Whitened LDA for Face Recognition *

Vo Dinh Minh Nhat
Ubiquitous Computing Lab
Kyung Hee University
Suwon, Korea
vdmnhat@oslab.khu.ac.kr

SungYoung Lee
Ubiquitous Computing Lab
Kyung Hee University
Suwon, Korea
sylee@oslab.khu.ac.kr

Hee Yong Youn
Mobile Computing Lab
SungKyunKwan University
Suwon, Korea
youn@ece.skku.ac.kr

ABSTRACT

Over the years, many Linear Discriminant Analysis (LDA) algorithms have been proposed for the study of high dimensional data in a large variety of problems. An intrinsic limitation of classical LDA is the so-called "small sample size (3S) problem" that is, it fails when all scatter matrices are singular. Many LDA extensions were proposed in the past to overcome the 3S problems. However none of the previous methods could solve the 3S problem completely in the sense that it can keep all the discriminative features with a low computational cost. By applying LDA after whitening data, we proposed the Whitened LDA (WLDA) which can find the most discriminant features without facing the 3S problem. In WLDA, only eigenvalue problems instead of generalized eigenvalue problems are performed, leading to the low computation cost of WLDA. Experimental results are shown using two most popular Yale and ORL databases. Comparisons are given against Linear Discriminant Analysis (LDA), Direct LDA (DLDA), Null space LDA (NLDA) and several matrix-based subspace analysis approaches developed recently. We show that our method is always the best.

Categories and Subject Descriptors

I.5.4 [**PATTERN RECOGNITION**]: Applications

General Terms

Algorithms, Theory, Experimentation, Performance

Keywords

Face Recognition, Data Whitening, LDA

*This research was supported by the MIC(Ministry of Information and Communication), Korea, under the ITRC(Information Technology Research Center) support program supervised by the IITA (Institute of Information Technology Advancement) (IITA-2006-C1090-0602-0002)

CIVR'07, July 9–11, 2007, Amsterdam, The Netherlands.

1. INTRODUCTION

A facial recognition system is a computer-driven application for automatically identifying a person from a digital image. It does that by comparing selected facial features in the live image and a facial database. With the rapidly increasing demand on face recognition technology, it is not surprising to see an overwhelming amount of research publications on this topic in recent years. Like other classical statistical pattern recognition tasks, we usually represent data samples with n-dimensional vectors, i.e. data is vectorized to form data vectors before applying any technique. However in many real applications, the dimension of those 1D data vectors is very high, leading to the "curse of dimensionality". The curse of dimensionality is a significant obstacle in pattern recognition and machine learning problems that involve learning from few data samples in a high-dimensional feature space. In face recognition, Principal component analysis (PCA)[5] and Linear discriminant analysis (LDA)[1] are the most popular subspace analysis approaches to learn the low-dimensional structure of high dimensional data. PCA is a subspace projection technique widely used for face recognition. It finds a set of representative projection vectors such that the projected samples retain most information about original samples. The most representative vectors are the eigenvectors corresponding to the largest eigenvalues of the covariance matrix. Unlike PCA, LDA finds a set of vectors that maximizes Fisher Discriminant Criterion. It simultaneously maximizes the between-class scatter while minimizing the within-class scatter in the projective feature vector space. While PCA can be called unsupervised learning techniques, LDA is supervised learning technique because it needs class information for each image in the training process. This method overcomes the limitations of the Eigenface method by applying the Fisher's Linear Discriminant criterion. This criterion tries to maximize the ratio

$$\frac{w^T S_b w}{w^T S_w w} \tag{1}$$

where S_b is the between-class scatter matrix, and S_w is the within-class scatter matrix. Thus, by applying this method, we find the projection directions that on one hand maximize the Euclidean distance between the face images of different classes and on the other minimize the distance between the face images of the same class. This ratio is maximized when the column vectors of the projection matrix $W$are the eigenvectors of $S_w^{-1}S_b$. In face recognition tasks, this method cannot be applied directly since the dimension of the sample space is typically larger than the number of samples in

the training set. As a consequence, S_w is singular. This problem is known as the "small sample size problem"[3].
A lot of methods have been proposed to solve this problem, and reviews of those methods could be found in papers relevant to this problem. For the sake of completeness, we here summarize the most novel approaches to overcome the 3S problem. In [1], they proposed a two stage PCA+LDA method, also known as the Fisherface method, in which PCA is first used for dimension reduction so as to make S_w nonsingular before the application of LDA. However, in order to make S_w nonsingular, some directions corresponding to the small eigenvalues of S_w are thrown away in the PCA step. Thus, applying PCA for dimensionality reduction has the potential to remove dimensions that contain discriminative information. In [12], they try to make make S_w become nonsingular by adding a small perturbation matrix Δ to S_w. However, this method is very computationally expensive and will not be considered in the experiment part of this paper. The Direct-LDA method is proposed in [11]. First, the null space of S_b is removed and, then, the projection vectors that minimize the within-class scatter in the transformed space are selected from the range space of S_b. However, removing the null space of S_b by dimensionality reduction will also remove part of the null space of S_w and may result in the loss of important discriminative information. In [2], Chen et al. proposed the null space based LDA (NLDA), where the between-class scatter is maximized in the null space of the within-class scatter matrix. The singularity problem is thus implicitly avoided. Huang et al. in [4] improved the efficiency of the algorithm by first removing the null space of the total scatter matrix. In orthogonal LDA (OLDA) [8], a set of orthogonal discriminant vectors is computed, based on a new optimization criterion. The optimal transformation is computed through the simultaneous diagonalization of scatter matrices, while the singularity problem is overcome implicitly. In [10], they showed that NLDA is equivalent to OLDA, under a mild condition that the rank of the total scatter matrix equals to the sum of the rank of the between-class scatter matrix and the rank of the within-class scatter matrix. So we will choose only NLDA for experiment comparison intead of both NLDA and OLDA. In general, due to the nonsingular of the within-class scatter matrix, none of the above methods could solve the 3S problem completely in the sense that it can keep all the discriminative features with a low computational cost. In our paper, we propose a new LDA algorithm that can overcome those above disadvantages of previous work. For the fairness in algorithm evaluation, we also consider some matrix-based approaches developed recently. While all above approaches are based on 1D vector data, recently, two-dimensional PCA (2DPCA) [6] has been proposed in which image covariance matrices can be constructed directly using original image matrices. A brief of history of matrix-based subspace analysis can be summarized as follow. Based on PCA, some image-based subspace analysis approaches have been developed such as 2DPCA [6], Generalized low rank approximations of matrices (GLRAM) [7]. While 2DPCA is one-side low-rank approximation algorithm, GLRAM is two-side low-rank approximation one. Based on LDA, 2DLDA [9] has been developed. 2DLDA aims to find the two-sided optimal transformations such that the class structure of the original high-dimensional space is preserved in the low-dimensional space.

The key idea of WLDA is that we apply data whitening before perform LDA. With some nice properties of whitened data, we show how to turn the generalized eigenvalue problem of LDA into simple eigenvalue problem, leading to low computation cost of algorithm. While in previous works, some discriminant information lost during performing algorithms, WLDA has the ability of keeping all discriminant information. Some main contributions of this paper can be described as: giving the solution to LDA without facing the 3S problem, keeping all the discriminative features, solving the problem with low computational cost. The outline of this paper is as follows. In Section 2, all important previous and related works are described such as : PCA, LDA, 2DPCA, GLRAM and 2DLDA. The proposed method is described in Section 3. In Section 4, experimental results are presented for the ORL and Yale face image databases to demonstrate the effectiveness of our method. Finally, conclusions are presented in Section 5.

2. SUBSPACE ANALYSIS

One approach to cope with the problem of excessive dimensionality of the image space is to reduce the dimensionality by combining features. Linear combinations are particularly attractive because they are simple to compute and analytically tractable. In effect, linear methods project the high-dimensional data onto a lower dimensional subspace. Basic notations are described in Table 1 for reference. Suppose that we have N sample images $\{x_1, x_2, ..., x_N\}$ taking values in an n-dimensional image space. Let us also consider a linear transformation mapping the original n-dimensional image space into an m-dimensional feature space, where $m < n$. The new feature vectors $y_k \in \Re^m$ are defined by the following linear transformation:

$$y_k = W^T(x_k - \mu) \tag{2}$$

where $k = 1, 2, ..., N$, $\mu \in \mathbb{R}^n$ is the mean of all samples, and $W \in \Re^{n \times m}$ is a matrix with orthonormal columns. Atfer the linear transformation, each data point x_k can be represented by a feature vector $y_k \in \Re^m$ which is used for classification.

2.1 Principal Component Analysis - PCA

Different objective functions will yield different algorithms with different properties. PCA aims to extract a subspace in which the variance is maximized. Its objective function is $w_{opt} = \arg\max_w (w^T S_t w)$, with the total scatter matrix is defined as

$$S_t = \frac{1}{N} \sum_{k=1}^{N} (x_k - \mu)(x_k - \mu)^T \tag{3}$$

where $\mu = \frac{1}{N} \sum_{i=1}^{N} x_i$ is the mean of all samples. The optimal projection $W = [w_1 w_2 ... w_m]$ is the set of n-dimensional eigenvectors of S_t corresponding to the m largest eigenvalues.

2.2 Linear Discriminant Analysis - LDA

While PCA seeks directions that are efficient for representation, LDA seeks directions that are efficient for discrimination. Assume that each image belongs to one of C classes $\{\Pi_1, \Pi_2, ..., \Pi_C\}$. Let N_i be the number of the samples in class $\Pi_i (i = 1, 2, ..., C)$, $\mu_i = \frac{1}{N_i} \sum_{x \in \Pi_i} x$ be the mean of the

Table 1: Basic Notations

Notations	Descriptions
$x_i \in \Re^n$	the i^{th}image point in vector form
$X_i \in \Re^{r \times c}$	the i^{th}image point in matrix form
Π_i	the i^{th}class of data points (both in vector and matrix form)
n	dimension of x_i
m	dimension of reduced feature vector y_i
r	number of rows in X_i
c	number of columns in X_i
N	number of data samples
C	number of classes
N_i	number of data samples in class Π_i
L	transformation on the left side
R	transformation on the right side
l_1	number of rows in Y_i
l_2	number of columns in Y_i

samples in class Π_i. Then the between-class scatter matrix S_b and the within-class scatter matrix S_w are defined

$$S_b = \frac{1}{N}\sum_{i=1}^{C} N_i(\mu_i - \mu)(\mu_i - \mu)^T \tag{4}$$

$$S_w = \frac{1}{N}\sum_{i=1}^{C}\sum_{x_k \in \Pi_i} (x_k - \mu_i)(x_k - \mu_i)^T \tag{5}$$

We should note some properties of these scatter matrices S_b, S_w, S_t as follow

$$\begin{array}{l} rank(S_t) = N - 1 \\ rank(S_b) = C - 1 \\ rank(S_w) = N - C \\ S_t = S_b + S_w \end{array} \tag{6}$$

In LDA, the projection W_{opt} is chosen to maximize the ratio of the determinant of the between-class scatter matrix of the projected samples to the determinant of the within-class scatter matrix of the projected samples, i.e.,

$$w_{opt} = \arg\max_w \frac{w^T S_b w}{w^T S_w w} \tag{7}$$

The optimal projection for LDA is $W = [w_1 w_2 ... w_m]$, where $\{w_i \,|i = 1, 2, ..., m\}$ is the set of generalized eigenvectors of S_b and S_w corresponding to the m largest generalized eigenvalues $\{\lambda_i \,|i = 1, 2, ..., m\}$, i.e.,

$$\begin{array}{l} S_b w_i = \lambda_i S_w w_i \\ \Leftrightarrow S_b S_w^{-1} w_i = \lambda_i w_i \end{array} \quad i = 1, 2, ..., m \tag{8}$$

To overcome the singularity of S_w while solving (7), we consider 3 main algorithms to solve the problem in this paper. First one is Fisherface method [1], in which PCA is first used for dimension reduction so as to make S_w nonsingular before the application of LDA. Second approach is the null space based LDA (NLDA) [2], where the between-class scatter is maximized in the null space of the within-class scatter matrix. The singularity problem is thus implicitly avoided. Final one is Direct-LDA method [11]. First, the null space of S_b is removed and, then, the projection vectors that minimize the within-class scatter in the transformed space are selected from the range space of S_b. Due to the limited length of paper, details of those algorithms can be found in respective references.

2.3 Two-dimensional PCA - 2DPCA

In 2D approach, the image matrix does not need to be previously transformed into a vector, so a set of N sample images is represented as $\{X_1, X_2, ..., X_N\}$ with $X_i \in \Re^{r \times c}$, which is a matrix space of size $r \times c$. The total scatter matrix is defined as

$$T_t = \frac{1}{N}\sum_{i=1}^{N} (X_i - M)^T (X_i - M) \tag{9}$$

with $M = \frac{1}{N}\sum_{i=1}^{N} X_i \in \Re^{r \times c}$ is the mean image of all samples. $T_t \in \Re^{r \times r}$ is also called image covariance (scatter) matrix. A linear transformation mapping the original $r \times c$ image space into an $r \times m$ feature space, where $m < c$. The new feature matrices $Y_i \in \Re^{r \times m}$ are defined by the following linear transformation:

$$Y_i = (X_i - M)W \in \Re^{r \times m} \tag{10}$$

where $i = 1, 2, ..., N$ and $W \in \Re^{r \times m}$ is a matrix with orthogonal columns. In 2DPCA, the projection W_{opt} is chosen to maximize $tr(W^T T_t W)$. The optimal projection $W_{opt} = [w_1 w_2 ... w_m]$ with $\{w_i \,|i = 1, 2, ..., m\}$ is the set of c-dimensional eigenvectors of T_t corresponding to the m largest eigenvalues. After a transformation by 2DPCA, a feature matrix is obtained for each image. Then, a nearest neighbor classifier is used for classification. Here, the distance between two arbitrary feature matrices$Y_i$and$Y_j$is defined by using Euclidean distance as follows:

$$d(Y_i, Y_j) = \sqrt{\sum_{u=1}^{k}\sum_{v=1}^{s} (Y_i(u, v) - Y_j(u, v))^2} \tag{11}$$

Given a test sampleY_t, if $d(Y_t, Y_c) = \min_j d(Y_t, Y_j)$, then the resulting decision is Y_t belongs to the same class as Y_c.

2.4 Generalized Low Rank Approximations of Matrices - GLRAM

In [7], they considered the problem of computing low rank approximations of matrices which are based on a collection

Table 2: Algorithm – GLRAM

Algorithm – GLRAM
Step 0
Initialize $L = L^{(0)} = [I_1, 0]^T$, and set $k = 0$.
Step 1
Compute $l_2$eigenvectors $\{\Phi_i^{R(k+1)}\}_{i=1}^{l_2}$of the matrix $S_R = \sum_{i=1}^{N} X_i^T L^{(k)} L^{(k)T} X_i$ corresponding to the largest l_2 eigenvalues and form$R^{(k+1)} = [\Phi_1^{R(k+1)}..\Phi_{l_2}^{R(k+1)}]$.
Step 2
Compute $l_1$eigenvectors $\{\Phi_i^{L(k+1)}\}_{i=1}^{l_1}$of the matrix $S_L = \sum_{i=1}^{N} X_i R^{(k+1)} R^{(k+1)T} X_i^T$ corresponding to the largest l_1 eigenvalues and form$L^{(k+1)} = [\Phi_1^{L(k+1)}..\Phi_{l_1}^{L(k+1)}]$.
Step 3
If$L^{(k+1)}$,$R^{(k+1)}$are not convergent then set increase$k$by 1 and go to Step 1, othervise proceed to Step 4.
Step 4
Let$L^* = L^{(k+1)}$,$R^* = R^{(k+1)}$and compute$Y_i^* = L^{*T} X_i R^*$for$i = 1..N$.

of matrices. By solving an optimization problem, which aims to minimize the reconstruction (approximation) error, they derive an iterative algorithm, namely GLRAM, which stands for the Generalized Low Rank Approximations of Matrices. GLRAM reduces the reconstruction error sequentially, and the resulting approximation is thus improved during successive iterations. Formally, they consider the following optimization problem

$$\begin{aligned} &\min_{L,R,Y_i} \sum_{i=1}^{N} \left\| X_i - L Y_i R^T \right\|_F^2 \\ &s.t. \quad L^T L = I_1, R^T R = I_2 \end{aligned} \tag{12}$$

where $L \in \Re^{r \times l_1}$, $R \in \Re^{c \times l_2}$, $Y_i \in \Re^{l_1 \times l_2}$ for $i = 1..N$, $I_1 \in \Re^{l_1 \times l_1}$ and $I_2 \in \Re^{l_2 \times l_2}$ are identity matrices, where $l_1 \leqslant r$ and $l_2 \leqslant c$. An iterative procedure for computing L and R is presented in Table 2.

2.5 2DLDA

In [9], they proposed a novel LDA algorithm, namely 2DLDA, which stands for 2-Dimensional Linear Discriminant Analysis. 2DLDA aims to find the two-sided optimal transformations (projections L and R) such that the class structure of the original high-dimensional space is preserved in the low-dimensional space. A natural similarity metric between matrices is the Frobenius norm. Under this metric, the (squared) within-class and between-class distances D_w and D_b can be computed as follows:

$$\begin{aligned} D_w &= \sum_{j=1}^{C} \sum_{X_i \in \Pi_j} \|X_i - M_j\|_F^2 \\ &= tr\left(\sum_{j=1}^{C} \sum_{X_i \in \Pi_j} (X_i - M_j)(X_i - M_j)^T \right) \end{aligned} \tag{13}$$

$$\begin{aligned} D_b &= \sum_{j=1}^{C} N_j \|M_j - M\|_F^2 \\ &= tr\left(\sum_{j=1}^{C} N_j (M_j - M)(M_j - M)^T \right) \end{aligned} \tag{14}$$

In the low-dimensional space resulting from the linear transformations L and R, the with-in and between-class distances $\tilde{D}_w$ and $\tilde{D}_b$ can be computed as follows:

$$\tilde{D}_w = tr\left(\sum_{j=1}^{C} \sum_{X_i \in \Pi_j} L^T (X_i - M_j) R R^T (X_i - M_j)^T L \right) \tag{15}$$

$$\tilde{D}_b = tr\left(\sum_{j=1}^{C} N_j L^T (M_j - M) R R^T (M_j - M)^T L \right) \tag{16}$$

The optimal transformations L and R would maximize $F(L, R) = \tilde{D}_b / \tilde{D}_w$. Let us define

$$S_w^R = \sum_{X_i \in \Pi_j} (X_i - M_j) R R^T (X_i - M_j)^T \tag{17}$$

$$S_b^R = \sum_{j=1}^{C} N_j (M_j - M) R R^T (M_j - M)^T \tag{18}$$

$$S_w^L = \sum_{X_i \in \Pi_j} (X_i - M_j)^T L L^T (X_i - M_j) \tag{19}$$

$$S_b^L = \sum_{j=1}^{C} N_j (M_j - M)^T L L^T (M_j - M) \tag{20}$$

After defining those matrices we can derive the 2DLDA algorithm as in Table 3.

3. WHITENED LDA

In this part, we first review data whitening and then derive the WLDA in details.

3.1 Data Whitening

Let $x \in \Re^n$ denote a random vector with mean μ and positive semi-definite covariance matrix C_x. We wish to whiten the vector $x \in \Re^n$ using a whitening transformation P to obtain the random vector $y = P(x - \mu)$, which is a zero-mean random vector and the covariance matrix C_y of y is equal to identity matrix I. Thus we seek a transformation P such that

$$C_y = P C_x P^T = I \tag{21}$$

We refer any matrix P satisfying (21) as a whitening transformation matrix. Given a covariance matrix C_x, there are many ways to choose a whitening transformation P satisfying (21). One popular method for whitening is to use the eigen-value decomposition (EVD) of the covariance matrix. By using EVD, we can diagonalize or factor the covariance matrix C_x with $r = rank(C_x)$ in the following way

$$C_x = U \Lambda U^T \tag{22}$$

where $U \in \Re^{n \times r}$ is the orthogonal matrix of eigenvectors corresponding to non-zero eigenvalues of covariance matrix C_x and $\Lambda = diag(\sigma_1, \sigma_2, ..., \sigma_r) \in \Re^{r \times r}, \sigma_1 \geq \sigma_2 \geq ... \geq$

Table 3: Algorithm – 2DLDA

Algorithm – 2DLDA
Step 0
Initialize $R = R^{(0)} = [I_2, 0]^T$, and set $k = 0$.
Step 1
Compute
$S_w^{R(k)} = \sum_{X_i \in \Pi_j} (X_i - M_j) R^{(k)} R^{(k)T} (X_i - M_j)^T$
$S_b^{R(k)} = \sum_{j=1}^{C} N_j (M_j - M) R^{(k)} R^{(k)T} (M_j - M)^T$
Step 2
Compute l_1 eigenvectors $\{\Phi_i^{L(k)}\}_{i=1}^{l_1}$ of the matrix $(S_w^{R(k)})^{-1} S_w^{R(k)}$ and form $L^{(k)} = [\Phi_1^{L(k)} .. \Phi_{l_1}^{L(k)}]$.
Step 3
Compute
$S_w^{L(k)} = \sum_{X_i \in \Pi_j} (X_i - M_j)^T L^{(k)} L^{(k)T} (X_i - M_j)$
$S_b^{L(k)} = \sum_{j=1}^{C} N_j (M_j - M)^T L^{(k)} L^{(k)T} (M_j - M)$
Step 4
Compute l_2 eigenvectors $\{\Phi_i^{R(k)}\}_{i=1}^{l_2}$ of the matrix $(S_w^{L(k)})^{-1} S_w^{L(k)}$ and form $R^{(k+1)} = [\Phi_1^{R(k)} .. \Phi_{l_1}^{R(k)}]$.
Step 5
If $L^{(k)}$,$R^{(k+1)}$ are not convergent then set increase k by 1 and go to Step 1, othervise proceed to Step 6.
Step 6
Let $L^* = L^{(k)}$,$R^* = R^{(k+1)}$ and compute $Y_i^* = L^{*T} X_i R^*$ for $i = 1..N$.

$\sigma_r > 0$ is the diagonal matrix of non-zero eigenvalues. Then whitening a random vector x with mean μ and covariance matrix C_x can be obtained by performing the following calculation with whitening transformation matrix $P = \Lambda^{-1/2} U^T$

$$y = P(x - \mu) = \Lambda^{-1/2} U^T (x - \mu) \tag{23}$$

Thus, the output of this transformation has expectation

$$E\{y\} = \Lambda^{-1/2} U^T (E\{x\} - \mu) = \Lambda^{-1/2} U^T (\mu - \mu) = 0 \tag{24}$$

Due to $E\{y\} = 0$ and covariance matrix C_y can be written as

$$\begin{aligned} C_y &= E\{yy^T\} = E\left\{\Lambda^{-1/2} U^T (x-\mu)(x-\mu)^T U \Lambda^{-1/2}\right\} \\ &= \Lambda^{-1/2} U^T C_x U \Lambda^{-1/2} = I \end{aligned} \tag{25}$$

Thus, with the above transformation, we can whiten the random vector to have zero mean and the identity covariance matrix.

3.2 Whitened LDA

The key idea of WLDA is that we apply data whitening before perform LDA. So we first perform data whitening by diagonalizing total covariance matrix S_t with $r = rank(S_t) = N - 1$ as

$$S_t = U \Lambda U^T \tag{26}$$

where $U \in \Re^{n \times r}$ is the orthogonal matrix of eigenvectors corresponding to non-zero eigenvalues of covariance matrix S_t and $\Lambda = diag(\sigma_1, \sigma_2, ..., \sigma_r) \in \Re^{r \times r}, \sigma_1 \geq \sigma_2 \geq ... \geq \sigma_r > 0$ is the diagonal matrix of non-zero eigenvalues. Then we can form the whitening transformation matrix $P = \Lambda^{-1/2} U^T$ and obtain the whitened data as

$$y_i = P(x_i - \mu) = \Lambda^{-1/2} U^T (x_i - \mu) \in \Re^r \tag{27}$$

where $i = 1..N$. Then the between-class scatter matrix G_b and the within-class scatter matrix G_w after data whitening are re-defined as

$$G_b = \frac{1}{N} \sum_{i=1}^{C} N_i \eta_i \eta_i^T \tag{28}$$

$$G_w = \frac{1}{N} \sum_{i=1}^{C} \sum_{y_k \in \Pi_i} (y_k - \eta_i)(y_k - \eta_i)^T \tag{29}$$

where η_i is the mean of whitened data in class i^{th}, $i = 1..C$ and defined as

$$\begin{aligned} \eta_i &= \tfrac{1}{N_i} \sum_{k=1}^{N_i} y_k = \tfrac{1}{N_i} \sum_{k=1}^{N_i} P(x_k - \mu) \\ &= P\left(\tfrac{1}{N_i} \sum_{k=1}^{N_i} x_k - \mu\right) = P(\mu_i - \mu) \end{aligned} \tag{30}$$

THEOREM 1. *Given the whitened data, the between-class scatter matrix G_b and the within-class scatter matrix G_w defined as in (28) and (29), then we have*

$$G_b + G_w = I_r$$

where $I_r \in \Re^{r \times r}$ is an identity matrix.

PROOF. We can re-write the between-class scatter matrix G_b

$$\begin{aligned} G_b &= \tfrac{1}{N} \sum_{i=1}^{C} N_i \eta_i \eta_i^T \\ &= \tfrac{1}{N} \sum_{i=1}^{C} N_i P(\mu_i - \mu)(\mu_i - \mu)^T P^T = P S_b P^T \end{aligned} \tag{31}$$

and the within-class scatter matrix G_w as

$$\begin{aligned} &G_w = \tfrac{1}{N} \sum_{i=1}^{C} \sum_{y_k \in \Pi_i} (P(x_k - \mu) - P(\mu_i - \mu)) \\ &(P(x_k - \mu) - P(\mu_i - \mu))^T = \\ &\tfrac{1}{N} \sum_{i=1}^{C} \sum_{y_k \in \Pi_i} P(x_k - \mu_i)(x_k - \mu_i)^T P^T = P S_w P^T \end{aligned} \tag{32}$$

From (31) and (31), we have

$$\begin{aligned} G_b + G_w &= P S_b P^T + P S_w P^T \\ &= P(S_b + S_w) P^T = P S_t P^T = I_r \end{aligned} \tag{33}$$

Proof is done. □

Now we can formulate the Whitened LDA problem as following optimization problem

$$J_{WLDA}(v) = \max_v \frac{v^T G_b v}{v^T G_w v} \tag{34}$$

where $v \in \Re^r$. The solution of this optimization problem can be obtained based on Theorem 2.

THEOREM 2. *The optimization problem in (34) is equal to the following optimization problem*

$$\max_{v^T v = 1} v^T G_b v \tag{35}$$

PROOF. Applying the fact that $G_b + G_w = I_r$ from Theorem 1, we can re-write the optimization problem in (34) as

$$\max_v \frac{v^T G_b v}{v^T G_w v} = \max_v \frac{v^T G_b v}{v^T (I_r - G_b) v} = \max_v \frac{v^T G_b v}{v^T v - v^T G_b v} \quad (36)$$

We can see that if $v = v_0$ is an optimal vector of (36) then cv is also an optimal solution of (36), where c scalar value. So we constrain v to unit length, i.e. $v^T v = 1$. Now, the problem in (36) can be written with constraints as follow,

$$\max_{v^T v=1} \frac{v^T G_b v}{v^T v - v^T G_b v} \quad (37)$$

It's easy to see from (37) that the solution for optimization problem in (34) can be obtained by solving the following simple eigenvalue problem

$$\max_{v^T v=1} v^T G_b v \quad (38)$$

Proof is done. □

It's easy to see that the optimal projection matrix $V = [v_1 v_2 ... v_m]$ for (38) is the set of r-dimensional eigenvectors of G_b corresponding to the m largest eigenvalues. Finally the optimal projection matrix for WLDA can be calculated as $W_{WLDA} = P^T V$. One should note that in WLDA we find the optimal projection vector w_{opt} among all vector w in the form of

$$w = P^T v \quad , \forall v \in \Re^r \quad (39)$$

Re-writing (39), we have

$$\begin{array}{l} w = P^T v = U \Lambda^{-1/2} v \quad , \forall v \in \Re^r \\ \Leftrightarrow w \in span\{u_1, u_2, .., u_r\} = range(S_t) \end{array} \quad (40)$$

Now we can see that the WLDA search optimal projection vectors in the range space of S_t which is actually the subspace that contain all the discriminant projection vectors. This fact is based on Theorem 3.

THEOREM 3. *All discriminant projection vectors can be found in the range space of S_t*

PROOF. Let V be the range space S_t, $V^\perp$ be the null space of S_t. Equivalently,

$$V = span\{\alpha_k | S_t \alpha_k \neq 0, k = 1, .., r\} \quad (41)$$

and

$$V^\perp = span\{\alpha_k | S_t \alpha_k = 0, k = r+1, .., n\} \quad (42)$$

where r is the rank of S_t, $\{\alpha_1, .., \alpha_d\}$ is an orthonormal set and $\{\alpha_1, .., \alpha_r\}$ is the set of orthonormal eigenvectors corresponding to the non-zero eigenvalues of S_t. Since $\Re^n = V \oplus V^\perp$, every vector $a \in \Re^n$ has a unique decomposition of the form $a = b + c$, where $b \in V$ and $c \in V^\perp$. And the projected value corresponding to x_k on the projection vector a can be formed as

$$\begin{array}{l} y_k^{(a)} = a^T (x_k - \mu) = b^T (x_k - \mu) + c^T (x_k - \mu) \\ = b^T (x_k - \mu) \end{array} \quad (43)$$

From (43) we see that we can find the all discriminant projection vectors from range space of S_t. Proof is done. □

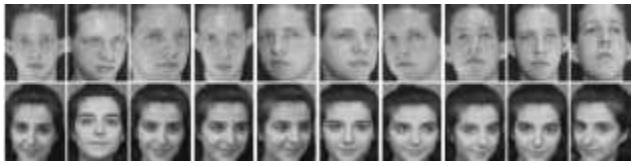

Figure 1: Twenty sample images from ORL face database

Figure 2: Ten sample images from Yale face database

4. EXPERIMENTAL RESULTS

This section evaluates the performance of PCA [5], Fisherface [1], Direct-LDA [11], NLDA [4], 2DPCA [6], GLRAM [7], 2DLDA [9] and our new approach WLDA based on using Yale face database and ORL face database. In this paper, we apply the nearest-neighbor classifier for its simplicity. The Euclidean metric is used as our distance measure. In short, the recognition process has three steps. First, we calculate the face subspace from the training set of face images; then the new face image to be identified is projected into low-dimensional subspace. Finally, the new face image is identified by a nearest neighbor classifier.Some sample images from ORL and Yale databases are shown in Fig. 1. and Fig. 2.

4.1 Yale Face Database

The Yale face Database contains 165 grayscale images in GIF format of 15 individuals. There are 11 images per subject, one per different facial expression or configuration: center-light, w/glasses, happy, left-light, w/no glasses, normal, right-light, sad, sleepy, surprised, and wink. A random subset with $k(k = 2, 3, 4, 5)$ images per individual was taken with labels to form the training set. The rest of the database was considered to be the testing set. 10 times

Table 4: Comparison of the top recognition accuracy (%) on Yale database.

k	2	3	4	5
PCA	75.556	83.333	84.762	87.778
Fisherfaces	84.444	86.667	93.333	93.333
DLDA	80	83.333	91.429	91.111
NLDA	85.185	87.5	94.286	93.333
2DPCA	76.296	83.333	88.571	88.889
GLRAM	74.2	84.35	84.762	88.1
2DLDA	82.1	84.3	89.2	90
WLDA	**86.481**	**89.167**	**94.524**	**96.111**

Table 5: Comparison of the top recognition accuracy (%) on ORL database.

k	2	3	4	5
PCA	81.875	87.5	90	90.5
Fisherfaces	81.563	85.714	86.667	82.5
DLDA	84.375	87.857	90.833	92.5
NLDA	84.063	87.857	91.25	91.5
2DPCA	84.063	86.071	89.167	91.5
GLRAM	81.875	86.75	89.45	90
2DLDA	82.175	84.234	90	90.75
WLDA	**88.875**	**89.857**	**92.417**	**94**

of random selection for training examples were performed and the average recognition result was recorded. The training samples were used to learn the subspace. The testing samples were then projected into the low-dimensional representation subspace. Recognition was performed using a nearest-neighbor classifier. We tested the recognition rates with different number of training samples. We show the best results obtained by PCA [5], Fisherface [1], Direct-LDA [11], NLDA [4], 2DPCA [6], GLRAM [7], 2DLDA [9] and our new approach WLDA in Table 4.

4.2 ORL Face Database

In the ORL database, there are ten different images of each of 40 distinct subjects. For some subjects, the images were taken at different times, varying the lighting, facial expressions (open / closed eyes, smiling / not smiling) and facial details (glasses / no glasses). All the images were taken against a dark homogeneous background with the subjects in an upright, frontal position (with tolerance for some side movement). A random subset with $k(k = 2, 3, 4, 5)$ images per individual was taken with labels to form the training set. The rest of the database was considered to be the testing set. 10 times of random selection for training example were performed and the average recognition result was recorded. The experimental protocol is the same as before. The best recognition result of each method are shown in Table 5.

Next, in order to test the performance of these LDA-based algorithms versus the reduced dimensions, we vary the reduced dimension m from 1 to 39 and perform these approaches based on image ORL database. Fig. 3,4,5,6. show the recognition accuracy versus reduced dimension of LDA-based algorithms and our algorithm, corresponding to the number of training sample $k = 2, 3, 4, 5$.

5. CONCLUSIONS

In this paper, we propose a new LDA algorithm that can overcome some disadvantages of previous work. The key idea of WLDA is that we apply data whitening before perform LDA. With some nice properties of whitened data, we show how to turn the generalized eigenvalue problem of LDA into simple eigenvalue problem, leading to low computation cost of algorithm. While in previous works, some discriminant information lost during performing algorithms, WLDA has the ability of keeping all discriminant information. Experiment results show us the effectiveness of our algorithm.

6. REFERENCES

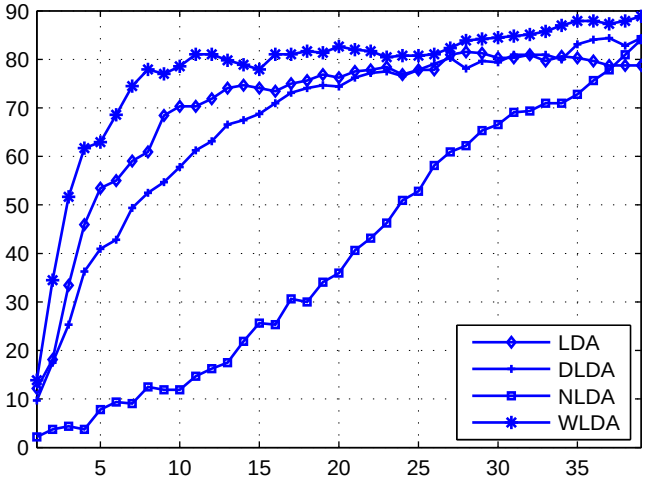

Figure 3: Comparison of the recognition accuracy (%) versus reduced dimension m (1 to 39) on ORL database with training sample $k = 2$

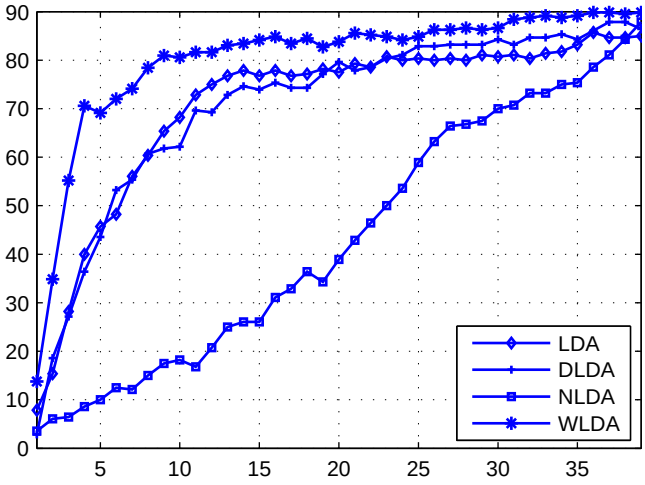

Figure 4: Comparison of the recognition accuracy (%) versus reduced dimension m (1 to 39) on ORL database with training sample $k = 3$

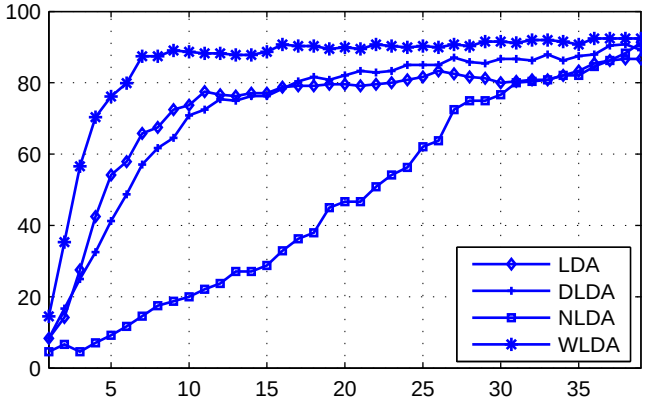

Figure 5: Comparison of the recognition accuracy (%) versus reduced dimension m (1 to 39) on ORL database with training sample $k = 4$

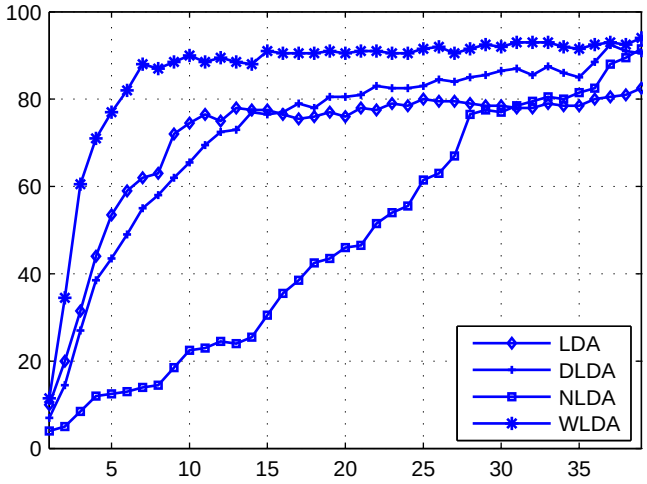

Figure 6: Comparison of the recognition accuracy (%) versus reduced dimension m (1 to 39) on ORL database with training sample $k = 5$

[1] P. Belhumeur, J. Hespanha, and D. Kriegman. Eigenfaces vs. fisherfaces: recognition using class specific linearprojection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 19(7):711–720, Jul 1997.

[2] L. Chen, H. Liao, M. Ko, J. Liin, and G. Yu. A new lda-based face recognition system which can solve the samll sample size problem. *Pattern Recognition*, 33(10):1713–1726, Oct. 2000.

[3] K. Fukunaga. *Introduction to statistical pattern recognition*. Academic Press Professional, Inc., San Diego, CA, USA, 1990.

[4] R. Huang, Q. Liu, H. Lu, and S. Ma. Solving the small sample size problem of lda. *Pattern Recognition, 2002. Proceedings. 16th International Conference on*, 3:29–32, 2002.

[5] M. Turk and A. Pentland. Eigenfaces for recognition. *Journal of Cognitive Neuroscience*, 3(1):71–86, 1991.

[6] J. Yang, D. Zhang, A. F. Frangi, and J. yu Yang. Two-dimensional pca: A new approach to appearance-based face representation and recognition. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, page 112, 2004.

[7] J. Ye. Generalized low rank approximations of matrices. *Proceedings of the twenty-first international conference on Machine learning*, 26(1):131– 137, Jan 2004.

[8] J. Ye. Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems. *Journal of Machine Learning Research*, 6:483–502, 2005.

[9] J. Ye, R. Janardan, and Q. Li. Two-dimensional linear discriminant analysis. *Neural Information Processing Systems*, 2004.

[10] J. Ye and T. Xiong. Null space versus orthogonal linear discriminant analysis. *Proceedings of the 23rd international conference on Machine learning*, pages 1073 – 1080, 2006.

[11] H. Yu and J. Yang. A direct lda algorithm for high-dimensional data - with application to face recognition. *Pattern Recognition*, (34):2067–2070, 2001.

[12] W. Zhao, R. Chellappa, and A. Krishnaswamy. Discriminant analysis of principal components for face recognition. *Proceedings of the 3rd. International Conference on Face and Gesture Recognition*, page 336, 1998.