DLDA and LDA/QR Equivalence Framework for Human Face Recognition

Rong-Hua Li, Eddie, C.L. Chan, George Baciu, *Member, IEEE*
$\{$*csrhli, csclchan, csgeorge*$\}$*@comp.polyu.edu.hk*

***Abstract*—Singularity problem in human face feature extraction is very challenging that has gained a lot of attentions during the last decade. A pseudo-inverse linear discriminant analysis (LDA) plays a important role to solve the singularity problem of the scatter matrices. In this paper, we make use of Linear Discriminant Analysis via QR decomposition (LDA/QR) and Direct Linear Discriminant Analysis (DLDA) to solve the singularity problem in face feature recognition. We also show that an equivalent relationship between DLDA and LDA/QR. They can be regarded as a special case of pseudo-inverse LDA. Similar to LDA/QR algorithm, DLDA could be a two-stage LDA method. Interestingly, we find that the first stage of DLDA can act as a dimensionality reduction algorithm. In our experiment, we compare DLDA and LDA/QR algorithms in terms of classification accuracy, computational complexity in ORL and Yale face datasets. We have also conducted experiments to compare their first stages on these datasets. Our results indicate that the empirical and theoretic proofs of equivalence between DLDA and LDA/QR algorithms coincidentally converge and verify their same capabilities in the dimension reduction.**

***Index Terms*—Face Recognition, Direct LDA, LDA/QR, Pseudo-inverse LDA**

I. Introduction

LINEAR Discriminant Analysis (LDA) is a well-known feature extraction algorithm for human face images which deals with the classical singularity problem [6] [2] of scatter matrices. Linear Discriminant Analysis via QR decomposition(LDA/QR) [9] and Direct Linear Discriminant Analysis (DLDA) [8] are two types of LDA algorithms to solve this singularity problem. This paper verifies the equivalence relationship among these two LDAs.

LDA/QR was developed in [9] which is a linear discriminant analysis (LDA) extension for coping with singularity problem. It achieves the efficiency by introducing QR decomposition on a small-size matrix, while keeping competitive discriminant performance. There are two stages of process in LDA/QR. In the first stage of LDA/QR, the separation between different classes is maximized via applying QR decomposition to a small size matrix. Remind that this stage can be used independently as a dimensionality reduction algorithm and it is called Pre-LDA/QR [9]. The second stage refines the first stage by applying LDA to the so-called reduced scatter matrices [9] from the first stage. LDA/QR is efficient and effective because of QR decomposition. The solution by the LDA/QR has been proved to be equivalent to the solution generalized by pseudo-inverse LDA [6].

DLDA [8] bases on the observation that the null space of between-class scatter matrix carries no useful discriminative information. It first diagonalizes between-class scatter matrix S_b and then adopts traditional eigen-analysis to discard the zero eigenvalues together with corresponding eigenvectors, since the null space of S_b contains no useful information [2]. Finally, it diagonalizes within-class scatter matrix S_w but remains the zero eigenvalues since its null space carries the most of useful discriminant information. Hence DLDA can take advantage of all the discriminant information. The great virtue of DLDA is that it can be applied to the high-dimensional input space directly without any intermediate dimensionality reduction stage, such as PCA [4] [3] [11], which have been used in fisherfaces [1].

In this paper, we make use of LDA/QR and DLDA algorithms to solve the singularity problem in face recognition. We also show the equivalence between LDA/QR and DLDA. Ye and Li [9] have demonstrated the equivalence between LDA/QR and pseudo-inverse LDA. We also demonstrate the equivalence between DLDA and pseudo-inverse LDA. Finally, we can leverage pseudo-inverse LDA as a bridge between LDA/QR and DLDA. This completes the proof of the equivalence between LDA/QR and DLDA.

Furthermore, we show that DLDA is a special case of pseudo-inverse LDA where the inverse of between-class matrix is replaced by pseudo-inverse. We points out DLDA can also be regarded as a two-stage LDA method like LDA/QR. The first stage can be even used independently as a dimensionality reduction algorithm. Singular value decomposition (SVD) [5] is applied to the first stage of DLDA to maximize the between-class distance, whereas QR decomposition [5] is applied to LDA/QR. Hence, LDA/QR is more efficient than DLDA because QR-decomposition is faster than SVD.

We have conducted a series of experiments to compare the two LDA methods. These experiments make use of both of their first stages that are used independently as dimensionality reduction algorithms to handle the recognition on ORL [**?**] and Yale [**?**] and Yale face datasets. Our experiment results verify to be consistent with our theoretical proof.

There are three main contributions of this paper. First, we verify the equivalence between Direct LDA and LDA/QR. Second, we show that DLDA is a special case of pseudo-inverse LDA (with the pseudo-inverse is applied to between-class scatter matrix). Finally, we also demonstrate DLDA can also be regarded as a two-stage LDA method. The first stage of DLDA can be used independently as dimensionality reduction algorithm.

This paper is organized as follows. Section II reviews four LDA methods including classical LDA, pseudo-inverse LDA,

Proc. 9th IEEE Int. Conf. on Cognitive Informatics (ICCI'10)
F. Sun, Y. Wang, J. Lu, B. Zhang, W. Kinsner & L.A. Zadeh (Eds.)
978-1-4244-8040-1/10/$26.00 ©2010 IEEE

Direct LDA, and LDA/QR. In Section III, we establish the equivalence relationship between DLDA and LDA/QR by pseudo-inverse LDA together with the theoretical analysis. Experimental results and discussion are presented in Section IV. Finally, we conclude this paper in Section V.

II. Related Works

In this section, we give a brief overview of classical LDA and three extensions: pseudo-inverse LDA, Direct LDA, and LDA/QR. For convenience, Table I shows the important notations used in this paper.

TABLE I
NOTATIONS

Notation	Description
N	number of training data samples
A	data matrix
k	number of classes
m	number of dimensions
S_w	within-class scatter matrix
S_b	between-class scatter matrix
S_t	total scatter matrix
H_w	precursor of within-class scatter
H_b	precursor of between-class scatter
P	transformation matrix
d	number of retained dimensions in LDA
c_i	centroid of the i-th class
c	global centroid of the training dataset
N_i	number of data samples in the i-th class
$\prod_i$	data matrix of the i-th class
$tr(\cdot)$	trace
†	pseudo-inverse

A. Classical Linear Discriminant Analysis

Given a data matrix $A \in R^{m \times N}$, LDA aims to find a linear projection matrix $P \in \Re$ that maps each columns a_i of A, for $1 \leq i \leq N$, in the m-dimensional space to a vector $y_i \in \Re^d$ in the d-dimensional lower space as follows:

$$P : y_i = P^T a_i \tag{1}$$

Assume that the original high-dimensional A is partitioned into k classes as $A = [\prod_1, ..., \prod_k]$, where the $\prod_i \in \Re^{m \times N_i}$ contains data samples from the i-th class and $\sum_{i=1}^{k} N_i = N$. The optimal P is obtained such that the class structure of the original space is preserved in the low-dimensional space.

The so-called between-class scatter matrix S_b, within-class scatter matrix S_w, and total scatter matrix S_t in discriminant analysis can be defined as follows [LDA].

$$S_w = \frac{1}{N} \sum_{i=1}^{k} \sum_{j \in \Pi_i} (a_j - c_i)(a_j - c_i)^T, \tag{2}$$

$$S_b = \frac{1}{N} \sum_{i=1}^{k} N_i (c_i - c)(c_i - c)^T, \tag{3}$$

$$S_t = S_w + S_b = \frac{1}{N} \sum_{i=1}^{N} (a_i - c)(a_i - c)^T, \tag{4}$$

where the precursors [2] H_w, H_b, and H_t of the within-class, between-class, and total scatter matrices in Eqs.(2), (3) and (4) are:

$$H_w = \frac{1}{\sqrt{N}} (\Pi_1 - c_1 \cdot e_1^T, \cdots, \Pi_k - c_k \cdot e_k^T), \tag{5}$$

$$H_b = \frac{1}{\sqrt{N}} (\sqrt{N_1}(c_1 - c), \cdots, \sqrt{N_k}(c_k - c)), \tag{6}$$

$$H_t = \frac{1}{\sqrt{N}} (A - ce^T), \tag{7}$$

$e_i = (1, ..., 1)^T \in \Re^{N_i}$, A_i is the data matrix of the i-th class, c_i is the centroid of the i-th class, and c is the global centroid of the training dataset. It is easy to verify that: $S_w = H_w$, $S_b = H_b H_b^T$ and $S_t = H_t H_t^T$.

LDA finds the optimal projection via maximizing the within-class distance and minimizing the between-class distance simultaneously. The common optimization criterions are used in LDA as below:

$$F(P) = \arg\min_{P} \{tr((P^T S_b P)^{-1} P^T S_w P)\}$$

where the operator $tr(\cdot)$ stands for the trace of the matrix. However, classical LDA cannot be directly applied due to the singularity problem (of scatter matrix S_b). Several extensions of LDA, including pseudo-inverse LDA [5], Direct LDA, and LDA/QR were proposed in recent years to deal with this problem.

B. Pseudo-inverse Linear Discriminant Analysis

Pseudo-inverse is commonly used to deal with the singularity of matrices [6] [5]. Although the inverse of a matrix may not exist, the pseudo-inverse of any matrices are well defined. Moreover, when the matrix is invertible, its pseudo-inverse coincides with its inverse.

The generalization of classical LDA is to use pseudo-inverse to replace the inverse of the scatter matrices, where the matrix singularity do not need consideration. Therefore, it forms three new generalized optimization criterions

$$F_1(P) = \arg\min_{P} \{tr((P^T S_b P)^{\dagger} P^T S_w P)\}$$

The LDA then is implemented by the eigen-decomposition on the matrix $S_b^{\dagger} S_w$, $S_w^{\dagger} S_b$, or $S_t^{\dagger} S_b$, where † denotes the pseudo-inverse.

The pseudo-inverse of a matrix can be computed by SVD [5] as $M^{\dagger} = V\Sigma^{-1}U^T$. The new optimization criterions can be solved by a technique named simultaneous diagonalization of three scatter matrices [7].

C. Direct Linear Discriminant Analysis

Direct LDA (DLDA) was developed in [8] and can be applied directly in singularity problem. The goal is to look for a transformation matrix that simultaneously diagonalizes both between- and within-class scatter matrix S_b and S_w. The detailed procedure of DLDA is as follows:

Step 1: Diagonalize S_b.

Find an orthogonal matrix V such that $V^T S_b V = \Lambda$, where

$$\Lambda = diag(\sigma_1, \sigma_2, \cdots, \sigma_m)$$

and

$$\sigma_1 \geq \sigma_2 \geq \cdots \geq \sigma_{\mathrm{m}} \geq 0.$$

Let $V = (Y, V_1)$ be a column partition of V as $Y \in \Re^{m\times r}$ and $V_1 \in \Re^{m\times(m-r)}$, where $r = rank(S_b)$. Based on the observation that the null space of S_b carries no useful discriminative information [8]. We thus discard the zero-value diagonal elements of Λ so that we have $Y^T S_b Y = D_b$, where D_b is the $r \times r$ principle sub-matrix of Λ. Let $Z = Y D_b^{-1/2}$, obviously, $Z^T S_b Z = I_b$, where I_b is an $r \times r$ unit matrix.

Step 2: Diagonalize the matrix $Z^T S_w Z$.

Perform the eigen-analysis on the the matrix $Z^T S_w Z$:

$$U^T Z^T S_w Z U = D_w,$$

where $U^T U = I_r$, D_w is a diagonal matrix with the diagonal entries sorted in ascending order. It is easy to check that the matrix $U^T Z^T$ diagonalizes both S_b and S_w simultaneously. The transformation matrix P of Direct LDA is given by $P = ZU$.

To reduce the computational burden, the diagonalization of matrix in step 1 and step 2 can be computed by SVD [8]. The time complexity of DLDA algorithm is $O(mnk)$.

D. Linear Discriminant Analysis Via QR-Decomposition

LDA/QR is a two-stage LDA method [9]. The detailed procedure of LDA/QR is presented as follows:

Stage 1: Apply skinny QR-decomposition to the matrix H_b as $H_b = QR$, where $Q \in \Re^{m\times k}$, $R \in \Re^{k\times k}$. Let $Q = (Q_1, Q_2)$ be a column partition of Q, where $Q_1 \in \Re^{m\times r}$, $Q_2 \in \Re^{m\times(k-r)}$, then

$$H_b = (Q_1, Q_2)\begin{pmatrix} R_1 \\ 0 \end{pmatrix} = Q_1 R_1,$$

where

$$R = \begin{pmatrix} R_1 \\ 0 \end{pmatrix}$$

is a row partition of R and $R_1 \in \Re^{r\times k}$.

Stage 2: Compute the optimal transformation matrix by solving the following optimization problem:

$$G^* = \arg\min_G tr((G^T Q_1^T S_b Q_1 G)^{-1}(G^T Q_1^T S_w Q_1 G))$$

then the optimal transformation matrix of LDA/QR is $P = Q_1 G^*$. The stage 1 of LDA/QR, which named pre-LDA/QR in [9], can be used independently as a dimensionality reduction algorithm. It seeks an optimal projection vectors W which solves the following optimization problems:

$$W^* = \arg\max_{W^T W = I_r} tr(W^T S_b W) \tag{8}$$

As shown in [9], the matrix Q_1 is an optimal solution of Eq.(10). The time complexity of LDA/QR algorithm is $O(mnk)$ and the detailed analysis can be found in [9].

III. Equivalence between DLDA and LDA/QR

In this section, we have two steps to demonstrate how to establish a connection among two LDA methods, the LDA/QR and DLDA. We, therefore, have two subsections. In subsection A, the key idea is to first establish a connection between DLDA and pseudo-inverse LDA. In subsection B, we use pseudo-inverse as a bridge between the two LDA methods. Since the solutions of DLDA and LDA/QR are closely related to the eigen-decomposition on $S_b^{\dagger} S_w$, it is reasonable to prove the equivalence between DLDA and LDA/QR.

A. Equivalence between DLDA and pseudo-inverse LDA

The equivalence between DLDA and pseudo-inverse LDA bases on the theorem 1.

Theorem 1: let P be the optimal projection matrix obtained by DLDA algorithm. Then the columns of P are eigenvectors of $S_b^{\dagger} S_w$ associated with the non-zero eigenvalues.

Proof: Consider the following eigen-equation (the pseudo-inverse is applied to the between-class scatter matrix):

$$S_b^{\dagger} S_w x = \lambda x \tag{9}$$

Let

$$S_b = V\Lambda V^T = V\begin{pmatrix} D_b & 0 \\ 0 & 0 \end{pmatrix} V^T$$

be the SVD of S_b, where V and D_b are defined in Section II.C. The matrix $S_b^{\dagger}$ can be computed as follows:

$$S_b^{\dagger} = V\begin{pmatrix} D_b^{-1} & 0 \\ 0 & 0 \end{pmatrix} V^T \tag{10}$$

since S_b is a real symmetric positive semi-definite matrix. Like the procedure of DLDA, let $V = (Y, V_1)$ be a column partition of V such that $Y \in \Re^{m\times r}$ and $V_1 \in \Re^{m\times(m-r)}$, where $r = rank(S_b)$. With Eqs.(11) and (12), we then have

$$S_b^{\dagger} S_w x = (Y, V_1)\begin{pmatrix} D_b^{-1} & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} S_w x = \lambda x \tag{11}$$

Applying the same operator $V = (Y, V_1)^T$ on the left and the right of (13) simultaneously, we have

$$\begin{pmatrix} D_b^{-1} & 0 \\ 0 & 0 \end{pmatrix}\begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} S_w x = \lambda \begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} x. \tag{12}$$

Hence

$$\begin{pmatrix} D_b^{-1} Y^T S_w x \\ 0 \end{pmatrix} = \lambda \begin{pmatrix} Y^T x \\ V_1^T x \end{pmatrix}$$

Apparently, $V_1^T x = 0$. According to (14), we also have

$$\begin{pmatrix} D_b^{-1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} S_w (Y, V_1) \begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} x = \lambda \begin{pmatrix} Y^T \\ V_1^T \end{pmatrix} x$$

With $V_1^T x = 0$, we can obtain

$$D_b^{-1}(Y^T S_w Y) Y^T x = \lambda Y^T x \tag{13}$$

Multiplying the left side and right side of (15) by $D_b^{1/2}$, we can deduce

$$D_b^{-1/2}(Y^T S_w Y) D_b^{-1/2}(D_b^{1/2} Y^T x) = \lambda (D_b^{1/2} Y^T x).$$

So $D_b^{1/2} Y^T x$ is the eigenvector of $D_b^{-1/2}(Y^T S_w Y) D_b^{-1/2}$. Assume that each column of the matrix $X \in \Re^{m \times r}$ is an eigenvector corresponding to the non-zero eigenvalue of $S_b^{\dagger} S_w$, and $r = rank(S_b)$. Obviously, the eigenvectors of

$$D_b^{-1/2}(Y^T S_w Y) D_b^{-1/2}$$

form the matrix $D_b^{1/2} Y^T X$.

In DLDA, the eigenvectors of $D_b^{-1/2}(Y^T S_w Y) D_b^{-1/2}$ form the matrix U [8]. And the transformation matrix is $P = Y D_b^{-1/2} U$. Following some simple linear algebraic steps, we have $U = D_b^{1/2} Y^T P$. Hence each column of P is an eigenvector of $S_b^{\dagger} S_w$. This completes the proof of theorem 1. ■

Theorem 1 shows the equivalence relationship between DLDA and pseudo-inverse LDA, which imply that DLDA is a special case of pseudo-inverse LDA with the pseudo-inverse applied to the matrix S_b.

B. Bridging DLDA and LDA/QR Via Pseudo-inverse LDA

Pseudo-inverse LDA bridges DLDA and LDA/QR. As introduced in [9], the columns of the optimal transformation matrix solved by LDA/QR algorithm are eigenvectors of $S_b^{\dagger} S_w$ corresponding to the non-zero eigenvalues. According to theorem 1, the eigenvectors of $S_b^{\dagger} S_w$ associate with the non-zero eigenvalues form the optimal projection matrix obtained from DLDA algorithm. Thus, the optimal transformation vectors of DLDA and LDA/QR can be given by the eigenvectors of $S_b^{\dagger} S_w$ corresponding to the non-zero eigenvalues. Therefore, the subspace of DLDA that spanned by the optimal projection vectors is equivalent to the subspace of LDA/QR. In this sense, the DLDA is equivalent to LDA/QR. Moreover, both DLDA and LDA/QR can be considered as a special case of pseudo-inverse LDA, where the pseudo-inverse is applied to the between scatter matrix S_b.

C. Equivalence between Pre-DLDA and Pre-LDA/QR

Similar to LDA/QR, the DLDA can also be regarded as a two-stage method. In the first stage, it aims to find a matrix Y that diagonalize the between-class scatter matrix S_b as $S_b = Y D_b Y^T$. In the second stage, DLDA solves the following generalized eigenvalue problems:

$$(Y^T S_w Y) x = \lambda (Y^T S_b Y) x$$

the r eigenvectors corresponding to the r smallest eigenvalues of (16) form the optimal transformation matrix.

For convenience, we call the first stage of DLDA Pre-DLDA. Like the Pre-LDA/QR (the first stage of LDA/QR), pre-DLDA can also be used independently as a dimensionality reduction algorithm. In first stage of DLDA, the matrix solves the optimization problem (9) is stated by the following theorem.

Theorem 2: For arbitrary orthogonal matrix $M \in \Re^{r \times r}$, $W = YM$ solves the optimization problems (10).

Proof: It is very easy to check that $(YM)^T (YM) = I_r$. Consider

$$tr(W^T S_b W) = tr(W^T Y D_b Y^T W) \leq tr(D_b),$$

if $Y^T W$ is an orthogonal matrix, the inequality becomes equality. Hence for an arbitrary orthogonal matrix $M \in \Re^{r \times r}$, $W = YM$ solves the optimization problems in (10). This completes the proof of theorem 2. ■

According to theorem 2, the fist stage of both DLDA and LDA/QR are also equivalent. However, DLDA uses SVD to solve the (10), whereas the LDA/QR uses QR decomposition to obtain the optimal solution. QR decomposition is more efficient than SVD numerically, though the worst time complexity of QR-decomposition and SVD are equivalent. Since QR decomposition is generally considered as a substage in SVD. Hence, LDA/QR can be regarded as an efficient implementation of DLDA.

IV. Experiment Setup

In this section, we describe the data sets used in our experiments and the settings of experiments. The purpose of experiment is to verify the equivalence between the DLDA and LDA/QR. In order to achieve this, we conduct our experiments on ORL and Yale face dataset. And the experiments are tested on the PC with Corel 2 Duo 2.33 GHz processor with 3.25GB of RAM. The description of two sets of experiments is given as below.

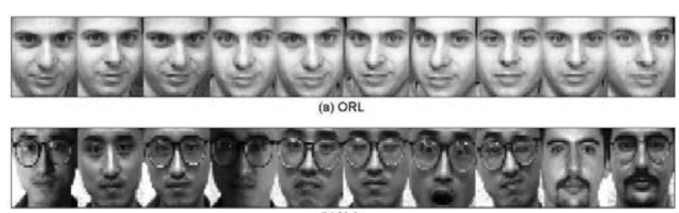

Fig. 1. Illustrations of Face Images of the two face datasets.

The ORL face dataset contains 400 facial images of 40 distinct individuals. Hence each person has ten different images. Some images were captured at different times and have different variations including expression (open or closed eyes, smiling or non-smiling) and facial details (glasses or no glasses). The images were taken with a tolerance for some tilting and rotation of the face up to 20 degrees and the size of each image is 92×92 pixels with 256 gray levels. We cropped the images to align eyes at the same position and resized them to 64×64 pixels. Figure 1a shows the some face images from ORL face dataset.

The Yale face dataset consists of 165 facial images of 15 different individuals, so each person has 11 different images. The original size of each image is 243×320 with 256 gray levels. We resized them to 64×64. Figure 1b shows the some face images from Yale face dataset.

In each face dataset, we test the classification accuracy and training time of DLDA and LDA/QR. The classification accuracy is to determine whether the DLDA is equal to LDA/QR. The training time is used to measure the efficiency between DLDA and LDA/QR. If the classification accuracy of DLDA and LDA/QR is the same, the solutions of the LDA criterion should be the same. Therefore, the same classification accuracy should be achieved in each datasets. The training time stands for the computation time of DLDA and LDA/QR to calculate the optimal projection vectors. However, DLDA and LDA/QR are two different LDA methods. LDA/QR is more efficient than DLDA. Hence, the expected training time of the DLDA should be slightly higher than LDA/QR. The detail experiments steps are described as below.

There are three main steps for testing the discriminant performance of LDA algorithms. First, we separate the datasets into training and testing dataset. We train the training sets with the projection vectors by optimizing the LDA criterion. Second, we project both training dataset and testing dataset into the optimal LDA subspace. As mentioned, the optimal LDA subspace helps to reduce the dimensionality significantly. Finally, we identify the classifiers in the data of the optimal LDA subspace. The 1-nearest neighbor classifier [10] (1-NNC) and Euclidean distance metric are adopted to be the final classifier in our experiments.

We conduct on 50 random partitions of training and test set. We select randomly as a training set for ORL and Yale face dataset. $l = (2, 3, ..., 8)$ face images per individual are randomly selected as training set and the rest forms the test set. The final results we taken is an average result of classification.

V. Results and Discussions

In this section we describe the result from our experiments. Subsection A reports the result of the face recognition accuracy. Subsection B reports the result of the training time on ORL and Yale face datasets.

A. Result of Face Recognition Accuracy

We compare the DLDA and LDA/QR, Pre-DLDA and Pre-LDA/QR by classification accuracy on ORL and Yale face datasets. The face recognition accuracy on ORL and Yale datasets are reported in Figure 2, 3 respectively.

As can be seen in these figures, the DLDA and LDA/QR algorithms achieve the same classification accuracy on both ORL and Yale face dataset, which verify our theoretical analysis. The interesting point to be stated is the Pre-DLDA and Pre-LDA/QR also obtain the same accuracy, which indicates

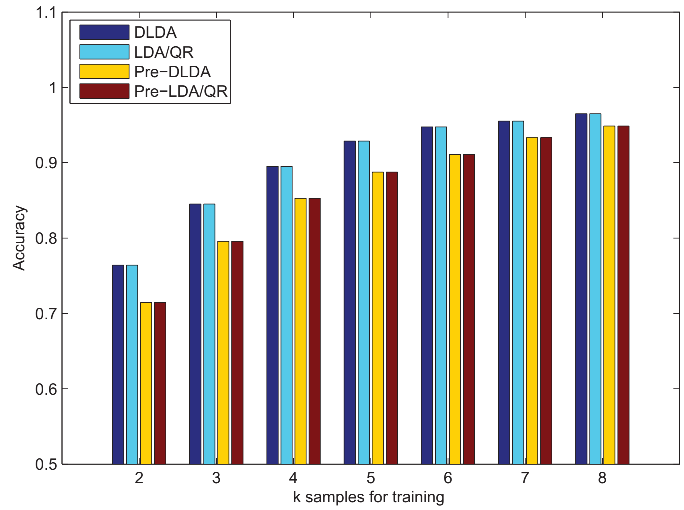

Fig. 2. Face recognition accuracy on ORL face dataset.

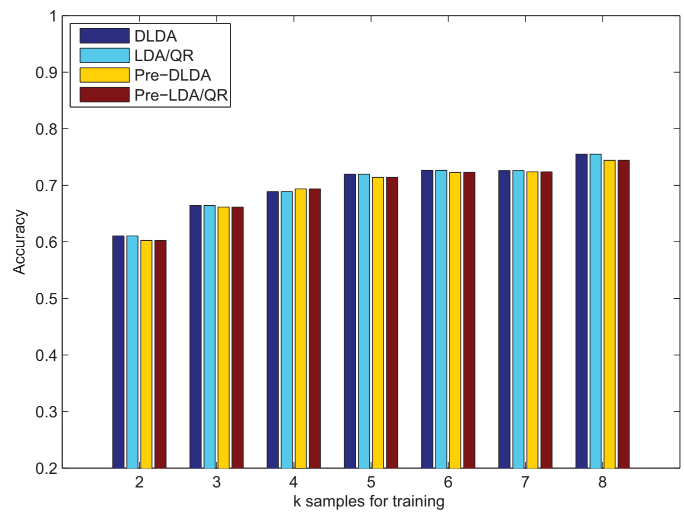

Fig. 3. Face recognition accuracy on Yale face dataset.

Pre-DLDA and Pre-LDA/QR algorithm should be equivalent. The result also indicates that the Pre-DLDA and Pre-LDA/QR typically achieve the lower classification accuracy than the DLDA and LDA/QR algorithms. This could imply to the Pre-LDA and Pre-LDA/QR algorithms only using the between-class scatter information to compute the projection vectors. However, these algorithms ignore the with-in class information, thereby achieves lower discriminant performance. Interestingly, in the Yale face dataset, the classification performance on Pre-DLDA and Pre-LDA/QR are competitive to DLDA and LDA/QR algorithms. Hence, in some case, the two first stages can be used as a dimension reduction methods independently.

B. Result of Training time on ORL and Yale face datasets

Figure 4 and 5 shows the training time on ORL and Yale face datasets respectively.

As can be seen in Figure 4 and 5, the training time of LDA/QR is distinctly lower than DLDA on ORL and Yale face dataset. The similar results are observed on Pre-LDA/QR

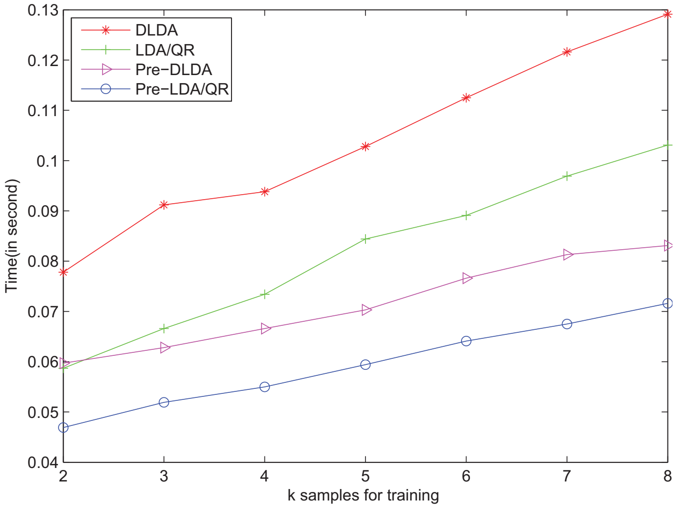

Fig. 4. Training time on ORL face datasets.

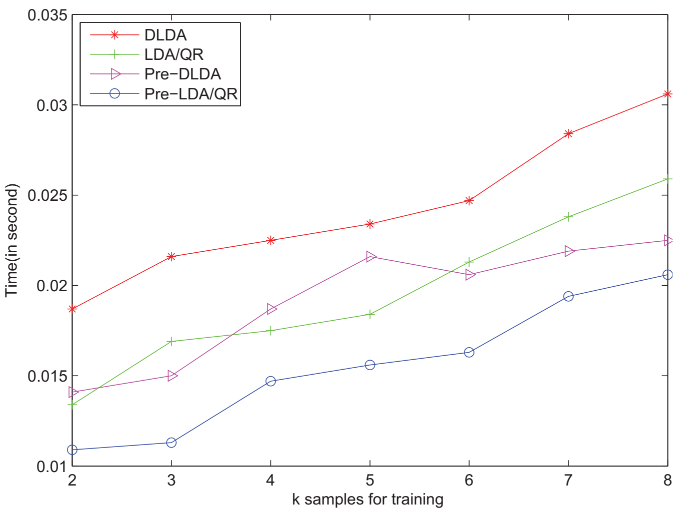

Fig. 5. Training time on Yale face datasets.

and Pre-DLDA algorithms. Recall that the LDA/QR and Pre-LDA/QR uses the QR decomposition but the DLDA and Pre-DLDA apply SVD to the between-class scatter matrix, thereby the training time of LDA/QR is lower than DLDA and Pre-LDA/QR is lower than Pre-DLDA, which coincide with the observed results completely.

VI. Conclusion

In this paper, we establish an equivalence framework between DLDA and LDA/QR for face recognition. Similar to LDA/QR, we show that the DLDA is also a two-stage LDA method. The first stage of DLDA, named Pre-DLDA, can be used independently as a dimensionality reduction algorithm. We also find that Pre-DLDA is equivalent to Pre-LDA/QR (the first stage of LDA/QR). We conduct experiments on two famous face datasets, and the classification accuracy and training time are chosen to our evaluation metrics. The experiment results show that the classification accuracy of DLDA is equal to LDA/QR while the training time of DLDA is higher than LDA/QR on all the datasets. Again, the Pre-DLDA and Pre-LDA/QR are proven to be equivalent. These empirical results confirm our theoretical analysis.

It is worth exploring in two directions. First, our theoretical analysis is based on Euclidean space, however, it also can be applied into Hilbert Reproduce Kernel Space, which has a non-linear mapping. Second, in our experiments, the first stage of DLDA and LDA/QR methods are shown to be effective in some applications with less computational time, the theoretical analysis of these algorithms could be further studied.

References

[1] P.N. Belhumeur, J. Hespanha and D.J. Kriegman, Eigenfaces vs. Fisherfaces: Recognition Using Class Specific Linear Projection, IEEE Trans. Pattern Anal. Mach. Intell., vol. 19, no. 7, pp. 711-720, 1997.
[2] L. Chen, H. Liao, M. Ko, J. Lin and G. Yu, A New LDA based Face Recognition System Which can Solve the Samll Sample Size Problem, Pattern Recognition, vol. 33, no. 10, pp. 1713-1726, 2000.
[3] M. Turk and A. Pentland, Eigenfaces for recognition, J. Cognitive Neuroscience, vol. 3, no. 1, pp. 71-86, 1991.
[4] A.M. Martinez and A.C. Kak, PCA versus LDA, IEEE Trans. Pattern Anal. Mach. Intell., vol. 23, no. 2, pp. 228-233, Feb. 2001.
[5] G.H. Golub, C.F.V. Loan, Matrix Computations, 3rd Edition, The Johns Hopkins University Press, Baltimore, MD, USA, 1996.
[6] J. Ye, R. Janardan, C. H. Park, and H. Park, An optimization criterion for generalized discriminant analysis on undersampled problems, IEEE Trans. Pattern Anal. Mach. Intell., vol. 26, no. 8, pp. 982C994, Aug. 2004.
[7] J. Ye, Characterization of a Family of Algorithms for Generalized Discriminant Analysis on Undersampled Problems, Journal of Machine Learning Research, vol. 6, pp. 483C502, 2005.
[8] H. Yu and J. Yang, A Direct LDA Algorithm for High Dimensional Data with Application to Face Recognition, Pattern Recognition, vol. 34, no. 10, pp. 2067-2070, 2001.
[9] J. Ye, Q, Li. “A Two-Stage Linear Discriminant Analysis via QR-Decomposition.” IEEE Trans. Pattern Anal. Mach. Intell., vol. 27, no. 6, pp. 929-941, 2005.
[10] T. Cover and P. Hart, Nearest neighbor pattern classification, IEEE Transactions on Information Theory, vol. 13, no. 1, pp. 21C27, 1967.
[11] J. Yang and J.Y. Yang, Why Can LDA Be Performed in PCA Transformed Space? Pattern Recognition, vol. 36, no. 2, pp. 563-566, 2003.
[12] [Online] http://www.uk.research.att.com/facedatabase.html
[13] [Online] http://cvc.yale.edu/projects/yalefaces/yalefaces.html