

Tensor Locally Linear Discriminative Analysis

Zhao Zhang, *Student Member, IEEE*, and Tommy W. S. Chow, *Senior Member, IEEE*

Abstract— In this letter, we present a **Tensor Locally Linear Discriminative Analysis (TLLDA)** method for image presentation and feature extraction. TLLDA is formulated based on the **Local Fisher Discriminant Analysis (LFDA)**, but TLLDA offers some attractive advantages over LFDA. First, TLLDA can preserve the local discriminative information of image data as LFDA. Second, TLLDA represents images as matrices or 2-order tensors rather than vectors, so TLLDA is able to exhibit natural representations of images and can keep the spatial locality of pixels in the images. Third, TLLDA can avoid the singularity may suffered by LFDA. Fourth, TLLDA is faster than LFDA. Comparative simulations on real UMIST and MIT CBCL databases are conducted to verify the validity of the TLLDA. The results show that TLLDA is highly competitive with some widely used state-of-the-art techniques.

Index Terms— Tensor representation; Discriminant analysis; Dimensionality reduction; Trace ratio optimization

I. INTRODUCTION

THE *Principal Component Analysis* (PCA) [1], *Fisher Linear Discriminant Analysis* (LDA) [1] and *Locality Preserving Projections* (LPP) [2] are the three most popular dimensionality reduction (DR) methods. Based on the ideas of LDA and LPP, an efficient preservation plus discriminant technique called *Local Fisher Discriminant Analysis* (LFDA) [3] was proposed for multimodal DR. LFDA can preserve the local property and discriminant information of data effectively. Note that the above methods all perform in vector space, extracting the features from the vector patterns directly. To learn such a LDA, PCA, LPP or LFDA subspace, images firstly need to be transformed into the 1D vectors in high-dimensional vector space. But images are intrinsically matrices or 2-order tensors. To address this issue, multi-dimensional extensions of PCA, LDA and LPP, namely 2DPCA [11], 2DLDA [10] and tensorized LPP (TLPP) [7], are presented. These methods aim to compute the subspaces directly from the 2D image matrices rather than 1D vectors.

This letter represents images of size $n_1 \times n_2$ as matrices in tensor space $\mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2}$. We then propose the tensorized LFDA, which we call *Tensor Locally Linear Discriminative Analysis* (TLLDA). Compared with LFDA, TLLDA has the following advantages. (1) TLLDA gives natural representations of images. (2) TLLDA is computationally efficient, since the decomposed matrices are of size $n_1 \times n_1$ or $n_2 \times n_2$, which is much smaller than that of size $n \times n$ ($n = n_1 \times n_2$) in LFDA. (3) TLLDA avoids the matrix singularity [6]. The trace ratio optimization [8][9][12] is

employed to solve TLLDA enabling TLLDA to deliver more specific solution according to the orthogonal constraints.

The outline of this letter is described as follows. Section II reviews LFDA. Section III shows our method. In Section IV, we conduct simulations. The conclusion is given in Section V.

II. LOCAL FISHER DISCRIMINANT ANALYSIS (LFDA)

Let $x_i \in \mathbb{R}^n$ ($i = 1, 2, \dots, m$) denote vectors of n -dimensional data and $y_i (\in \{1, 2, \dots, c\})$ be the class labels, where c is the class number, then LFDA finds a $n \times d$ projection matrix Ξ and projects each x_i into low-dimensional representation $\tilde{x}_i = \Xi^T x_i$, where notation T is the transpose of a matrix or vector. Let $\tilde{A}^{(w,c)}$ and $\tilde{A}^{(b,c)}$ denote the local within- and between-class scatters of LFDA respectively, then we have

$$\tilde{S}^{(w,c)} = \frac{1}{2} \sum_{i,j=1}^m (x_i - x_j)(x_i - x_j)^T \tilde{P}_{i,j}^{(w,c)}, \tilde{S}^{(b,c)} = \frac{1}{2} \sum_{i,j=1}^m (x_i - x_j)(x_i - x_j)^T \tilde{P}_{i,j}^{(b,c)}, \quad (1)$$

where $\|\cdot\|$ is the Euclidean distance. Let m_l be the number of points in class l , then matrices $\tilde{P}^{(w,c)}$ and $\tilde{P}^{(b,c)}$ are defined as

$$\tilde{P}_{i,j}^{(w,c)} = \begin{cases} W_{i,j} / m_l & \text{if } y_i = y_j = l \\ 0 & \text{else if } y_i \neq y_j \end{cases}, \quad (2a)$$

$$\tilde{P}_{i,j}^{(b,c)} = \begin{cases} W_{i,j} / (1/m - 1/m_l) & \text{if } y_i = y_j = l \\ 1/m & \text{else if } y_i \neq y_j \end{cases}, \quad (2b)$$

where W represents the similarity between each paired x_i and x_j . W can be defined by the simple-minded method [4], heat kernel [4] or local scaling heuristic method [5]. The value $W_{i,j}$ is large if x_i and x_j are neighbors, and else small. The neighbors of x_i is determined by the k -nearest neighbor search [4]. Then the objective function of LFDA can be defined as

$$\underset{\Xi \in \mathbb{R}^{n \times d}}{\text{Max}} \text{Tr}(\Xi^T \tilde{A}^{(b,c)} \Xi), \text{s.t. } \Xi^T \tilde{A}^{(w,c)} \Xi = I, \quad (3)$$

where I is an identity matrix, and $\text{Tr}(\cdot)$ is matrix trace. Then Ξ can be obtained by solving the following problem:

$$\Xi^* = \underset{\Xi \in \mathbb{R}^{n \times d}}{\text{argmax}} \text{Tr} \left[\left(\Xi^T \left(\tilde{A}^{(b,c)} + \beta I \right) \Xi \right)^{-1} \left(\Xi^T \tilde{A}^{(w,c)} \Xi \right) \right], \quad (4)$$

where βI with $\beta = 0.01$ is added to avoid the singularity in this letter. Thus the LFDA projection axes are not orthogonal.

III. TENSOR LOCALLY LINEAR DISCRIMINATIVE ANALYSIS

A. Matrix (or, Tensor) Interpretation of Images

Let $x \in \mathbb{R}^{n_1 \times n_2}$ be an image of size $n_1 \times n_2$, then a second order tensor of x can be mathematically formulated as

$$x = \sum_{i,j} u_i^T x v_j u_i v_j^T \quad (5)$$

in tensor space $\mathbb{R}^{n_1} \otimes \mathbb{R}^{n_2}$, where $(u_1, u_2, \dots, u_{n_1})$ is a set of basis functions in \mathbb{R}^{n_1} and $(v_1, v_2, \dots, v_{n_2})$ is a set of basis functions in

Manuscript received June 29, 2011; revised August 01, 2011; accepted August 07, 2011. Date of publication August 22, 2011; date of current version September 22, 2011. The associate editor coordinating the review of this manuscript and approving it for publication was Prof. Yong Man Ro.

Z. Zhang and Tommy W.S. Chow are with the Department of Electronic Engineering, City University of Hong Kong, 83 Tat Chee Avenue, Kowloon Hong Kong. E-mails: itzzhang@ee.cityu.edu.hk, eetchow@cityu.edu.hk.

Digital Object Identifier 10.1109/LSP.2011.2165538

\mathbb{R}^{n_2} [7][12]. So $\{u_i v_j^T\}$ forms a basis of tensor space $\mathbb{R}^n \otimes \mathbb{R}^{n_2}$. Let U be a subspace of \mathbb{R}^n spanned by $\{u_i\}_{i=1}^{d_1}$ and V be a subspace of \mathbb{R}^{n_2} spanned by $\{v_i\}_{i=1}^{d_2}$, then tensor product $U \otimes V$ is a subspace of $\mathbb{R}^n \otimes \mathbb{R}^{n_2}$. By projecting each $x_i \in \mathbb{R}^{n \times n_2}$ onto $U \otimes V$, we obtain tensors $\eta_i = U^T x_i V \in \mathbb{R}^{d_1 \times d_2}$ ($d_1 \leq n_1, d_2 \leq n_2$).

B. The Objective Function

Let $\tilde{\Upsilon}^{(lbc)}$ and $\tilde{\Upsilon}^{(hwc)}$ denote the inter- and intra-class scatters of TLLDA, we have the following expressions:

$$Tr(\tilde{\Upsilon}^{(hwc)}) = \frac{1}{2} \sum_{i,j=1}^m \|\eta_i - \eta_j\|^2 \tilde{P}_{i,j}^{(hwc)}, \quad Tr(\tilde{\Upsilon}^{(lbc)}) = \frac{1}{2} \sum_{i,j=1}^m \|\eta_i - \eta_j\|^2 \tilde{P}_{i,j}^{(lbc)}, \quad (6)$$

where $\eta_i = U^T x_i V$ and $\eta_j = U^T x_j V$, thus $\tilde{\Upsilon}^{(lbc)} = U^T \tilde{S}^{(lbc)} V$ and $\tilde{\Upsilon}^{(hwc)} = U^T \tilde{S}^{(hwc)} V$. The entries of weight matrices $\tilde{P}^{(hwc)}$ and $\tilde{P}^{(lbc)}$ for measuring the similarities are defined as

$$\tilde{P}_{i,j}^{(hwc)} = \begin{cases} W_{i,j} / \sqrt{m} & \text{if } y_i = y_j \\ 0 & \text{else if } y_i \neq y_j \end{cases}, \quad (7a)$$

$$\tilde{P}_{i,j}^{(lbc)} = \begin{cases} W_{i,j} / (1/\sqrt{2m} - 1/\sqrt{m}) & \text{if } y_i = y_j \\ 1/\sqrt{m} & \text{else if } y_i \neq y_j \end{cases}, \quad (7b)$$

where W is similarly defined as LFDA. Then the objective function of TLLDA can be formulated as the following trace ratio criterion based problem [8][9]:

$$(U, V) = \arg \max_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(\tilde{\Upsilon}^{(lbc)})}{Tr(\tilde{\Upsilon}^{(hwc)})} = \arg \max_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(U^T \tilde{A}^{(lbc)} V)}{Tr(U^T \tilde{A}^{(hwc)} V)}, \quad (8)$$

from which the transforming axes U and V can be computed and then the tensor subspace $U \otimes V$ can be constructed.

C. Computational Analysis

Let $D^{(bc)}$ be a diagonal matrix with entries being $D_{ii}^{(bc)} = \sum_j \tilde{P}_{i,j}^{(bc)}$. Since $\|A\|^2 = Tr(AA^T)$, we can obtain

$$Tr(\tilde{\Upsilon}^{(hwc)}) = \frac{1}{2} \sum_{i,j=1}^m \|\eta_i - \eta_j\|^2 \tilde{P}_{i,j}^{(hwc)} = \frac{1}{2} \sum_{i,j=1}^m Tr((\eta_i - \eta_j)(\eta_i - \eta_j)^T) \tilde{P}_{i,j}^{(hwc)} \quad (9)$$

$$= Tr\left(\sum_i D_{ii}^{(hwc)} \eta_i \eta_i^T - \sum_{i,j} \tilde{P}_{i,j}^{(hwc)} \eta_i \eta_j^T\right)$$

By substituting $\eta_i = U^T x_i V$ and $\eta_j = U^T x_j V$ into (9), we can get the following formulation from (9):

$$Tr(\tilde{\Upsilon}_+^{(hwc)}) = Tr\left(\sum_i D_{ii}^{(hwc)} U^T x_i V V^T x_i^T U - \sum_{i,j} \tilde{P}_{i,j}^{(hwc)} U^T x_i V V^T x_j^T U\right), \quad (10)$$

$$= Tr\left(U^T (F_V^{(hwc)} - H_V^{(hwc)}) U\right)$$

where $F_V^{(hwc)} = \sum_i D_{ii}^{(hwc)} x_i V V^T x_i^T$, $H_V^{(hwc)} = \sum_{i,j} \tilde{P}_{i,j}^{(hwc)} x_i V V^T x_j^T$. Similarly the inter-class scatter $\tilde{\Upsilon}^{(lbc)}$ can be transformed to

$$Tr(\tilde{\Upsilon}_+^{(lbc)}) = Tr\left(\sum_i D_{ii}^{(lbc)} U^T x_i V V^T x_i^T U - \sum_{i,j} \tilde{P}_{i,j}^{(lbc)} U^T x_i V V^T x_j^T U\right), \quad (11)$$

$$= Tr\left(U^T (F_V^{(lbc)} - H_V^{(lbc)}) U\right)$$

where $D_{ii}^{(bc)} = \sum_j \tilde{P}_{i,j}^{(bc)}$, $F_V^{(bc)} = \sum D_{ii}^{(bc)} x_i V V^T x_i^T$, $H_V^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} x_i V V^T x_j^T$. Due to the property of matrix trace, we have $\|A\|^2 = tr(A^T A)$, thus the traces of the matrices $\tilde{\Upsilon}^{(hwc)}$ and $\tilde{\Upsilon}^{(lbc)}$ can be converted to

$$Tr(\tilde{\Upsilon}_-^{(hwc)}) = Tr\left(\sum_i D_{ii}^{(hwc)} V^T x_i^T U U^T x_i V - \sum_{i,j} \tilde{P}_{i,j}^{(hwc)} V^T x_i^T U U^T x_j V\right), \quad (12)$$

$$= Tr\left(V^T (F_U^{(hwc)} - H_U^{(hwc)}) V\right)$$

$$Tr(\tilde{\Upsilon}_-^{(lbc)}) = Tr\left(\sum_i D_{ii}^{(lbc)} V^T x_i^T U U^T x_i V - \sum_{i,j} \tilde{P}_{i,j}^{(lbc)} V^T x_i^T U U^T x_j V\right), \quad (13)$$

$$= Tr\left(V^T (F_U^{(lbc)} - H_U^{(lbc)}) V\right)$$

where $F_U^{(bc)} = \sum D_{ii}^{(bc)} x_i^T U U^T x_i$, $H_U^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} x_i^T U U^T x_j$, $F_U^{(bc)} = \sum D_{ii}^{(bc)} x_i^T U U^T x_i$, $H_U^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} x_i^T U U^T x_j$. Then we obtain

$$\text{Max}_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(\tilde{\Upsilon}_+^{(lbc)})}{Tr(\tilde{\Upsilon}_+^{(hwc)})} = \text{Max}_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(U^T (F_V^{(lbc)} - H_V^{(lbc)}) U)}{Tr(U^T (F_V^{(hwc)} - H_V^{(hwc)}) U)}, \quad (14)$$

$$\text{Max}_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(\tilde{\Upsilon}_-^{(lbc)})}{Tr(\tilde{\Upsilon}_-^{(hwc)})} = \text{Max}_{U \in \mathbb{R}^{n \times d_1}, V \in \mathbb{R}^{n_2 \times d_2}} \frac{Tr(V^T (F_U^{(lbc)} - H_U^{(lbc)}) V)}{Tr(V^T (F_U^{(hwc)} - H_U^{(hwc)}) V)}. \quad (15)$$

Note that the above problems are all subjected to U and V , so the above problems can not be solved independently. In this work, we adopt the similar computational method as [7] to compute the matrices U and V . By setting the matrix U to be the identity matrix and let $\tilde{x}_i = x_i^T U$, we obtain

$$\tilde{F}_U^{(bc)} = \sum_i D_{ii}^{(bc)} \tilde{x}_i \tilde{x}_i^T, \quad \tilde{H}_U^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \tilde{x}_i \tilde{x}_j^T, \quad (16)$$

$$\tilde{F}_U^{(bc)} = \sum_i D_{ii}^{(bc)} \tilde{x}_i \tilde{x}_i^T, \quad \tilde{H}_U^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \tilde{x}_i \tilde{x}_j^T$$

from which we see $\tilde{F}_U^{(bc)}$, $\tilde{H}_U^{(bc)}$, $\tilde{F}_U^{(bc)}$ and $\tilde{H}_U^{(bc)}$ are simplified independently only depending on the data points. Thus the optimization problem in (14) can be simplified as

$$\text{Max}_{V \in \mathbb{R}^{n_2 \times d_2}, V^T V = I_{d_2 \times d_2}} \frac{Tr\left(V^T \left(\sum_i D_{ii}^{(bc)} \tilde{x}_i \tilde{x}_i^T - \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \tilde{x}_i \tilde{x}_j^T\right) V\right)}{Tr\left(V^T \left(\sum_i D_{ii}^{(bc)} \tilde{x}_i \tilde{x}_i^T - \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \tilde{x}_i \tilde{x}_j^T\right) V\right)} \quad (17)$$

with respect to $V^T V = I_{d_2 \times d_2}$, from which V can be computed using the trace ratio optimization [8][9]. After the matrix V is obtained, we similarly set $\hat{x}_i = x_i V$, then $F_V^{(bc)}$, $H_V^{(bc)}$, $F_V^{(bc)}$ and $H_V^{(bc)}$ can also be simplified as

$$\tilde{F}_V^{(bc)} = \sum_i D_{ii}^{(bc)} \hat{x}_i \hat{x}_i^T, \quad \tilde{H}_V^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \hat{x}_i \hat{x}_j^T, \quad (18)$$

$$\tilde{F}_V^{(bc)} = \sum_i D_{ii}^{(bc)} \hat{x}_i \hat{x}_i^T, \quad \tilde{H}_V^{(bc)} = \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \hat{x}_i \hat{x}_j^T$$

Similarly the problem in (15) can be transformed to

$$\text{Max}_{U \in \mathbb{R}^{n \times d_1}, U^T U = I_{d_1 \times d_1}} \frac{Tr\left(U^T \left(\sum_i D_{ii}^{(bc)} \hat{x}_i \hat{x}_i^T - \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \hat{x}_i \hat{x}_j^T\right) U\right)}{Tr\left(U^T \left(\sum_i D_{ii}^{(bc)} \hat{x}_i \hat{x}_i^T - \sum_{i,j} \tilde{P}_{i,j}^{(bc)} \hat{x}_i \hat{x}_j^T\right) U\right)} \quad (19)$$

subject to $U^T U = I_{d_1 \times d_1}$. Thus the matrix U can be similarly achieved by using the trace ratio optimization approach [8][9].

D. Effective Solution Using Trace Ratio Optimization

In this section, we show how to compute the matrices U and V by the iterative trace ratio optimization method called ITR in [8]. We consider the following trace ratio (TR) problem:

$$\text{Max}_{\Theta^T \Theta = I} Tr\left(\Theta^T \tilde{L}^{(bc)} \Theta\right) / Tr\left(\Theta^T \tilde{L}^{(wc)} \Theta\right). \quad (20)$$

Then ITR tackles this TR problem by directly optimizing the objective function $Tr(\Theta^T \tilde{L}^{(bc)} \Theta) / Tr(\Theta^T \tilde{L}^{(wc)} \Theta)$ by assuming that the column vectors of matrix Θ are unitary and orthogonal together. Given TR value λ^v at each iteration v , the matrix Θ^v can be obtained by the following trace difference problem:

$$\Theta^v = \arg \max_{\Theta^T \Theta = I} Tr(\Theta^T (\tilde{L}^{(bc)} - \lambda^v \tilde{L}^{(wc)}) \Theta), \quad (21)$$

and renew λ^{v+1} as the trace ratio value which is given by Θ^v : $\lambda^{v+1} = Tr(\Theta^{vT} \tilde{L}^{(bc)} \Theta^v) / Tr(\Theta^{vT} \tilde{L}^{(wc)} \Theta^v)$ until converges to the global optimum [9]. Note that Θ^0 was initialized as an arbitrary orthogonal matrix in ITR so ITR maybe unstable due to the randomness. Most importantly, the orthogonal initialized Θ^0 is difficult and a bad initialization may greatly increase the number of iterations. This letter initializes $\lambda^0 = 0$ instead of initializing Θ^0 to be a columnly orthogonal matrix as [9]. The algorithmic procedures can be described as follows:

- 1) Initialize $\lambda^v = 0$, step $v = 0$;
- 2) Solve eigen-problem $(\tilde{L}^{(bc)} - \lambda^v \tilde{L}^{(wc)}) \pi = \tau^v \pi$ and calculate the vectors $\{\pi_\delta^v\}_{\delta=1}^d$ of $(\tilde{L}^{(bc)} - \lambda^v \tilde{L}^{(wc)})$ by eigen-decomposition;
- 3) Matrix $\Theta^v = \{\pi_\delta^v\}_{\delta=1}^d$ is formed by the eigenvectors according to the d leading eigenvalues $\{\tau_\delta^v\}_{\delta=1}^d$ of $(\tilde{L}^{(bc)} - \lambda^v \tilde{L}^{(wc)})$;
- 4) Renew λ^{v+1} by $Tr((\Theta^v)^T \tilde{L}^{(bc)} \Theta^v) / Tr((\Theta^v)^T \tilde{L}^{(wc)} \Theta^v)$;
- 5) If $|\lambda^{v+1} - \lambda^v| < \varepsilon$, go to step 6; else $v = v + 1$, steps 2-4 repeat;
- 6) Output the optimal TR value $\lambda^* = \lambda^v$, and projection matrix $\Theta^* = \arg \max_{\Theta^T \Theta = I} Tr(\Theta^T (\tilde{L}^{(bc)} - \lambda^* \tilde{L}^{(wc)}) \Theta)$.

To achieve the transforming matrix V , we firstly fix matrix U . Let $\tilde{L}_U^{(bc)} = \tilde{F}_U^{(bc)} - \tilde{H}_U^{(bc)}$, $\tilde{L}_U^{(wc)} = \tilde{F}_U^{(wc)} - \tilde{H}_U^{(wc)}$, $\tilde{L}_V^{(bc)} = \tilde{F}_V^{(bc)} - \tilde{H}_V^{(bc)}$, $\tilde{L}_V^{(wc)} = \tilde{F}_V^{(wc)} - \tilde{H}_V^{(wc)}$, by setting $\tilde{L}^{(bc)} = \tilde{L}_U^{(bc)}$ and $\tilde{L}^{(wc)} = \tilde{L}_U^{(wc)}$, then optimal V can be obtained by using the above procedures. After V is computed, U can be similarly achieved by setting $\tilde{L}^{(bc)} = \tilde{L}_V^{(bc)}$ and $\tilde{L}^{(wc)} = \tilde{L}_V^{(wc)}$. Note that under TR criterion, the obtained transformation matrix is orthogonal [8][9].

IV. SIMULATION RESULTS AND ANALYSIS

A. Simulation Setting and Data Preparation

In this section, we conduct the simulations to verify the effectiveness of our proposed TLLFDA. The performance is compared with that of 2DPCA, 2DLDA and TLPP. When setting the weights for TLPP, LFDA and TLLDA, the heat kernel, i.e. $\exp(-\|x_i - x_j\|/t)$, is used, where t is determined by calculating all pairwise distances among points in the whole training set. The number of neighbors, k , is set to 11. Two real face databases are tested. The first one is the MIT CBCL face database (Available at <http://cbcl.mit.edu/software-datasets/heisele/download/download.html>). The second one is UMIST database (Available at <http://www.sheffield.ac.uk/eee/research/iel/research/face>). The face images are all re-sized into 32×32 pixels. This letter considers the case: $n_1 = n_2, d_1 = d_2$. In solving the optimal subspaces of TLPP and TLLDA, all the images are represented as (32×32) -dimensional matrices. Then new faces are projected into the $(d \times d)$ -dimensional tensor subspace. For face recognition, the one-nearest-neighbor (1NN) classifier with

Euclidean metric is used. For each tested case, the results are averaged over 10 random splits of training and test samples.

B. Face Recognition on MIT CBCL Database

The synthetic sample set from the MIT CBCL database is tested. In this set, there are 324 images of one of 10 individuals. In our study, 200 images per individual (totally 2000 images) are selected for simulations. For each individual, $L (= 3, 6, 9, 12)$ images are selected for training the face subspaces and the rest are used for testing. Figure 1 plots the accuracy of each method vs. reduced dimensions. For each method, we only show its performance in the d - or $(d \times d)$ -dimensional subspace, where $d=1, 2, \dots, 32$. Seeing from Figure 1, the performance of all the methods varies with the reduced dimensions. We also show the mean and best result of each method in Table 1. The subspaces according to the best records are called optimal face subspaces. Observing from Figure 1 and Table 1, we conclude that our proposed TLLDA delivers the highest accuracies in most cases. 2DPCA and TLPP perform comparatively to our method, whilst 2DLDA performs poorly on this dataset.

C. Face Recognition on UMIST Database

This study tests 2DPCA, 2DLDA, TLPP and TLLDA using the UMIST database. This database consists of 564 images of 20 persons. For each individual, $L (= 2, 3, 4, 5)$ images are selected for training and the rest are for testing. Figure 2 exhibits the results over different reduced dimensions. For each method, we show its performance in the d - or $(d \times d)$ -dimensional subspace, where $d=1, 2, \dots, 32$. The mean and best result by each method are list in Table 2. We have the following observations. (1) The performance of each method varies with increasing dimensions. As the number of dimensions increases, the result of 2DPCA is monotonically increasing, but the increasing dimensions cause the 2DLDA, TLPP and TLLDA results to deteriorate. (2) Our TLLDA outperforms the other methods in most cases. TLPP is worse for each case. 2DPCA and 2DLDA are comparable. As the training sample size increases, the accuracy of 2DLDA increases faster than 2DPCA and TLPP.

We also compare TLLDA with LFDA. The same settings are tested. We compute d -dimensional vector subspace for LFDA. The results are given in Table 3. We find TLLDA outperforms LFDA. The running time is significantly reduced by TLLDA.

V. CONCLUDING REMARKS

This letter introduces a new technology called Tensor Locally Linear Discriminative Analysis for image feature extraction. The key difference between TLLDA and LFDA is that TLLDA adopts the matrix representation of images, enabling TLLDA to extract the features from 2D images directly. TLLDA has the analytic solution of projection matrix as LFDA. We investigate TLLDA through simulations over two real databases. From all our investigated cases, the overall performance of TLLDA is comparable or even outperforms some state-of-the-art methods, including LFDA. We also experimentally observe that TLLDA is more computationally efficient than LFDA.

REFERENCES

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

[2] X. F. He, S. C. Yan, Y. X. Hu, P. Niyogi, and H. J. Zhang, "Face Recognition Using Laplacianfaces," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol.27, no.3, pp.328-340, 2005.

[3] M. Sugiyama, "Dimensionality reduction of multimodal labeled data by local fisher discriminant analysis," *J Mach Learn Res*, vol.8, pp.1027-1061, 2007.

[4] M. Belkin and P. Niyogi, "Laplacian Eigenmaps for Dimensionality Reduction and Data Representation," *Neural Comput.*, vol.15, no.6, pp.1373-1396, 2003.

[5] L. Zelnik-Manor and P. Perona, "Self-tuning spectral clustering," In L. K. Saul, Y. Weiss, and L. Bottou, editors. In: *NIPS*, pp.1601-1608, 2005.

[6] F. P. Nie, S. Xiang, and C. S. Zhang, "Neighborhood MinMax Projections," In: *Proceedings of IJCAI*, San Francisco, CA, USA, pp.993-998, 2007.

[7] X. He, D. Cai, and P. Niyogi, "Tensor Subspace Analysis," In: *Advances in Neural Information Processing Systems*. Vancouver, Canada, 2005.

[8] H. Wang, S. Yan, D. Xu, X. Tang, and T. Huang, "Trace ratio vs. ratio trace for dimensionality reduction," In: *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Minneapolis, MN, pp.1-8, 2007.

[9] Y. Jia, F. Nie and C.S. Zhang, "Trace ratio problem revisited," *IEEE Trans. on Neural Network*, vol.20, no.4, pp.729-735, 2008.

[10] H. Kong, E. K. Teoh, J. G. Wang, R. Venkateswarlu, "Two Dimensional Fisher Discriminant Analysis: Forget about Small Sample Size Problem," In: *Proceedings of ICASSP*, pp.761-764, 2005.

[11] J. Yang, D. Zhang, A. F. Frangi, and J.Y. Yang, "Two-Dimensional PCA: A New Approach to Appearance-Based Face Representation and Recognition," *IEEE Trans. on Pattern Anal. Mach. Intell.*, vol.26, no.4, pp. 131-137, 2004.

[12] F. P. Nie, S. Xiang, Y. Q. Song, and C. S. Zhang, "Extracting the Optimal Dimensionality for Local Tensor Discriminant Analysis," *Pattern Recognition*, 42(1), pp.105-114, 2009.

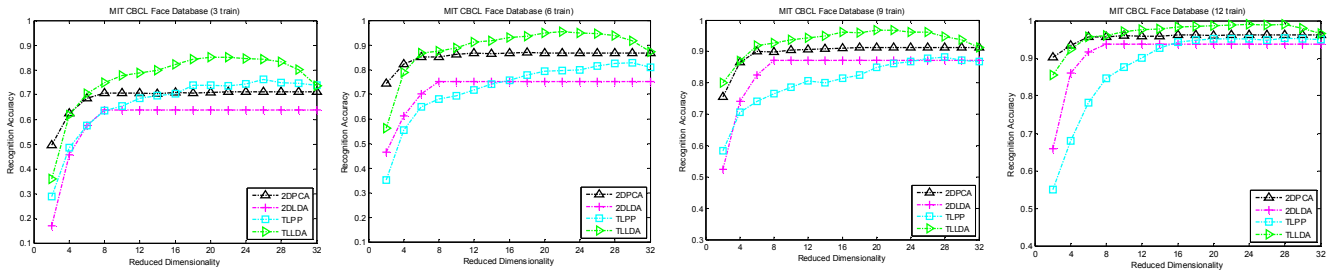


Figure 1. Recognition accuracy vs. reduced dimensionality on the MIT CBCL face database.

Table 1. Performance comparison on the MIT CBCL face database.

Method \ Result	MIT CBCL (3 train)			MIT CBCL (6 train)			MIT CBCL (9 train)			MIT CBCL (12 train)		
	Mean	Best	Dim	Mean	Best	Dim	Mean	Best	Dim	Mean	Best	Dim
2DPCA	0.6892	0.7122	24	0.8546	0.8701	18	0.8971	0.9129	18	0.9554	0.9617	22
2DLDA	0.5935	0.6382	8	0.7205	0.7501	8	0.8393	0.8723	8	0.9134	0.9368	8
TLPP	0.6671	0.7607	26×26	0.7240	0.8271	30×30	0.8068	0.8828	28×28	0.8821	0.9527	28×28
TLLDA	0.7637	0.8525	20×20	0.8877	0.9534	22×22	0.9329	0.9679	20×20	0.9676	0.9899	28×28

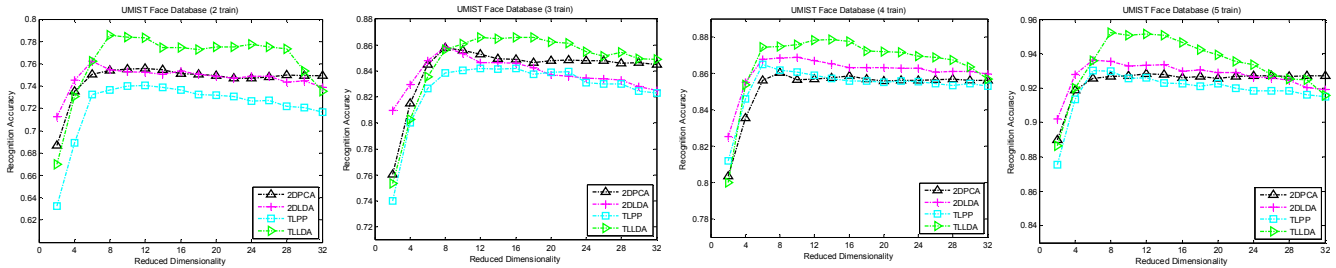


Figure 2. Recognition accuracy vs. reduced dimensionality on the UMIST face database.

Table 2. Performance comparison on the UMIST face database.

Method \ Result	UMIST Face (2 train)			UMIST Face (3 train)			UMIST Face (4 train)			UMIST Face (5 train)		
	Mean	Best	Dim	Mean	Best	Dim	Mean	Best	Dim	Mean	Best	Dim
2DPCA	0.7460	0.7558	12	0.8413	0.8579	8	0.8523	0.8605	8	0.9210	0.9283	12
2DLDA	0.7473	0.7623	6	0.8380	0.8582	8	0.8610	0.8689	10	0.9254	0.9329	6
TLPP	0.7221	0.7404	12×12	0.8266	0.8421	12×12	0.8534	0.8648	6×6	0.9175	0.9310	6×6
TLLDA	0.7627	0.7857	8×8	0.8471	0.8659	16×16	0.8660	0.8788	14×14	0.9368	0.9521	8×8

Table 3. Performance comparison of LFDA and TLLDA on the UMIST face database.

Method \ Result	UMIST Face (2 train)			UMIST Face (3 train)			UMIST Face (4 train)			UMIST Face (5 train)		
	Mean	Best	Time	Mean	Best	Time	Mean	Best	Time	Mean	Best	Time
LFDA	0.7370	0.7607	4.5204	0.8215	0.8478	4.6774	0.8492	0.8659	4.9276	0.9206	0.9295	5.1615
TLLDA	0.7627	0.7857	0.2677	0.8471	0.8659	0.3106	0.8660	0.8788	0.3927	0.9368	0.9521	0.4689