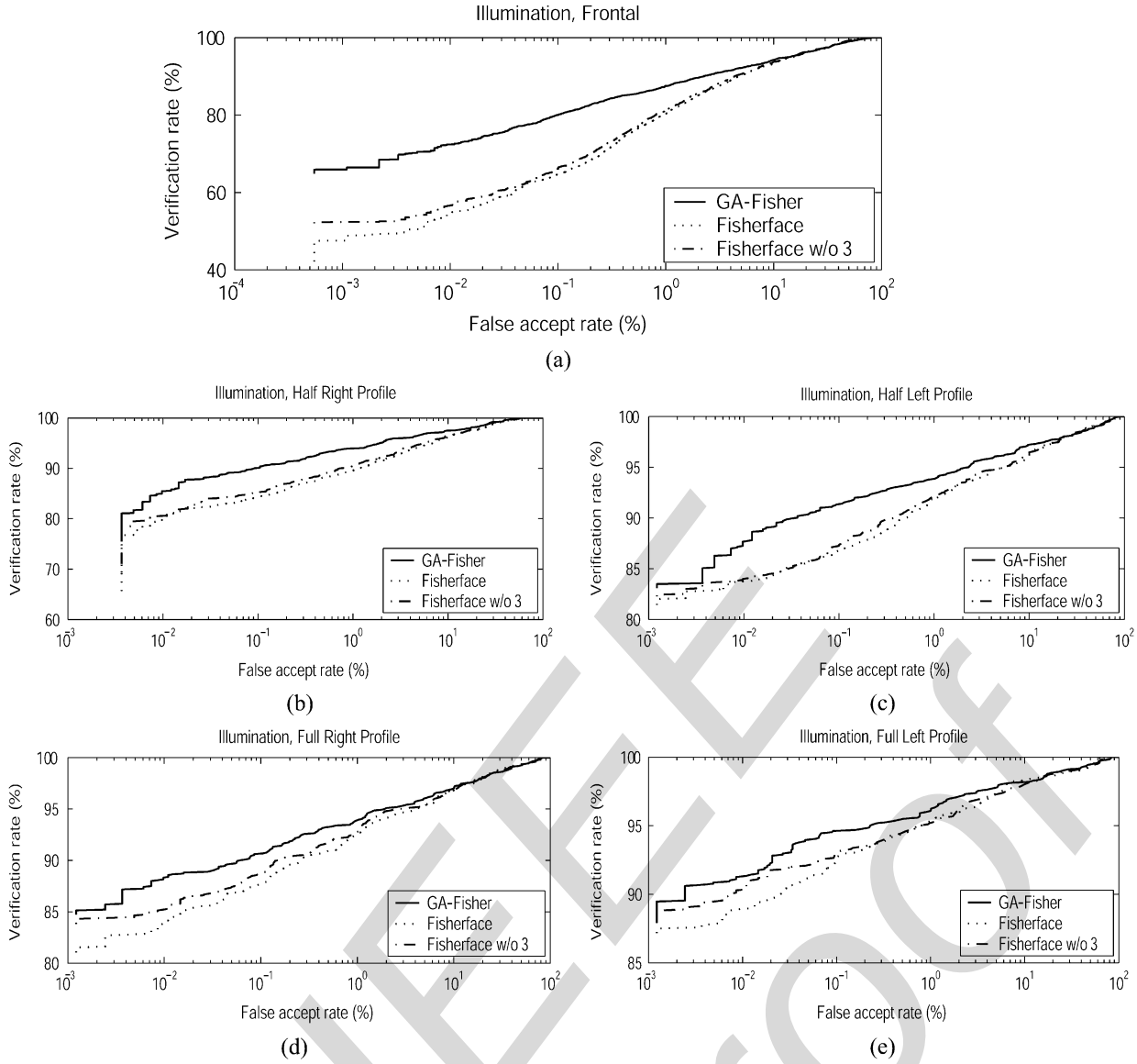


Fig. 10. ROC Curves of five subsets in CMU PIE database, where (a) to (e) refer to subset-1 to subset-5, respectively.

robust pose estimation algorithms, which are not the scope of this paper, have been developed. As described, the images in the CMU PIE database are divided into five subsets according to the pose variation, as this paper mainly focuses on the recognition part. For each subset, we randomly selected three images from each individual for training, and the rest of the images in that subset are used for testing. Again, this experiment is repeated for ten runs, and the average rank accuracy is recorded and tabulated in Table III. It can be seen from Table III that the proposed GA-Fisher gives the best performance in all five subsets. On the average, the proposed GA-Fisher gives 2% to 3% improvement on both Fisherface and Fisherface w/o 3.

Finally, we would like to see the performance of the GA-Fisher from the ROC curve. The ROC curve of each subset of images are calculated and plotted in Figs. 10(a) to (e). It can be seen that the proposed GA-Fisher gives the best performance in most cases.

VIII. Conclusion

This paper performs a comprehensive investigation into the principal component selection of S_t for dimensionality reduction. We have proposed a Principal Component Analysis Dimension Reduction Theory (PCA-DRT) to address the problem as to why the principal components can be used to reduce the dimensionality while guaranteeing the within-scatter matrix to be nonsingular after the transformation. This paper also points out that some smaller principal components are useful, and some larger principal components can be removed. Along this line, we propose a new principal component selection algorithm called GA-PCA. Integrating the GA-PCA with a whitening transformation into LDA, a new LDA, called the GA-Fisher (GA-PCA+Whitening Transformation+LDA) has been developed. GA-Fisher reveals the importance of dimensionality reduction before LDA in the case of small sample size.

GA-Fisher offers two additional advantages over existing LDA algorithms in solving the small sample size problem. One is the optimal basis for dimensionality reduction derived from GA-PCA. Another is the optimization of the calculation of the features by the LDA procedure after dimensionality reduction. Experimental results of both identification performance and verification performance on FERET and CMU PIE show that GA-Fisher performs best in most cases.

Nevertheless, a new framework for principal components selection for dimension reduction in LDA has been set up. GA-PCA has been demonstrated to be an efficient algorithm for principal component selection. Even if we have lots of (sufficient) samples, dimensionality reduction is also needed if we cannot afford the high computation due to high dimensionality.

APPENDIX

A. Proof of PCA-DRT Theorem

We need two lemmas in order to prove the theorem. Lemma 1 is obvious. Huang [13] has proved Lemma 2.

Lemma 1: Suppose matrix $A = U\Lambda V^T$, where Λ is a diagonal matrix, $|\Lambda| \neq 0$, and $U^TU = I, V^TV = I$. Then, $\text{rank}(A) = \text{rank}(\Lambda) = \text{rank}(U) = \text{rank}(V)$.

Lemma 2: $N(S_t) = N(S_w) \cap N(S_b)$, where $N(S_t)$ is the nullspace of S_t, $N(S_w)$ is the nullspace of S_w, and $N(S_b)$ is the nullspace of S_b [13].

Proof of PCA-DRT Theorem: From Section II, $S_w = \Phi_w\Phi_w^T, \Phi_w \in R^{n\times N}$. Applying the singular value decomposition on Φ_w, we get that $\Phi_w = U_w\Lambda_wV_w^T$ satisfies $|\Lambda_w| \neq 0$, where $U_w \in R^{n\times l}, V_w \in R^{N\times l}, \Lambda_w \in R^{l\times l}, U_w^TU_w = I, V_w^TV_w = I$ ($n > N$ in the small sample size problem case). Therefore, $S_w = U_w\Lambda_w^2U_w^T, \Lambda_w^2 = \text{diag}(\lambda_{w_1}, \ldots, \lambda_{w_l}), \lambda_{w_i} > 0, i = 1, \ldots, l$, i.e., $S_wU_w = U_w\Lambda_w^2$.

Define $\widehat{S}_w = W_t^TS_wW_t$; then, $\widehat{S}_w = \hat{\Phi}_w\hat{\Phi}_w^T$, where $\hat{\Phi}_w = W_t^TU_w\Lambda_w$.

Next, we need to prove that $\text{rank}(\widehat{S}_w) = \text{rank}(S_w)$. If it is true, the proof of the PCA-DRT Theorem is straightforward.

Since $S_w = U_w\Lambda_w^2U_w^T$ and $\widehat{S}_w = \hat{\Phi}_w\hat{\Phi}_w^T$, we have the following relations:

$$\text{rank}(S_w) = \text{rank}(\Lambda_w^2) \tag{A.1}$$

$$\text{rank}(\widehat{S}_w) = \text{rank}(\hat{\Phi}_w) \tag{A.2}$$

It is obvious that if $\text{rank}(\hat{\Phi}_w) = \text{rank}(\Lambda_w)$, then $\text{rank}(\widehat{S}_w) = \text{rank}(S_w)$.

Furthermore, $\Phi_w = U_w\Lambda_wV_w^T$; then

$$\text{rank}(U_w) = \text{rank}(\Lambda_w). \tag{A.3}$$

In addition, we know that $\hat{\Phi}_w = W_t^TU_w\Lambda_w, W_t^TU_w \in R^{m\times l}, m \geq l$. We conclude that

$$\text{rank}\big(W_t^TU_w\big) = \text{rank}\big(W_t^TU_w\Lambda_w\big) = \text{rank}(\hat{\Phi}_w). \tag{A.4}$$

According to (A.1)–(A.4), in order to prove $\text{rank}(\widehat{S}_w) = \text{rank}(S_w)$, we need to prove

$$\text{rank}\big(W_t^TU_w\big) = \text{rank}(U_w). \tag{A.5}$$

To prove (A.5), suppose $U_w = (\hat{u}_1, \ldots, \hat{u}_l)$; then, $W_t^TU_w = (W_t^T\,\hat{u}_1, \ldots, W_t^T\,\hat{u}_l) \in R^{m\times l}$. Then, $\{W_t^T\hat{u}_i\}_{i=1}^l$ will be a set of linearly independent vectors. If not, there must exist a set $\{\alpha_i\}$ such that $\sum_{i=1}^l(\alpha_iW_t^T\hat{u}_i) = 0$, where $\alpha_i \in R, i = 1, 2, \ldots, l$ with some α_i nonzero. Then, defining $\bar{\hat{u}} = \sum_{i=1}^l(\alpha_i\hat{u}_i)$, we have $W_t^T\bar{\hat{u}} = W_t^T\sum_{i=1}^l(\alpha_i\hat{u}_i) = 0$.

Moreover, $S_t\bar{\hat{u}} = W_t\Lambda_tW_t^T\bar{\hat{u}} = W_t\Lambda_t0 = 0$, i.e., $\bar{\hat{u}} \in N(S_t)$. By virtue of Lemma 2, we obtain $\bar{\hat{u}} \in N(S_w)$, i.e., $S_w\bar{\hat{u}} = 0$.

However, $S_w\bar{\hat{u}} = S_w(\sum_{i=1}^l(\alpha_i\hat{u}_i)) = \sum_{i=1}^l\alpha_i(S_w\hat{u}_i) = \sum_{i=1}^l(\alpha_i\lambda_{w_i})\hat{u}_i = 0$. Since $U_w^TU_w = I, \{\hat{u}_i\}_{i=1}^l$ is a set of linearly independent vectors, $\alpha_i\lambda_{w_i} \equiv 0$ for any i. However, it is known that $\lambda_{w_i} > 0, i = 1, 2, \ldots, l$; therefore, $\alpha_i \equiv 0, i = 1, 2, \ldots, l$, which is a contradiction.

Hence, $\text{rank}(W_t^TU_w) = l = \text{rank}(U_w)$; then, $\text{rank}(\widehat{S}_w) = \text{rank}(S_w)$ by (A.1)–(A.5).

Finally, $S_w^{\#}$ is derived from the following:

$$\text{Denote } W_t^T = \begin{pmatrix} w_1^T \\ \vdots \\ w_m^T \end{pmatrix}, \quad W_t^TU_w = \begin{pmatrix} w_1^TU_w \\ \vdots \\ w_m^TU_w \end{pmatrix} \in R^{m\times l}$$

since $m \geq l$; therefore, $\text{rank}(\{w_i^TU_w\}_{i=1}^m) = \text{rank}(W_t^TU_w) = l$. Then, there exists $\{w_{i_1}^TU_w, \ldots, w_{i_l}^TU_w\}$ having $i_j \leq m, j = 1, 2, \ldots, l$, and $i_h \neq i_j$ for any $h \neq j$, such that $\text{rank}(\{w_{i_j}^TU_w\}_{j=1}^l) = l$.

Thus

$$\hat{W}_t = (w_{i_1}, \ldots, w_{i_l})$$

$$\hat{W}_t^TU_w = \begin{pmatrix} w_{i_1}^TU_w \\ \vdots \\ w_{i_l}^TU_w \end{pmatrix} \in R^{l\times l}$$

$$\text{rank}\Big(\hat{W}_t^TU_w\Big) = l$$

i.e., $|\hat{W}_t^TU_w| \neq 0$.

Denote $S_w^{\#} = \hat{W}_t^TS_w\hat{W}_t = (\hat{W}_t^TU_w)\Lambda_w^2(\hat{W}_t^TU_w)^T \in R^{l\times l}$. Then, $|S_w^{\#}| = |\hat{W}_t^TU_w||\Lambda_w^2||\hat{W}_t^TU_w| \neq 0$, therefore $\text{rank}(S_w^{\#}) = l = \text{rank}(S_w)$. This finishes the proof. □

B. Proof of Proposition 1

This section follows the notations defined in Section V. To prove Proposition 1, we need to introduce two lemmas (proof will be given).

Lemma 3: $S_w^{*-1}S_b^*$ is symmetrical matrix and $S_w^{*-1}S_b^* = S_w^{*-1} - I_{l\times l}$, where $I_{l\times l}$ is an unit matrix with $\text{rank}(I_{l\times l}) = l$.

Proof: Since $S_t = S_w + S_b$, we have the following.

$\Lambda_{\text{pca}}^{-1/2}\Lambda_{\text{pca}}\Lambda_{\text{pca}}^{-1/2} = \Lambda_{\text{pca}}^{-1/2}W_{\text{pca}}^TS_tW_{\text{pca}}\Lambda_{\text{pca}}^{-1/2} = S_w^* + S_b^*$, i.e., $I_{l\times l} = S_w^* + S_b^*$.

Therefore, $S_w^{*-1}S_b^* = S_w^{*-1}(I_{l\times l} - S_w^*) = S_w^{*-1} - I_{l\times l}$.

Since S_w^* is a symmetrical and positive definite matrix, it yields $S_w^* = U_w^*\Lambda_w^*U_w^{*T}$, where U_w^* is an orthogonal matrix, and $\Lambda_w^* = \text{diag}(\lambda_{w_1}^*, \ldots, \lambda_{w_l}^*), 0 < \lambda_{w_1}^* \le \cdots \le \lambda_{w_l}^*$.

Therefore, $S_w^{*-1} = U_w^*\Lambda_w^{*-1}U_w^{*T}$, which indicates that S_w^{*-1} is symmetrical so that $S_w^{*-1}S_b^* = S_w^{*-1} - I_{l\times l}$ is a symmetrical matrix. This finishes the proof. □

Lemma 4: Suppose λ is any eigenvalue of $S_w^{*-1}S_b^*$. Then, $\lambda \ge 0$.

Proof: Note that from Lemma 3, we obtain $S_w^{*-1} = U_w^*\Lambda_w^{*-1}U_w^{*T}$. Suppose w is the eigenvector of $S_w^{*-1}S_b^*$ corresponding to the eigenvalue λ. Then, we have

$$S_w^{*-1}S_b^*w = U_w^*\Lambda_w^{*-1}U_w^{*T}S_b^*w = \lambda w.$$

Furthermore, we obtain

$$\left(\Lambda_w^{*-\frac{1}{2}}U_w^{*T}\right)S_b^*\left(\Lambda_w^{*-\frac{1}{2}}U_w^{*T}\right)^T\left(\Lambda_w^{*\frac{1}{2}}U_w^{*T}\right)w = \lambda\left(\Lambda_w^{*\frac{1}{2}}U_w^{*T}\right)w.$$

Let

$$G = \Lambda_w^{*-\frac{1}{2}}U_w^{*T}, \quad S_{\text{bg}} = GS_b^*G^T, \quad w_{\text{bg}} = \left(\Lambda_w^{*\frac{1}{2}}U_w^{*T}\right)w.$$

Therefore, $S_{\text{bg}}w_{\text{bg}} = \lambda w_{\text{bg}}$, and hence, λ is the eigenvalue of S_{bg}. Moreover, we obtain

$$S_{\text{bg}} = GS_b^*G^T = \left(G\Lambda_{\text{pca}}^{-\frac{1}{2}}W_{\text{pca}}^T\Phi_b\right)\left(G\Lambda_{\text{pca}}^{-\frac{1}{2}}W_{\text{pca}}^T\Phi_b\right)^T.$$

Therefore, S_{bg} is a semipositive definite matrix, implying that $\lambda \ge 0$. This finishes the proof. □

Now let us begin to prove Proposition 1.

Proof of Proposition 1: Let w be an eigenvector of $S_w^{*-1}S_b^*$ that corresponds to the eigenvalue λ, i.e., $S_w^{*-1}S_b^*w = \lambda w$. From Lemma 4, we know that $\lambda \ge 0$, and from Lemma 3, we get

$$\left(S_w^{*-1} - I_{l\times l}\right)w = S_w^{*-1}w - w = \lambda w.$$

Thus, $S_w^{*-1}w = \lambda w + w = (\lambda + 1)w$.

Therefore, any w that is the eigenvector of $S_w^{*-1}S_b^*$ corresponding to the eigenvalue λ is exactly the eigenvector of S_w^{*-1} corresponding to the eigenvalue $\lambda + 1$.

This finishes the proof of the first case. The proof of the second case is easy and is just the inverse of the proof of the first case. □

Then, the proof of the Proposition 1 is finished.

C. Example

We use the genetic algorithm to develop the example. The algorithm is ready made in this paper, except for a little modification. Use the model in Section IV and modify the fitness function as follows:

$$F(P) = \frac{1}{\text{rcond}(P^TS_wP)}$$

where $\text{rcond}(P^TS_wP)$ is the reciprocal condition number estimate of matrix P^TS_wP (just like the command "rcond" in Matlab). We see that P^TS_wP is just the within-class scatter matrix after dimensionality reduction. We knowthat if P^TS_wP is well conditioned, $\text{rcond}(P^TS_wP)$ is near 1.0. If P^TS_wP is badly conditioned, $\text{rcond}(P^TS_wP)$ is near 0.0. Therefore, if P^TS_wP becomes singular, then $\text{rcond}(P^TS_wP)$ will be very small, and $F(P)$ will be large.

This model works on a subset of FERET, which consists of 72 objects and three training images per person. The crossover rate and mutation rate are set to be 80% and 23%, respectively, at the beginning, then decrease gradually, and finally reach at 70% and 3%, respectively. The population and generation of GA are 100 and 200, respectively. The experiment has been run many times with different training lists, and in some cases, we have found some basis that would make P^TS_wP singular. An instance of the order of principal components selected in our experiment is

[1 3 4 5 7 9 11 13 14 15 16 17 18 19 21 23 24 27 32 33 34 37 38 40 42 48 49 50 51 52 53 55 57 61 62 63 64 65 66 67 68 70 72 73 75 76 77 79 80 81 82 83 84 86 88 89 90 91 93 94 96 97 98 99 100 101 105 106 107 108 109 110 111 113 114 115 116 117 119 120 122 123 124 125 127 128 129 130 131 132 138 139 140 142 145 146 147 148 149 151 152 155 157 159 160 161 163 164 165 167 168 169 170 174 176 177 179 182 183 185 186 187 188 189 190 191 192 193 194 195 196 197 199 200 201 202 203 206 207 209 210 211 213 214];

We see that there are 144 principal components selected, where $\text{rank}(S_w) = 144$. Using Matlab, we have calculated that

$$\text{size}(P^TS_wP) = (144\,144)$$
$$\text{rank}(P^TS_wP) = 143$$

where size and rank are commands in Matlab.

In addition, Matlab's warning message is that when calculating the inverse of P^TS_wP:

Warning: Matrix is close to singular or badly scaled.

Results may be inaccurate. rcond = 7.173 845e−017.

We see that P^TS_wP is really singular after dimensionality reduction, at least from the computation point of view.

In conclusion, this experiment also tells us why theorem PCA-DRT is needed.

D.

This example focuses on the mathematical operation.

Let

$$\Phi_t^T = \begin{pmatrix} 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}, \quad \Phi_w^T = \begin{pmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$\Phi_b^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

$$S_t = \Phi_t\Phi_t^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$S_w = \Phi_w \Phi_w^T = \begin{pmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}$$

$$S_b = \Phi_b \Phi_b^T = \begin{pmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \end{pmatrix}.$$

It is satisfied that $S_t = S_w + S_b, \text{rank}(S_w) = 2, \text{rank}(S_t) = 3, \text{rank}(S_b) = 1$ and is obvious that S_w is singular. We know that the eigenvalue of S_t is either 0 or 1 and that $w_1 = (0,0,1,0,0)^T, w_2 = (1,0,0,0,0)^T, w_3 = (0,1,0,0,0)^T$ are eigenvectors corresponding to eigenvalue 1, where they are linearly independent.

If $W_{\text{pca}} = (w_1, w_2)$, then $\widehat{S}_w = W_{\text{pca}}^T S_w W_{\text{pca}} = \begin{pmatrix} 1 & 0 \\ 0 & 0 \end{pmatrix}$, $\widehat{S}_w$ is still singular after dimensionality reduction. If $W_{\text{pca}} = (w_1, w_3)$, then $\widehat{S}_w = W_{\text{pca}}^T S_w W_{\text{pca}} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix}, \text{rank}(\widehat{S}_w) = \text{rank}(S_w) = 2$, and therefore, $\widehat{S}_w^{-1}$ exists.

ACKNOWLEDGMENT

The authors would like to thank the U.S. Army Research Laboratory and Carnegie Mellon University for the contribution of the FERET and CMU PIE databases, respectively. They would also like to thank X.-M. Du for the discussion about Appendix D. The authors would also like to thank the anonymous reviewers for their valuable comments and suggestions, which improved the paper.

REFERENCES

[1] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.

[2] H. Yu and J. Yang, "A direct LDA algorithm for high-dimensional data—With application to face recognition," *Pattern Recogn.*, vol. 34, pp. 2067–2070, 2001.

[3] L. Chen, H. Liao, M. Ko, J. Lin, and G. Yu, "A new LDA-based face recognition system, which can solve the small sample size problem," *Pattern Recogn.*, vol. 33, no. 10, pp. 1713–1726, 2000.

[4] Z. Jin, J. Y. Yang, Z. S. Hu, and Z. Lou, "Face recognition based on the uncorrelated discriminant transformation," *Pattern Recogn.*, vol. 34, pp. 1405–1416, 2001.

[5] J. Duchene and S. Leclercq, "An optimal transformation for discriminant and principal component analysis," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 10, no. 6, pp. 978–983, Jun. 1988.

[6] M. Kirby and L. Sirovich, "Application of the Karhunen-Loeve procedure for the characterization of human faces," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 12, no. 1, pp. 103–108, Jan. 1990.

[7] A. M. Martinez and A. C. Kak, "PCA versus LDA," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 23, no. 2, pp. 228–233, Feb. 2001.

[8] M. Srinivas and L. M. Patnaik, "Genetic algorithms: A survey," *IEEE Comput.*, vol. 27, no. 6, pp. 17–26, Jun. 1994.

[9] P. J. Phillips, H. Moon, S. A. Rizvi, and P. J. Rauss, "The feret evaluation methodology for face recognition algorithms," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[10] W. S. Yambor, B. A. Draper, and J. R. Beveridge, "Analyzing PCA-based face recognition algorithms: Eigenvector selection and distance measures," in *Proc. Second Workshop Empirical Evaluation Computer Vision*, 2000.

[11] Z. Sun, G. Bebis, X. Yuan, and S. J. Louis, "Genetic feature subset selection for gender classification: A comparison study," in *Proc. Sixth IEEE Workshop Applications Computer Vision*, 2002.

[12] A. Jain and D. Zongker, "Feature selection: Evaluation, application, and small sample performance," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 19, no. 2, pp. 153–158, Feb. 1997.

[13] R. Huang, Q. Liu, H. Lu, and S. Ma, "Solving the small sample size problem in LDA," in *Proc. Int. Conf. Pattern Recognition*, vol. 3, Aug. 2002.

[14] C. Liu and H. Wechsler, "Evolutionary pursuit and its application to face recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 22, no. 6, pp. 570–582, Jun. 2000.

[15] R. A. Fisher, "The use of multiple measures in taxonomic problems," *Ann. Eugenics*, vol. 7, pp. 179–188, 1936.

[16] W. Zhao, R. Chellappa, A. Rosenfeld, and P. J. Phillips, "Face recognition: A literature survey," *ACM Comput. Surveys*, pp. 399–458, 2003.

[17] H. Zhang and G. Sun, "Feature selection using tabu search method," *Pattern Recogn.*, vol. 35, pp. 701–711, 2002.

[18] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.

[19] P. J. Phillips, P. Grother, R. Micheals, D. M. Blackburn, E. Tabassi, and M. Bone. Face Recognition Vendor Test 2002: Evaluation Rep.. [Online]. Available: http://www.frvt.org/DLs/FRVT_2002_Evaluation_Report.pdf

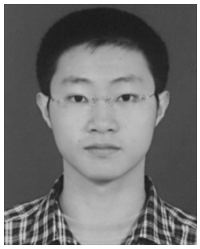
[20] C. Liu, "Gabor-based kernel PCA with fractional power polynomial models for face recognition," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. 26, no. 5, pp. 572–581, May 2004.

[21] A. Pentland, T. Starner, N. Etcoff, N. Masoiu, O. Oliyide, and M. Turk, "Experiments with eigenfaces," in *Proc. Looking at People Workshop Int. Joint Artificial Intelligence*, 1993.

[22] W. Zhao, R. Chellappa, and P. J. Phillips, "Subspace Linear Discriminant Analysis for Face Recognition," Univ. Maryland, College Park, MD, Tech. Rep. CAR-TR-914, CS-TR-4009.

[23] J. Ye and Q. Li, "LDA/QR: An efficient and effective dimension reduction algorithm and its theoretical foundation," *Pattern Recogn.*, vol. 37, pp. 851–854, 2004.

[24] T. Sim, S. Baker, and M. Bsat, "The CMU Pose, Illumination, and Expression (PIE) Database of Human Faces," The Robotics Inst., Carnegie Mellon Univ., Pittsburgh, PA, CMU-RI-TR-01-02, Jan. 2001.



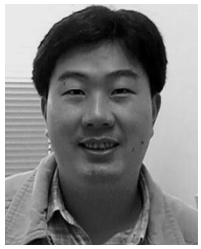
Wei-Shi Zheng was born in Guangzhou, China, in 1981. He received the B.S. degree in science with specialties in mathematics and computer science from Sun Yat-Sen University, Guangzhou, in 2003. He is now pursuing the Masters degree in applied mathematics with Sun Yat-Sen University and will start the pursuit of the Ph.D. degree in September 2005.

His current research interests include machine learning, pattern recognition, and computer vision.

Jian-Huang Lai was born in 1964. He received the M.Sc. degree in applied mathematics in 1989 and the Ph.D. degree in mathematics in 1999 from Sun Yat-Sen University, Guangzhou, China.

He has been teaching at Sun Yat-Sen University since 1989, where currently, he is a Professor with the Department of Mathematics and Computational Science. He had been a visiting scientist with the Harris Digital System Company, Center of Software of U.N. College, Macao, China, and Department of Computer Science, Hong Kong Baptist University. He has published over 40 scientific papers in the international journals, book chapters, and conferences. His current research interests are in the areas of digital image processing, pattern recognition, multimedia communication, wavelets, and their applications.

Dr. Lai had successfully organized the International Conference on Advances in Biometric Personal Authentication' 2004, which was also the Fifth Chinese Conference on Biometric Recognition (Sinobiometrics'04), Guangzhou, in December 2004. He has taken charge of more than four research projects, including NSFC (number 60144001, 60 373 082), the Key (Key grant) Project of Chinese Ministry of Education (number 105 134), and NSF of Guangdong, China (number 021 766). He serves as a board member of the Image and Graphics Association of China and also serves as a board member and secretary-general of the Image and Graphics Association of Guangdong.



Pong C. Yuen received the B.Sc. degree in electronic engineering with first-class honors in 1989 from City Polytechnic of Hong Kong and the Ph.D. degree in electrical and electronic engineering in 1993 from The University of Hong Kong.

Currently, he is an Associate Professor with the Department of Computer Science, Hong Kong Baptist University. His major research interests include human face recognition, signature recognition, and medical image processing. He has published more than 70 scientific articles in these areas. He was a recipient of the University Fellowship to visit The University of Sydney, Sydney, Australia, in 1996, where he was associated with the Laboratory of Imaging Science and Engineering, Department of Electrical Engineering. In 1998, he spent a six-month sabbatical leave with the Institute for Advanced Computer Studies (UMIACS), University of Maryland, College Park, where he was also associated with the Computer Vision Laboratory.

Dr. Yuen is the Director of the Croucher Advanced Study Institute on Biometric Authentication for 2004.