# Discriminant Locality Preserving Projections Based on L1-Norm Maximization

Fujin Zhong, Jiashu Zhang, and Defang Li

***Abstract*— Conventional discriminant locality preserving projection (DLPP) is a dimensionality reduction technique based on manifold learning, which has demonstrated good performance in pattern recognition. However, because its objective function is based on the distance criterion using L2-norm, conventional DLPP is not robust to outliers which are present in many applications. This paper proposes an effective and robust DLPP version based on L1-norm maximization, which learns a set of local optimal projection vectors by maximizing the ratio of the L1-norm-based locality preserving between-class dispersion and the L1-norm-based locality preserving within-class dispersion. The proposed method is proven to be feasible and also robust to outliers while overcoming the small sample size problem. The experimental results on artificial datasets, Binary Alphadigits dataset, FERET face dataset and PolyU palmprint dataset have demonstrated the effectiveness of the proposed method.**

***Index Terms*— Discriminant locality preserving projections, L1-norm, L2-norm, optimization, outliers.**

## I. Introduction

THERE are many dimensionality reduction techniques used to reduce the number of input variables to simplify data analysis problems, which play an important role in machine learning, information retrieval, pattern recognition, and so on. Among them, linear approaches have demonstrated excellent performance in many fields such as face recognition, handprint recognition, human action recognition, and even generic object recognition [1]–[8]. Principal component analysis (PCA) [9] and linear discriminant analysis (LDA) [LDA, also known as Fisher discriminant analysis (FDA)] [10] are the two most well-known classical linear techniques. PCA is an unsupervised approach which learns a set of projection vectors so that the variance of given data in feature space is maximized. These projection vectors constitute a low-dimensional linear subspace by which the most representative feature information about original samples can be effectively captured. But PCA deals with the data for the principal components without considering the underlying class structure. LDA uses the class information to learn an optimal matrix that maximizes the between-class scatter while minimizing the within-class scatter in feature space. Unfortunately, LDA suffers from the small sample size (SSS) problem because the number of samples in the training set is often much lower than the dimensionality of each sample in practice.

Compared with linear techniques, nonlinear manifold-based approaches can effectively discover the geometrical structure of the underlying manifold. Some representative algorithms include Isomap [11], locally linear embedding [12], and Laplacian Eigenmaps [13]. However, they yield maps that are defined only on the training data and it is unclear how to evaluate the maps on the testing data [14]. For overcoming it, some linear versions of these methods are proposed such as neighborhood preserving projection [15], neighborhood preserving embedding [16], and locality preserving projection (LPP) [17]. LPP can be viewed as a linear version of Laplacian Eigenmaps and more crucially defined everywhere in ambient space rather than just on the training data points. However, LPP is an unsupervised learning method without employing the underlying class information. To utilize the underlying class structure, several methods are proposed to provide discriminant locality preserving projections. The orthogonal neighborhood preserving discriminant analysis [18] effectively combines the advantages of LPP and LDA. Discriminant locality preserving projections (DLPP) [19] improves the classification performance of LPP by making full use of the class structure. Orthogonal discriminant locality preserving projections (ODLPP) [20] improves the recognition performance of DLPP. The above locality preserving methods cannot overcome the SSS problem yet and PCA is often used as the preprocessing method before LPP or DLPP. The null space discriminant locality preserving projections [21] and discriminant locality preserving projections based on maximum margin criterion (DLPPs/MMC) [22] overcome the SSS problem. By extending DLPP to 2-D data, 2-D discriminant locality preserving projections (2-D-DLPPs) [23], [24] and diagonal discriminant locality preserving projections (Dia-DLPPs) [25] are proposed.

However, all classical linear methods and linear versions of manifold-based approaches are sensitive to outliers because

Manuscript received November 13, 2012; revised October 22, 2013 and January 19, 2014; accepted January 27, 2014. Date of publication March 26, 2014; date of current version October 15, 2014. This work was supported in part by the National Science Foundation of China under Grant 61271341 and Grant 60971104 and in part by the Sichuan Basic Science and Technology Foundation under Grant 2013JY0036.

F. Zhong is with the Sichuan Province Key Laboratory of Signal and Information Processing, Southwest Jiaotong University, Chengdu 610031, China, and also with the School of Computer and Information Engineering, Yibin University, Yibin 644000, China (e-mail: fujin-zhong@163.com).

J. Zhang is with the Sichuan Province Key Laboratory of Signal and Information Processing, Southwest Jiaotong University, Chengdu 610031, China (e-mail: jszhang@home.swjtu.edu.cn).

D. Li is with the Psychological Research and Consulting Center, Southwest Jiaotong University, Chengdu, 610031, China (e-mail: ldf125@home.swjtu.edu.cn).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2014.2303798

their distance criterions are based on the L2-norm which magnifies the effect of outliers by the square operation. It is well known that the L1-norm is more robust to the presence of outliers than L2-norm [26]–[30]. Recently, principal component analysis based on L1-norm maximization (PCA-L1) [31] is a fast, robust, and rotational invariant L1-norm-based PCA which learns the local optimal projection axes by maximizing the L1-norm-based variance in the feature space. The optimization algorithm of PCA-L1 is intuitive, simple and easy to implement. According to the report in [31], PCA-L1 obtains lower reconstruction errors than conventional PCA on several databases. Li *et al*. [32] generalizes this algorithm to 2-DPCA to propose L1-norm-based 2-DPCA (2-DPCA-L1) which has lower reconstruction error than PCA-L1, conventional PCA and 2-DPCA. Further, Pang *et al*. [33] proposed a robust tensor analysis with L1-norm (TPCA-L1) [30]. Motivated by PCA-L1, LPP-L1 is proposed to improve the robustness of LPP against outliers and the optimization process of LPP-L1 is similar to that of PCA-L1. Recently, L1-norm-based common spatial pattern (CSP-L1) provides a robust version of CSP [29]. CSP-L1 maximizes the ratio of the L1-dispersion of one class to the other class to obtain the spatial filters by an iterative algorithm which is easy to implement.

In this paper, we focus on improving the robustness of DLPP against outliers and propose a robust version of DLPP based on L1-norm distance (termed as DLPP-L1). By analyzing the objective function of conventional DLPP, we get the transformed version of its objective function, which is formed using L2-norm. Referring to the L2-norm-based objective function of conventional DLPP, the proposed method has its own objective function based on L1-norm, that is, the ratio of the L1-norm-based locality preserving between-class dispersion and the L1-norm-based locality preserving within-class dispersion. DLPP-L1 aims to obtain a set of projection axes by maximizing the L1-norm-based objective function. But, it is very difficult to directly find the global optimal solution of the new L1-norm-based objective function. To solve this problem, we simplify the new objective function to a problem of finding the single local optimal projection vector which can be solved by an iteration algorithm and multiple local optimal projection vectors can be gained by a greedy search method. The solution of DLPP-L1 is theoretically proven to be feasible. Moreover, it is robust to outliers while overcoming the SSS problem of conventional DLPP (termed as DLPP-L2).

The remainder of this paper is organized as follows. In Section II, the problem is formulated. The proposed DLPP-L1 is presented and the feasibility of the solving scheme is theoretically proven in Section III. The proposed method is applied to several pattern recognition problems and the experimental results are reported in Section IV. Lastly, Section V concludes this paper.

## II. Problem Formulation

Let $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_N] \in \mathbf{R}^{m\times N}$ be the given training samples, where $N$ is the number of samples and $\mathbf{x}_i$ denotes an $m$-dimensional column vector. If the given samples contain $\varsigma$ classes, the $c$th class has $N_c$ samples $\left(\sum_{c=1}^{\varsigma} N_c = N\right)$. So, each sample $\mathbf{x}_i$ belongs to one of the $\varsigma$ classes $\{\mathbf{X}_1, \mathbf{X}_2, \ldots, \mathbf{X}_\varsigma\}$, where $\mathbf{X}_c = \{\mathbf{x}_1^c, \mathbf{x}_2^c, \ldots, \mathbf{x}_{N_c}^c\}$, $(c = 1, 2, \ldots, \varsigma)$. DLPP-L2 aims to find an optimal projection matrix $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_n] \in \mathbf{R}^{m\times n}$ $(n < m)$ whose columns $\{\mathbf{a}_k\}$ $(k = 1, \ldots, n)$ constitute the basis vectors of the $n$-dimension subspace. Projecting the sample $\mathbf{x}_i$ onto $\mathbf{A}$ yields an $n$-dimension vector $\mathbf{y}_i$, i.e., $\mathbf{y}_i = \mathbf{A}^T\mathbf{x}_i$ where $\mathbf{y}_i$ is called the feature of $\mathbf{x}_i$ in the $n$-dimension subspace. The optimal basis vector $\mathbf{a} \in \mathbf{R}^{m\times 1}$ can be gained by maximizing the following objective function as [19]:

$$\mathbf{a} = \arg\max_{\mathbf{a}} J(\mathbf{a}) = \arg\max_{\mathbf{a}} \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\left(\mathbf{a}^T\mathbf{m}_i - \mathbf{a}^T\mathbf{m}_j\right)^2}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left(\mathbf{a}^T\mathbf{x}_i^c - \mathbf{a}^T\mathbf{x}_j^c\right)^2} \quad (1)$$

where the superscript $T$ is the transposition operation, $\mathbf{m}_i$ and $\mathbf{m}_j$ are the mean vectors of the $i$th class and $j$th class samples, respectively, that is, $\mathbf{m}_i = 1/N_i \sum_{k=1}^{N_i} \mathbf{x}_k^i$ and $\mathbf{m}_j = 1/N_j \sum_{k=1}^{N_j} \mathbf{x}_k^j$. $B_{ij}$ is the weight between the mean vectors of the $i$th class and $j$th class samples, and its components can be defined as: $B_{ij} = \exp\left(-\|\mathbf{m}_i - \mathbf{m}_j\|^2/s\right)$, where $s$ is an empirically determined parameter. $W_{ij}^c$ is the weight between any two samples $\mathbf{x}_i^c$ and $\mathbf{x}_j^c$ in the $c$th class, and its components can be defined as: $W_{ij}^c = \exp\left(-\|\mathbf{x}_i^c - \mathbf{x}_j^c\|^2/t\right)$ where $t$ is a parameter that can be determined empirically.

The global solution of (1) is provided by the generalized eigenvalue problem, which is also the solution of the following problem according to convex optimization theory [34]:

$$\mathbf{A}_{\text{opt}} = \arg\max_{\mathbf{A}} \times \frac{\text{trace}\left(\sum_{i,j=1}^{\varsigma} B_{ij}\left(\mathbf{A}^T\mathbf{m}_i - \mathbf{A}^T\mathbf{m}_j\right)\left(\mathbf{A}^T\mathbf{m}_i - \mathbf{A}^T\mathbf{m}_j\right)^T\right)}{\text{trace}\left(\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left(\mathbf{A}^T\mathbf{x}_i^c - \mathbf{A}^T\mathbf{x}_j^c\right)\left(\mathbf{A}^T\mathbf{x}_i^c - \mathbf{A}^T\mathbf{x}_j^c\right)^T\right)} \quad (2)$$

where $\text{trace}(\cdot)$ is the trace of a matrix. By simply algebra transforming, (2) can be rewritten as

$$\begin{aligned}\mathbf{A}_{\text{opt}} &= \arg\max_{\mathbf{A}} \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\left\|\mathbf{A}^T\mathbf{m}_i - \mathbf{A}^T\mathbf{m}_j\right\|_2^2}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left\|\mathbf{A}^T\mathbf{x}_i^c - \mathbf{A}^T\mathbf{x}_j^c\right\|_2^2} \\ &= \arg\max_{\mathbf{A}} \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\left\|\mathbf{A}^T(\mathbf{m}_i - \mathbf{m}_j)\right\|_2^2}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left\|\mathbf{A}^T(\mathbf{x}_i^c - \mathbf{x}_j^c)\right\|_2^2}\end{aligned} \quad (3)$$

where $\|\cdot\|_2$ is the L2-norm.

According to (3), we find that the objective function of DLPP-L2 is obviously based on the L2-norm distance criterion. However, it is well known that L2-norm is sensitive

to outliers because the square operator magnifies the function of the outliers. From a statistical point of view, the L1-norm-based methods are more robust to outliers than the L2-norm-based methods [31]. Motivated by the above idea, we present DLPP-L1 which obtains the optimal projection matrix by solving the optimization problem as the following:

$$\begin{aligned}\mathbf{A}_{\text{opt}} &= \arg\max_{\mathbf{A}} F(\mathbf{A}) = \arg\max_{\mathbf{A}} \frac{\sum_{i,j=1}^{\varsigma} B_{ij} \left\|\mathbf{A}^T(\mathbf{m}_i - \mathbf{m}_j)\right\|_1}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \left\|\mathbf{A}^T(\mathbf{x}_i^c - \mathbf{x}_j^c)\right\|_1} \\ &= \arg\max_{\mathbf{A}} \frac{\sum_{i,j=1}^{\varsigma} B_{ij} \sum_{k=1}^{n} \left|\mathbf{a}_k^T(\mathbf{m}_i - \mathbf{m}_j)\right|}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \sum_{k=1}^{n} \left|\mathbf{a}_k^T(\mathbf{x}_i^c - \mathbf{x}_j^c)\right|} \\ &\quad \text{subject to} \quad \mathbf{A}^T\mathbf{A} = \mathbf{I}_n \end{aligned} \tag{4}$$

where $\|\cdot\|_1$ is L1-norm and $\mathbf{A}^T\mathbf{A} = \mathbf{I}_n$ is to ensure the orthonormality of the projection matrix. However, it is very difficult to obtain the global optimal solution of (4) for $n > 1$. To overcome this problem, we can simplify (4) into a series of $n = 1$ optimization problems by a greedy search method. If $n$ is set to 1, (4) is transformed to the following optimization problem:

$$\mathbf{a}* = \arg\max_{\mathbf{w}} F(\mathbf{a}) \quad \text{subject to } \mathbf{a}^T\mathbf{a} = 1 \tag{5}$$

where

$$F(\mathbf{a}) = \frac{\sum_{i,j=1}^{\varsigma} B_{ij} \left|\mathbf{a}^T(\mathbf{m}_i - \mathbf{m}_j)\right|}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \left|\mathbf{a}^T(\mathbf{x}_i^c - \mathbf{x}_j^c)\right|}. \tag{6}$$

The successive greedy solutions of (5) may be different from the global optimal solution of (4), but it is expected to provide a good approximation of solving (4) [31]. However, the absolute value operation is not differentiable, which makes it difficult to directly find the global optimal solution of (5). In Section III-A, an iterative algorithm to find one local optimal solution of (5) is presented. A greedy search algorithm for $n > 1$ is presented in Section III-B.

## III. Solution of DLPP-L1

### A. Algorithm of DLPP-L1 for n = 1

Suppose $\mathbf{a}* \in \mathbf{R}^{m\times 1}$ is the local optimal solution of (5). Because $\mathbf{a}*$ is the convergence result of iterations, we can use $\mathbf{a}(t)$ to denote the result of the $t^{\text{th}}$ iteration and assume that $\mathbf{a}(t) \neq \mathbf{0}$. The solving algorithm of DLPP-L1 for $n = 1$ can be described as the following.

*Step 1 (Initialization):* Set $t = 0$, initialize $\mathbf{a}(0) = \arg\max_{\mathbf{m}_i} \|\mathbf{m}_i\|_2$ and rescale it to unit length as $\mathbf{a}(0) = \mathbf{a}(0)/\|\mathbf{a}(0)\|_2$.

*Step 2:* Defining of two polarity functions

$$p_{ij}(t) = \begin{cases} 1, & \text{if } \mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j) \geq 0 \\ -1, & \text{if } \mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j) < 0 \end{cases} (i, j = 1, \ldots, \varsigma) \tag{7}$$

$$q_{ij}^c(t) = \begin{cases} 1, & \text{if } \mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right) \geq 0 \\ -1, & \text{if } \mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right) < 0 \end{cases} \begin{pmatrix} i, j = 1, \ldots, N_C \\ \text{and } c = 1, \ldots, \varsigma \end{pmatrix}. \tag{8}$$

Substituting (7) and (8) into (6), we have that

$$F(\mathbf{a}(t)) = \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\, p_{ij}(t)\mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j)}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c q_{ij}^c(t)\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)}. \tag{9}$$

*Step 3:* Updating $\mathbf{a}(t+1)$: Let

$$\begin{aligned}\mathbf{d}(t) = &\frac{\sum_{i,j=1}^{\varsigma} B_{ij}\, p_{ij}(t)(\mathbf{m}_i - \mathbf{m}_j)}{\sum_{i,j=1}^{\varsigma} B_{ij} \left|\mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j)\right|} \\ &- \frac{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c q_{ij}^c(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \left|\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right|}. \end{aligned} \tag{10}$$

Update $\mathbf{a}(t+1)$ by

$$\mathbf{a}(t+1) = \mathbf{a}(t) + \beta\mathbf{d}(t). \tag{11}$$

Rescale $\mathbf{a}(t+1)$ to unit length and set $t = t + 1$. Here, $\beta$ is the update rate taking a small positive value. In the real experiments, we can try a series of different small positive values of $\beta$, and select the optimal updating rate which leads to the largest value of the objective function $F(\mathbf{a}(\cdot))$. Therefore, we can avoid the bad case that $\mathbf{a}(t+1)$ overshoots the optimal point that results in a smaller value of $F(\mathbf{a}(\cdot))$.

*Step 4:* Convergence check: If $F(\mathbf{a}(t+1))$ cannot increase significantly (measured by the increase speed), the iteration is ended and the local optimal $\mathbf{a}^* = \mathbf{a}(t)$ can be output. Otherwise, go to Step 2.

*Theorem 1:* With the above iterative procedure, $F(\mathbf{a}(t))$ is a nondecreasing function of $t$.

*Proof:* At the $t$th iteration, we have that

$$F(\mathbf{a}(t)) = \frac{\sum_{i,j=1}^{\varsigma} B_{ij} \left|\mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j)\right|}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \left|\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right|}. \tag{12}$$

According to (7), the numerator of (12) can be rewritten as

$$\begin{aligned}\sum_{i,j=1}^{\varsigma} B_{ij} \left|\mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j)\right| &= \mathbf{a}^T(t) \sum_{i,j=1}^{\varsigma} B_{ij}\, p_{ij}(t)(\mathbf{m}_i - \mathbf{m}_j) \\ &= \mathbf{a}^T(t)\mathbf{u}(t) \end{aligned} \tag{13}$$

where

$$\mathbf{u}(t) = \sum_{i,j=1}^{\varsigma} B_{ij}\, p_{ij}(t)(\mathbf{m}_i - \mathbf{m}_j). \tag{14}$$

The denominator of (12) can be rewritten as

$$\begin{aligned}
&\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c \left|\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right| \\
&= \frac{1}{2}\mathbf{a}^T(t)\left(\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} \frac{W_{ij}^c\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)^T W_{ij}^c}{\left|W_{ij}^c \mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right|}\right)\mathbf{a}(t) \\
&\quad + \frac{1}{2}\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c}\left|W_{ij}^c \mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right| \\
&= \frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1
\end{aligned} \tag{15}$$

where

$$\mathbf{V}(t) = \sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} \frac{W_{ij}^c\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)^T W_{ij}^c}{\left|z_{ij}^c(t)\right|} \tag{16}$$

and $\mathbf{z}(t)$ is the vector having the entries $z_{ij}^c(t) = W_{ij}^c \mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)$ $(i, j = 1, \ldots, N_C$ and $c = 1, \ldots, \varsigma)$. Substituting (13) and (15) into (12), we get that

$$F(\mathbf{a}(t)) = \frac{\mathbf{a}^T(t)\mathbf{u}(t)}{\frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1}. \tag{17}$$

Because it is intractable to directly gain the derivative of $F(\cdot)$, we introduce a surrogate function as the following:

$$L(\psi) = \frac{\psi^T \mathbf{u}(t)}{\frac{1}{2}\psi^T\mathbf{V}(t)\psi + \frac{1}{2}\|\mathbf{z}(t)\|_1}. \tag{18}$$

We note that $\mathbf{a}(t)$ is a constant vector at the $t$th iteration, so $\mathbf{u}(t)$, $\mathbf{V}(t)$ and $\mathbf{z}(t)$ are fixed values, only $\psi$ is the variable in the function $L(\psi)$. Therefore, the gradient of $L(\psi)$ with respect to $\psi$ can be calculated as the following:

$$\begin{aligned}
g(\psi) &= \frac{\partial L(\psi)}{\partial \psi} \\
&= \frac{\left(\frac{1}{2}\psi^T\mathbf{V}(t)\psi + \frac{1}{2}\|\mathbf{z}(t)\|_1\right)\mathbf{u}(t) - \left(\psi^T\mathbf{u}(t)\right)\mathbf{V}(t)\psi}{\left(\frac{1}{2}\psi^T\mathbf{V}(t)\psi + \frac{1}{2}\|\mathbf{z}(t)\|_1\right)^2}.
\end{aligned} \tag{19}$$

Substituting $\mathbf{a}(t)$ into (19), we can get the gradient value at the point $\mathbf{a}(t)$

$$\begin{aligned}
g(\mathbf{a}(t)) = &\frac{\left(\frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1\right)\mathbf{u}(t)}{\left(\frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1\right)^2} \\
&- \frac{\left(\mathbf{a}^T(t)\mathbf{u}(t)\right)\mathbf{V}(t)\mathbf{a}(t)}{\left(\frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1\right)^2}.
\end{aligned} \tag{20}$$

Substituting $\mathbf{u}(t)$, $\mathbf{V}(t)$ and $\mathbf{z}(t)$ into (20), which can be rewritten as

$$\begin{aligned}
&g(\mathbf{w}(t)) \\
&= \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\, p_{ij}(t)(\mathbf{m}_i - \mathbf{m}_j)}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left|\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right|} \\
&\quad - \frac{\sum_{i,j=1}^{\varsigma} B_{ij}\left|\mathbf{a}^T(t)(\mathbf{m}_i - \mathbf{m}_j)\right|\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c q_{ij}^c(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)}{\left(\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left|\mathbf{a}^T(t)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right|\right)^2} \\
&\quad \propto \mathbf{d}(t).
\end{aligned} \tag{21}$$

According to (21), $\mathbf{d}(t)$ is exactly the vector that points to the ascending direction of $g(\psi)$ at the point $\mathbf{a}(t)$. So, with the vector $\mathbf{a}(t+1)$ defined by (11), replacing $\psi$ with $\mathbf{a}(t)$ and $\mathbf{a}(t+1)$, we have that $L(\mathbf{a}(t+1)) \geq L(\mathbf{a}(t))$, that is

$$\begin{aligned}
&\frac{\mathbf{a}^T(t+1)\mathbf{u}(t)}{\frac{1}{2}\mathbf{a}^T(t+1)\mathbf{V}(t)\mathbf{a}(t+1) + \frac{1}{2}\|\mathbf{z}(t)\|_1} \\
&\qquad \geq \frac{\mathbf{a}^T(t)\mathbf{u}(t)}{\frac{1}{2}\mathbf{a}^T(t)\mathbf{V}(t)\mathbf{a}(t) + \frac{1}{2}\|\mathbf{z}(t)\|_1} = F(\mathbf{a}(t)).
\end{aligned} \tag{22}$$

It is obvious that both the right side of the above inequality and denominator of the left side are non-negative, so the numerator of the left side must be non-negative, that is, $\mathbf{a}^T(t+1)\mathbf{u}(t) \geq 0$. The purpose of (22) is to bridge the justification of $F(a(t+1)) \geq F(a(t))$. Therefore, the next steps just need prove the following inequality:

$$F(\mathbf{a}(t+1)) \geq \frac{\mathbf{a}^T(t+1)\mathbf{u}(t)}{\frac{1}{2}\mathbf{a}^T(t+1)\mathbf{V}(t)\mathbf{a}(t+1) + \frac{1}{2}\|\mathbf{z}(t)\|_1}. \tag{23}$$

First, for proving the inequality of (23), we introduce the following lemma.

*Lemma 1:* For any vector $\mathbf{e} = (e_1, \ldots, e_N)^T \in \mathbf{R}^M$, the following variational equality holds [35]:

$$\|\mathbf{e}\|_1 = \sum_{k=1}^{M}|e_k| = \min_{\xi\in\mathbf{R}_+^M}\frac{1}{2}\sum_{k=1}^{M}\frac{e_k^2}{\xi_k} + \frac{1}{2}\|\xi\|_1 \tag{24}$$

and the minimum is uniquely reached at $\xi_k = |e_k|$ for $k = 1, \ldots, N$, where $\xi = (\xi_1, \ldots, \xi_N)^T$.

Let $e_k = z_{ij}^c(t+1)$ $\left(k = 1, \ldots, M, M = \sum_{c=1}^{\varsigma} N_c^2\right.$, $c = 1, \ldots, \varsigma$, and $i, j = 1, \ldots, N_C)$, (24) can be rewritten as the following:

$$\begin{aligned}
\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} W_{ij}^c\left|\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right| = &\min_{\xi\in\mathbf{R}_+^M}\frac{1}{2}\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c} \\
&\frac{\left(W_{ij}^c\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c - \mathbf{x}_j^c\right)\right)^2}{\xi_k} + \frac{1}{2}\|\xi\|_1.
\end{aligned} \tag{25}$$

Second, because $p_{ij}(t+1)$ is the polarity of $\mathbf{a}^T(t+1)(\mathbf{m}_i - \mathbf{m}_j)$, the quantity $p_{ij}(t+1)\mathbf{a}^T(t+1)$ $(\mathbf{m}_i - \mathbf{m}_j)$ is always non-negative for any $(i, j = 1, \ldots, \varsigma)$. But, it is possible that $p_{ij}(t)\mathbf{a}^T(t+1)(\mathbf{m}_i - \mathbf{m}_j)$ is not always non-negative for all $(i, j = 1, \ldots, \varsigma)$. So, for the

numerator of the right side of (23), the following inequality holds:

$$\begin{aligned}\sum_{i,j=1}^{\varsigma} B_{ij}\left|\mathbf{a}^T(t+1)(\mathbf{m}_i-\mathbf{m}_j)\right| &= \mathbf{a}^T(t+1)\sum_{i,j=1}^{\varsigma} B_{ij}p_{ij}(t+1)(\mathbf{m}_i-\mathbf{m}_j)\\ &\geq \mathbf{a}^T(t+1)\sum_{i,j=1}^{\varsigma} B_{ij}p_{ij}(t)(\mathbf{m}_i-\mathbf{m}_j)\\ &= \mathbf{a}^T(t+1)\mathbf{u}(t)\geq 0. \end{aligned} \tag{26}$$

On the other hand, for the denominator of (23), we can get the following:

$$\begin{aligned}&\frac{1}{2}\mathbf{a}^T(t+1)\mathbf{V}(t)\mathbf{a}(t+1)+\frac{1}{2}\left\|\mathbf{z}(t)\right\|_1\\ &=\frac{1}{2}\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c}\frac{\left(W_{ij}^c\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c-\mathbf{x}_j^c\right)\right)^2}{\left|z_{ij}^c(t)\right|}+\frac{1}{2}\left\|\mathbf{z}(t)\right\|_1\\ &\geq \min_{\xi\in\mathbf{R}_+^M}\frac{1}{2}\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c}\frac{\left(W_{ij}^c\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c-\mathbf{x}_j^c\right)\right)^2}{\xi_k}+\frac{1}{2}\left\|\xi\right\|_1. \end{aligned} \tag{27}$$

According to (25) and (27), the following holds:

$$\begin{aligned}&\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c}W_{ij}^c\left|\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c-\mathbf{x}_j^c\right)\right|\\ &\qquad\leq\frac{1}{2}\mathbf{a}^T(t+1)\mathbf{V}(t)\mathbf{a}(t+1)+\frac{1}{2}\left\|\mathbf{z}(t)\right\|_1. \end{aligned} \tag{28}$$

Combining (26) and (28), we can get the following:

$$\begin{aligned}&\frac{\sum_{i,j=1}^{\varsigma}B_{ij}\left|\mathbf{a}^T(t+1)(\mathbf{m}_i-\mathbf{m}_j)\right|}{\sum_{c=1}^{\varsigma}\sum_{i,j=1}^{N_c}W_{ij}^c\left|\mathbf{a}^T(t+1)\left(\mathbf{x}_i^c-\mathbf{x}_j^c\right)\right|}\\ &\qquad\geq\frac{\mathbf{a}^T(t+1)\mathbf{u}(t)}{\frac{1}{2}\mathbf{a}^T(t+1)\mathbf{V}(t)\mathbf{a}(t+1)+\frac{1}{2}\left\|\mathbf{z}(t)\right\|_1}. \end{aligned} \tag{29}$$

Finally, it is found that the left side of (29) is exactly equal to $F(\mathbf{a}(t+1))$. Therefore, we can conclude that

$$F(\mathbf{a}(t+1))\geq F(\mathbf{a}(t)). \tag{30}$$

In addition, according to (12), we get that $F(\lambda\mathbf{a}(t))=F(\mathbf{a}(t))$ where $\lambda$ is a scale constant. Therefore, the projection vector $\mathbf{a}(t)$ can be normalized without effect for the local optimal solution. Essentially, only the direction of the projection vector is of interest.

We have proved that $F(\mathbf{a}(t))$ is nondecreasing with respect to iteration $t$. It seldom happens that $F(\mathbf{a}(t))$ remains unchangeable during the prophase of iteration procedure, which ensures that $\mathbf{a}(t)$ moves toward a local optimal solution of (5). However, it is very difficult to obtain the global optimal solution. In the future, further work is aimed at this problem.

### *B. Extension to n > 1*

In the previous section, we have presented the algorithm that can extract one local optimal projection vector which maximizes the objective function (5). We can easily extend it to extract multiple projection vectors when $n>1$ by a greedy search method as the following.

*Step 1 (Initialization):* set

$$\begin{aligned}&\mathbf{a}_0=\mathbf{0};\ (\mathbf{m}_i)_0=\mathbf{m}_i\ (i=1,\ldots,\varsigma);\\ &\left(\mathbf{x}_i^c\right)_0=\mathbf{x}_i^c\ (i=1,\ldots,N_C\ \text{and}\ c=1,\ldots,\varsigma).\end{aligned}$$

*Step 2 (Extracting n Projection Vectors):* for $k=1$ to $n$

$$\begin{aligned}(\mathbf{m}_i)_k&=(\mathbf{m}_i)_{k-1}-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T(\mathbf{m}_i)_{k-1}\ (i=1,\ldots,\varsigma);\\ \left(\mathbf{x}_i^c\right)_k&=\left(\mathbf{x}_i^c\right)_{k-1}-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\left(\mathbf{x}_i^c\right)_{k-1}\\ &\quad\times\left(i=1,\ldots,N_C\text{and}\ c=1,\ldots,\varsigma\right).\end{aligned} \tag{31}$$

To extract $\mathbf{a}_k$, apply the iteration algorithm of the above subsection using the samples $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$. ■

These projection vectors extracted by the above procedure are guaranteed to be orthonormal. First, $\mathbf{a}_k$ is a unit vector due to the normalization during the course of iteration algorithm. Next, we can justify that $\mathbf{a}_k$ is orthogonal to $\mathbf{a}_{k-1}$ for all $k$ $(k=2,\ldots,n)$ as the following.

1) According to the initialized value of $\mathbf{a}(0)$, (10) and (11), $\mathbf{a}_k$ is a linear combination of the samples $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$, so $\mathbf{a}_k$ is in the space spanned by $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$.
2) By multiplying $\mathbf{a}_{k-1}^T$ to both sides of $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$ in (31), we have that
$$\begin{aligned}\mathbf{a}_{k-1}^T(\mathbf{m}_i)_k&=\mathbf{a}_{k-1}^T\left(\mathbf{m}_i\right)_{k-1}\\ &\quad-\mathbf{a}_{k-1}^T\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T(\mathbf{m}_i)_{k-1}=0\\ \mathbf{a}_{k-1}^T(\mathbf{x}_i^c)_k&=\mathbf{a}_{k-1}^T(\mathbf{x}_i^c)_{k-1}-\mathbf{a}_{k-1}^T\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\left(\mathbf{x}_i^c\right)=0.\end{aligned}$$
3) From 2), $\mathbf{a}_{k-1}$ is orthogonal to $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$, which shows that $\mathbf{a}_k$ is orthogonal to $\mathbf{a}_{k-1}$ according to 1).

Then, we can justify that $\mathbf{a}_k$ is orthogonal to $\mathbf{a}_{k-2}$ for all $k$ $(k=3,\ldots,n)$ as the following.

1) By recursively calling (31), we have that
$$\begin{aligned}(\mathbf{m}_i)_k&=\left(\mathbf{I}_m-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\right)\left(\mathbf{I}_m-\mathbf{a}_{k-2}\mathbf{a}_{k-2}^T\right)(\mathbf{m}_i)_{k-2}\\ (\mathbf{x}_i^c)_k&=\left(\mathbf{I}_m-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\right)\left(\mathbf{I}_m-\mathbf{a}_{k-2}\mathbf{a}_{k-2}^T\right)\left(\mathbf{x}_i^c\right)_{k-2}.\end{aligned}$$
2) By multiplying $\mathbf{w}_{k-2}^T$ to both sides of $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$ in above equations, respectively, we get
$$\begin{aligned}\mathbf{a}_{k-2}^T(\mathbf{m}_i)_k&=\mathbf{a}_{k-2}^T\left(\mathbf{I}_m-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\right)\left(\mathbf{I}_m-\mathbf{a}_{k-2}\mathbf{a}_{k-2}^T\right)\\ &\qquad(\mathbf{m}_i)_{k-2}=0;\\ \mathbf{a}_{k-2}^T(\mathbf{x}_i^c)_k&=\mathbf{a}_{k-2}^T\left(\mathbf{I}_m-\mathbf{a}_{k-1}\mathbf{a}_{k-1}^T\right)\left(\mathbf{I}_m-\mathbf{a}_{k-2}\mathbf{a}_{k-2}^T\right)\\ &\qquad(\mathbf{x}_i^c)_{k-2}=0.\end{aligned}$$
3) From 2), $\mathbf{a}_{k-2}$ is orthogonal to $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$, which shows that $\mathbf{a}_k$ is orthogonal to $\mathbf{a}_{k-2}$ because $\mathbf{a}_k$ is a linear combination of $(\mathbf{m}_i)_k$ and $\left(\mathbf{x}_i^c\right)_k$.

Finally, with the induction and summarization of the above two proof procedures, we can conclude that these $n$ projection vectors are orthonormal, that is

$$\mathbf{a}_s^T\mathbf{a}_t=\begin{cases}1, & \text{if } s=t\\ 0, & \text{if } s\neq t.\end{cases}$$

The successive greedy algorithm may not derive the optimal solution of (4), but it is expected to obtain a series of good projection vectors that maximize the L1-norm-based objective function. But the following experimental results show that the proposed DLPP-L1 is more robust to outliers than DLPP-L2. On the other hand, in DLPP-L2, one sometimes has to be confronted with the SSS problem in practice. However, from the above description, DLPP-L1 overcomes the SSS problem completely.

## IV. Experimental Result

We perform several experiments to show the robustness of the proposed DLPP-L1 against outliers. In Section IV-A, we have taken two toy experiments on artificial datasets for comparing DLPP-L1 with DLPP-L2. In Sections IV-B and IV-C, the experimental results on Binary Alphadigits database [36] and FERET face database [37] are, respectively, presented. Lastly, we give the experimental results on PolyU palmprint database [38], [39]. In all experiments, the updating parameter $\beta$ of DLPP-L1 is set to 0.01. In the last three sections, an Euclidean distance-based nearest neighbor classifier [40] is used for object classification. On the other hand, PCA is used as a preprocessing method for overcoming the SSS problem of LDA, DLPP-L2, and ODLPP. In addition, all experiments are executed on a computer system of Intel T2350 1.86 GHz and 1 GB RAM with MATLAB 7.9.

### A. Toy Experiments

We conduct two experiments on artificial datasets. In the first experiment, we create two data classes which are Gaussian classes with covariance matrices being [0.05 0; 0 2] and means being [−2 0] and [2 0], respectively. Each class consists of 20 2-D data samples as depicted in Fig. 1, where the two classes of data points are specified by black "o" and "x." Without any outlier, for classification, the real optimal discriminant projection vector should be $\mathbf{w} = [1, 0]^T$ ($\theta = 0°$). We extract the projection vectors using DLPP-L2 and DLPP-L1 for the above samples as the training dataset, respectively. Fig. 1 shows these two projection vectors, that is, $\mathbf{w}_{\mathrm{DLPP-L2}} = [0.9999, -0.0133]^T$ ($\theta_{L2} = -0.76^{\circ}$) and $\mathbf{w}_{\mathrm{DLPP-L1}} = [0.9988, -0.0486]$ ($\theta_{L1} = 2.79^{\circ}$). We find that $\mathbf{w}_{\mathrm{DLPP-L2}}$ and $\mathbf{w}_{\mathrm{DLPP-L1}}$ are close to the optimal projection vector. For comparing the robustness of DLPP-L2 and DLPP-L1 to outlier, we add an outlier, that is, [8, 8] specified by red "o" in Fig. 1. Then we construct another training dataset based on the above dataset in which the first data point of class "o" is replaced with the outlier [8, 8]. Then, we extract the projection vectors using DLPP-L2 and DLPP-L1 on the new training dataset including the outlier, respectively. Fig. 1 shows these two projection vectors, that is, $\mathbf{w}^{\mathrm{outlier}}_{\mathrm{DLPP-L2}} = [0.8242, -0.5663]^T$ ($\theta_{L2} = -34.49^{\circ}$) and $\mathbf{w}^{\mathrm{outlier}}_{\mathrm{DLPP-L1}} = [0.9897, -0.1430]^T$ ($\theta_{L1} = -8.22^{\circ}$). Obviously, $\mathbf{w}^{\mathrm{outlier}}_{\mathrm{DLPP-L2}}$ is more severely deviated from the original optimal projection vector than $\mathbf{w}^{\mathrm{outlier}}_{\mathrm{DLPP-L1}}$ for the outlier's interference, which indicates that DLPP-L1 is more robust to outliers than DLPP-L2.

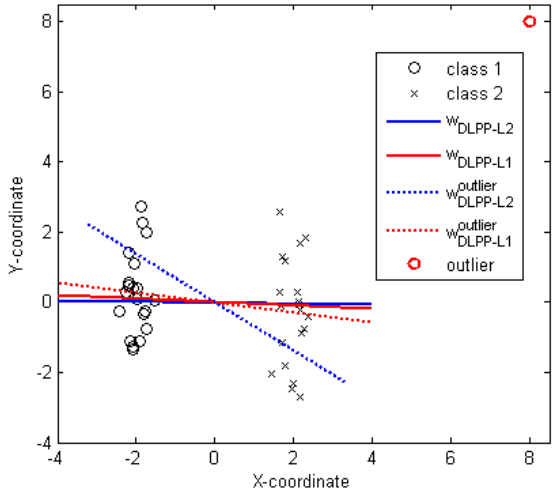


Fig. 1. Projection vectors extracted by DLPP-L2 and DLPP-L1 on an artificial dataset.

In the second experiment, we create three Gaussian classes and each class contains 40 2-D samples shown in Fig. 2(a). The simulated three data classes follow Gaussian distribution with covariance matrices being [0.5 0; 0 0.5], and means being [−2 0], [2 0], and [0 3.5], respectively. Similarly, we introduce an additional outlier, that is, [1000, 1000]. This experiment consists of two tests. In the first test, 120 Gaussian samples of three classes are used as the training dataset. After obtaining two projection vectors by DLPP-L2 and DLPP-L1, respectively, we map these three classes of inlying data to another 2-D space. Fig. 2(b) shows the result of DLPP-L2 and Fig. 2(c) the result of DLPP-L1. It is found that both DLPP-L2 and DLPP-L1 perform almost perfectly in revealing the structure of the original Gaussian dataset. In the second test, the first sample of the training dataset in the first test is replaced with the outlier. The renewed training dataset is used for the training dataset of the second test. Similarly, we map the inlying data to another 2-D space based on two projection vectors extracted using DLPP-L2 and DLPP-L1, respectively. Fig. 2(d) shows the result of DLPP-L2 and it is easy to see that DLPP-L2 severely distorts the structure of the inlying data for the interference of the outlier. However, Fig. 2(e) shows the result of DLPP-L1, which indicates that our proposed DLPP-L1 is strongly robust to outlier when the data samples follow Gaussian distribution. In practice, there seldom exists such normative dataset. Therefore, the following experiments will be performed on real databases.

### B. Experiments on Binary Alphadigits

Binary Alphadigits database contains binary $20 \times 16$ digits of "0" through "9" and capital "A" through "Z." There are 39 examples of each class. All samples are divided into two groups: digits and capital letters. Fig. 3(a) shows the 390 samples of digits and Fig. 3(b) shows the 1014 samples of capital letters. We can find that most samples are relatively standardized, but a few samples are illegible and hardly distinguished visually. To some extent, those relatively

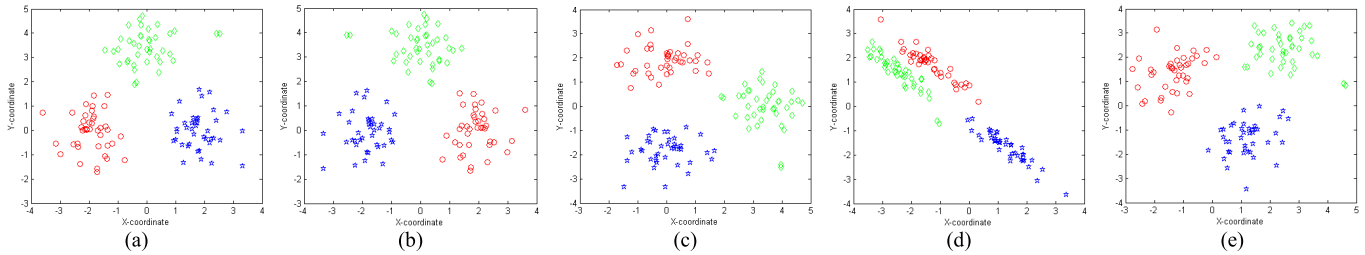


Fig. 2. Original data of three Gaussian classes and projected data obtained by applying DLPP-L2 or DLPP-L1 on these data. (a) Original data. (b) Projected data of DLPP-L2 without outlier. (c) Projected data of DLPP-L1 without outlier. (d) Projected data of DLPP-L2 with outlier. (e) Projected data of DLPP-L1 with outlier.

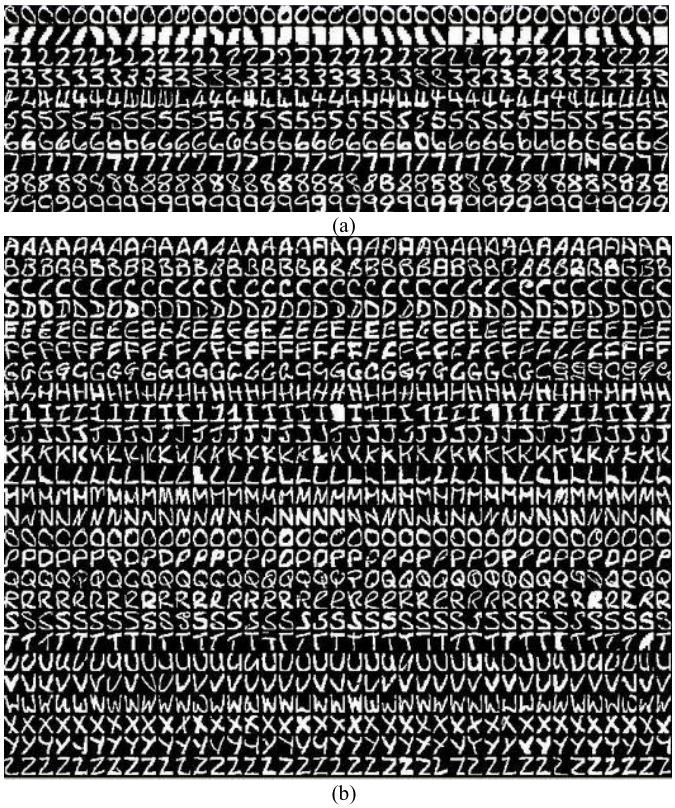


Fig. 3. Binary Alphadigits samples. (a) Digits. (b) Capital letters.

normative samples can be viewed as inlying data and those illegible samples can be viewed as outlying data. We conduct objection recognition tests on two groups, respectively. The $k$ ($k = 3, 6, 9, 12, 15, 18$) samples of each class are randomly selected as the training set and the remaining 39-$k$ images of every class are used as the testing set. The tests are repeated 10 times with each special training sample number. When PCA uses the maximum $N-C$ principal components where $N$ is the number of all training samples and $C$ is the number of classes, we record the optimal correct recognition rates of LDA, DLPP-L2, ODLPP, and DLPP-L1. Then, the average optimal recognition rates and the standard deviations are calculated and shown in Table I. We find that DLPP-L1 outperforms other three methods completely. On the other hand, it is normal that the recognition accuracy increases with the increasing of the training sample number. However, the recognition accuracy

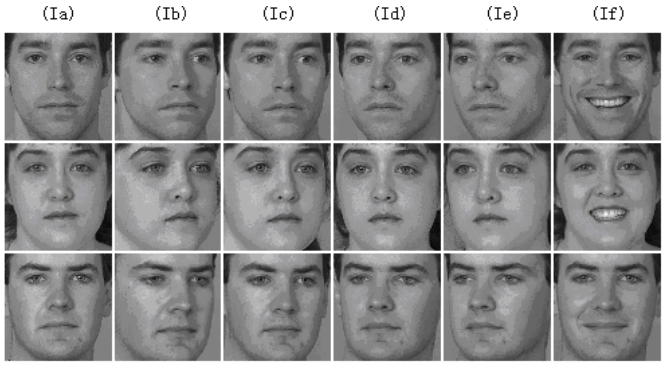


Fig. 4. Images of three subjects in partial FERET database.

of DLPP-L2 decreases on the digits group when the training number overruns 9 and grows worse with the increasing of the training number on the capital letters group from the very beginning. On the contrary, the recognition accuracy of DLPP-L1 grows better with the increasing of the training number on both groups. The key reason should be that some illegible samples are gradually included in the training set according to the probability while the training number is increasing and DLPP-L1 is more robust to those outlying data than DLPP-L2. Because it is well known that PCA can reduce the noise while using appropriate feature dimensionality, we also record the experimental results in Table I when PCA uses 29 principal components. Obviously, the performances of three traditional methods are much improved because the effect of the outliers is significantly reduced by PCA. All experimental results are comprehensively considered, we conclude that DLPP-L1 is more robust to outliers than other three methods. In this paper, we focus on comparing their performances when outliers are present, so the PCA step simply uses the maximum $N$–$C$ principal components in the following experiments.

### *C. Experiments on FERET Database*

In our experiments, we select a subset which includes 1200 images of 200 different subjects from the FERET face database, that is, each subject contains six images, which are marked with two-character strings: “Ia,” “Ib,” “Ic,” “Id,” “Ie,” “If” to denote different face variation, respectively. Fig. 4 shows the samples of three subjects in the FERET subset. The images called “Ia” are neutral expression and frontal view, the

TABLE I

OPTIMAL AVERAGE RECOGNITION RATES (%) AND STANDARD DEVIATION ON BINARY ALPHADIGITS DATABASE

| Dataset | PCA Step | Methods | 3 Training Samples | 6 Training Samples | 9 Training Samples | 12 Training Samples | 15 Training Samples | 18 Training Samples |
|---|---|---|---|---|---|---|---|---|
| Digits | Using *N*-*C* components | LDA | 67.7±4.9 | 73.5±2.9 | 77.2±1.9 | 75.3±3.3 | 72.6±2.7 | 70.7±3.5 |
| | | DLPP-L2 | 68.3±4.9 | 74.6±2.4 | 78.3±1.9 | 77.4±2.5 | 73.6±2.3 | 71.0±3.5 |
| | | ODLPP | 71.0±4.6 | 78.2±2.4 | 80.7±2.0 | 80.7±1.7 | 78.6±2.2 | 76.7±2.7 |
| | Using 29 components | LDA | 67.7±4.9 | 76.6±2.8 | 82.0±2.6 | 84.5±1.8 | 85.8±1.6 | 87.2±1.8 |
| | | DLPP-L2 | 68.3±4.9 | 76.8±2.7 | 82.3±2.2 | 85.0±1.6 | 86.7±1.7 | 88.4±2.0 |
| | | ODLPP | 71.0±4.6 | 80.0±2.7 | 83.3±2.4 | 85.0±1.2 | 86.8±1.6 | 88.4±1.9 |
| | No PCA Step | DLPP-L1 | 72.7±3.8 | 80.9±2.3 | 85.9±2.0 | 87.5±1.8 | 88.0±1.8 | 88.3±2.6 |
| Capital Letters | Using *N*-*C* components | LDA | 43.3±1.8 | 40.7±2.5 | 33.4±2.0 | 32.3±2.2 | 38.0±2.7 | 44.2±2.0 |
| | | DLPP-L2 | 43.6±1.7 | 41.1±2.5 | 33.6±1.9 | 32.2+2.2 | 38.1±2.6 | 44.3±2.0 |
| | | ODLPP | 51.0±1.9 | 52.2±2.3 | 44.9±1.9 | 43.2±3.1 | 44.6±2.3 | 50.3±2.1 |
| | Using 29 components | LDA | 48.7±2.8 | 61.0±2.3 | 65.7±1.5 | 69.8±2.1 | 70.8±2.1 | 73.2±1.4 |
| | | DLPP-L2 | 49.7±2.8 | 62.0±1.5 | 67.0±1.2 | 71.1+1.7 | 72.4±0.6 | 74.4±1.1 |
| | | ODLPP | 53.4±2.5 | 64.5±1.4 | 68.5±1.4 | 72.6±1.4 | 74.3±1.1 | 76.0±1.3 |
| | No PCA Step | DLPP-L1 | 53.8±2.5 | 64.5±1.3 | 69.3±1.0 | 72.6±1.6 | 74.8±0.9 | 76.6±1.4 |

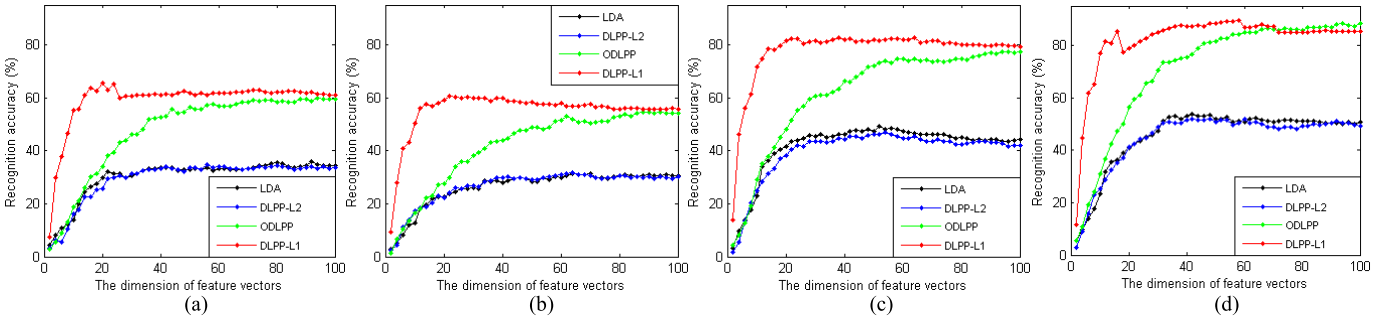


Fig. 5. Recognition accuracy versus feature dimension on FERET database. (a) Test #1. (b) Test #2. (c) Test #3. (d) Test #4.

images called "Ib" are neutral expression and large-angle right view, the images called "Ic" are neutral expression and small-angle right view, the images called "Id" are neutral expression and small-angle left view, the images called "Ie" are neutral expression and large-angle left view, and the images called "If" are smiling and frontal view. In our experiments, the facial portion of each original image is manually cropped according to the location of eyes and mouth, and scaled to $80 \times 80$ pixels. We design four tests called Test #1, Test #2, Test #3, and Test #4. Each test uses three images of every subject for training and the rest for testing. Obviously, the image "Ia" is the most standard and most beneficial to correct recognition of all images. So, we can view images "Ia" as inliers and other images as outliers to some extent. In view of this, the training set of each test includes image "Ia" and two images being left or right view of each subject as follows. In Test #1, the training set contains "Ia," "Ib," and "Ic," in Test #2, the training set contains "Ia," "Id," and "Ie," in Test #3, the training set contains "Ia," "Ic," and "Id," and in Test #4, the training set contains "Ia," "Ib," and "Ie." Fig. 5(a)–(d) show the recognition accuracy versus feature dimension variation on Test #1, Test #2, Test #3, and Test #4, respectively. Moreover, Table II gives the optimal correct recognition rates of each method in four tests. It is clearly found that the performance of our proposed DLPP-L1 is significantly better than those of other three methods when the outlying data are present in the training dataset.

TABLE II

OPTIMAL RECOGNITION RATES (%) ON PARTIAL FERET DATABASE

| Methods | Test #1 | Test #2 | Test #3 | Test #4 |
|---|---|---|---|---|
| LDA | 36.0 | 31.3 | 49.2 | 53.8 |
| DLPP-L2 | 34.7 | 31.7 | 47.0 | 52.0 |
| ODLPP | 59.8 | 54.5 | 77.5 | 88.3 |
| DLPP-L1 | 65.5 | 60.5 | 82.5 | 89.3 |

### *D. Experiments on PolyU Database*

The PolyU palmprint database consists of 600 gray images of 100 palms with six images of each palm. They were collected in two sessions separated by two months: three images in the first session and another three in the second session [25]. The central parts of all images were cropped and aligned using an algorithm similar to that of [38]. Then, each aligned image was resized to $64 \times 64$ pixels and the preprocessed images of one palm are shown in the first row of Fig. 6. In our experiments, these preprocessed images from the first dataset called "Original" because of no occlusion. In addition, we construct the other three datasets based on the "Original" dataset by partially adding occlusion with some rectangle noise as outliers. The rectangle noise consists of black or white dots in random distribution and its location in image is random. The second dataset called "Outlier 1" includes 600 preprocessed images in the "Original" dataset,

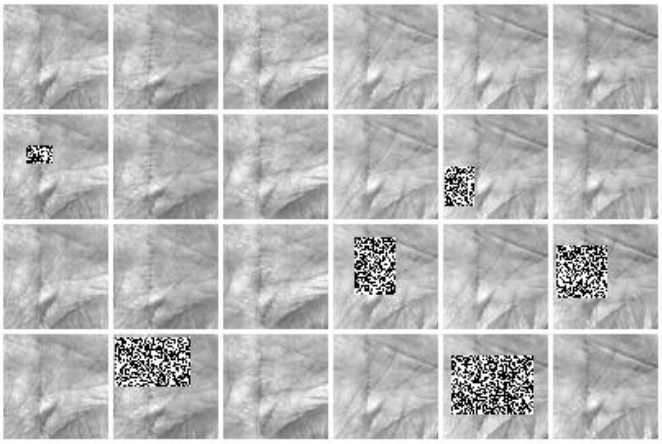

Fig. 6. Images of one palm from four datasets based on the PolyU database. First row: Original. Second row: Outlier 1. Third row: Outlier 2. Fourth row: Outlier 3.

TABLE III
OPTIMAL AVERAGE RECOGNITION RATES (%) AND STANDARD DEVIATION ON FOUR DATASETS BASED ON POLYU DATABASE

| $k$ | Methods | Original | Outlier 1 | Outlier 2 | Outlier 3 |
|---|---|---|---|---|---|
| $k$=2 | LDA | 98.1±2.1 | 94.4±3.2 | 90.0±4.1 | 80.8±4.3 |
| | DLPP-L2 | 98.0±2.6 | 94.6±3.0 | 90.2±4.8 | 80.9±4.1 |
| | ODLPP | 98.3±1.5 | 96.9±2.2 | 93.1±4.0 | 85.9±3.3 |
| | DLPP-L1 | 98.9±1.7 | 97.7±2.6 | 94.3±3.5 | 87.4±4.1 |
| $k$=3 | LDA | 98.7±1.5 | 97.5±2.5 | 95.4±4.6 | 89.9±3.8 |
| | DLPP-L2 | 99.3±0.4 | 97.4±2.7 | 95.3±3.9 | 89.9±3.8 |
| | ODLPP | 99.2±1.6 | 98.0±1.6 | 96.9±4.3 | 90.5±4.0 |
| | DLPP-L1 | 99.9±0.1 | 98.3±2.7 | 97.5±2.6 | 92.0±4.9 |

among which two images of every palm are randomly selected to be added rectangle outliers whose sizes range from $10 \times 10$ to $30 \times 30$. The second row of Fig. 6 shows the corresponding samples of one palm in the second dataset "Outlier 1." Similarly, the third dataset "Outlier 2" and the fourth dataset "Outlier 3" are constructed and the corresponding samples are shown by the third and fourth row of Fig. 6, respectively. In the dataset "Outlier 2," the sizes of the outliers range from $20 \times 20$ to $40 \times 40$. The sizes of the outliers in the dataset "Outlier 3" range from $30 \times 30$ to $50 \times 50$.

The object recognition experiments are, respectively, conducted on the above four datasets and all images are normalized by mapping each image's means to 0 and deviations to 1 before the learning algorithms extract projection vectors. Similar to the experiments of Section IV-B, we randomly select $k$ ($k = 2, 3$) images of each palm to form the training set and use the rest images as the testing set. For each $k$, we repeat the experiments ten times and record the optimal correct recognition rates of four methods. Then, the average optimal recognition rates and the standard deviations are calculated. The results on the four datasets are shown in Table III. Table III clearly shows that the recognition accuracies of other three methods decrease more remarkably than that of DLPP-L1 with more and more occlusion of partial palm images. These results also show that DLPP-L1 is more robust to outliers than other three methods.

## V. CONCLUSION

In this paper, a new method, called discriminant locality preserving projections based on L1-norm maximization (DLPP-L1), is proposed. DLPP-L1 aims to learn a set of local optimal projection vectors by maximizing the ratio of the L1-norm-based locality preserving between-class dispersion and the L1-norm-based locality preserving within-class dispersion. DLPP-L1 is more robust to outliers than conventional L2-norm-based DLPP because L1-norm can significantly suppress the negative effects of outliers to some extent. The experimental results on two artificial datasets, Binary Alphadigits database, FERET face database, and PolyU palmprint database have demonstrated that DLPP-L1 is more robust to outliers than LDA, DLPP-L2, and ODLPP. However, the iteration procedure results in that DLPP-L1 has more computational cost than other three methods. In the future, we try to exploit a more efficient algorithm to find the global optimal solution of DLPP-L1.

## ACKNOWLEDGMENT

The authors would like to thank the associate editor and all reviewers for their constructive comments and advice.

## REFERENCES

[1] H. Murase, F. Kimura, M. Yoshimura, and Y. Miyake, "An improvement of the auto-correlation matrix in pattern matching method and its application to handprinted 'HIRAGANA'," *Trans. IECE*, vol. 8, no. 3, pp. 1–15, 1981.
[2] G. W. Cottrell and M. K. Fleming, "Face recognition using unsupervised feature extraction," in *Proc. Int. Neural Netw. Conf.*, 1990, pp. 322–325.
[3] M. Turk and A. Pentland, "Face recognition using eigenfaces," in *Proc. IEEE Conf. Comput. Vis. Pattern Recognit.*, 1991, pp. 586–591.
[4] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cognit. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.
[5] A. Iosifidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Multi-view human movement recognition based on fuzzy distances and linear discriminant analysis," *Comput. Vis. Image Understand.*, vol. 116, no. 3, pp. 347–360, 2012.
[6] P. N. Belhumeur, J. P. 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, Jul. 1997.
[7] W. Zhao, R. Chellappa, and A. Krishnaswamy, "Discriminant analysis of principal components for face recognition," in *Proc. 3rd IEEE Int. Conf. Autom. Face Gesture Recognit.*, Nara, Japan, Apr. 1998, pp. 336–341.
[8] D. L. Swets and J. J. Weng, "Using discriminant eigenfeatures for image retrieval," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 18, no. 8, pp. 831–836, Aug. 1996.
[9] I. T. Jolliffe, *Principal Component Analysis*. New York, NY, USA: Springer-Verlag, 1986.
[10] R. A. Fisher, "The use of multiple measures in taxonomic problems," *Ann. Eugenics*, vol. 7, no. 2, pp. 179–188, Sep. 1936.
[11] J. B. Tenenbaum, V. de Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, 2000.
[12] S. T. Roweis and L. K. Saul, "Nonlinear dimension reduction by locally linear embedding," *Science*, vol. 290, pp. 2323–2326, Dec. 2000.
[13] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural Comput.*, vol. 15, no. 6, pp. 1373–1396, 2003.
[14] X. He, S. Yan, Y. Hu, P. Niyogi, and H.-J. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.
[15] Y. Pang, L. Zhang, Z. Liu, N. Yu, and H. Li, "Neighborhood preserving projections (NPP): A novel linear dimension reduction method," *Lecture Notes Comput. Sci.*, vol. 36, no. 1, pp. 117–125, 2005.

[16] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. 10th IEEE Int. Conf. Comput. Vis.*, Beijin, China, Oct. 2005, pp. 1208–1213.
[17] X. He, S. Yan, Y. Hu, and H. Zhang, "Learning a locality preserving subspace for visual recognition," in *Proc. 9th Int. Conf. Comput. Vis.*, Nice, France, Oct. 2003, pp. 385–392.
[18] H. Hu, "Orthogonal neighborhood preserving discriminant analysis for face recognition," *Pattern Recognit.*, vol. 41, no. 6, pp. 2045–2054, 2008.
[19] W. Yu, X. Teng, and C. Liu, "Face recognition using discriminant locality preserving projections," *Image Vis. Comput.*, vol. 24, no. 3, pp. 239–248, 2006.
[20] L. Zhu and S. Zhu, "Face recognition based on orthogonal discriminant locality preserving projections," *Neurocomputing*, vol. 70, nos. 7–9, pp. 1543–1546, 2007.
[21] L. Yang, W. Gong, X. Gu, W. Li, and Y. Liang, "Null space discriminant locality preserving projections for face recognition," *Neurocomputing*, vol. 71, nos. 16–18, pp. 3644–3649, 2008.
[22] G. Lu, L. Zhong, and J. Zhong, "Face recognition using discriminant locality preserving projections based on maximum margin criterion," *Pattern Recognit.*, vol. 43, no. 10, pp. 3572–3579, 2010.
[23] R. Zhi and Q. Ruan, "Facial expression recognition based on two-dimensional discriminant locality preserving projections," *Neurocomputing*, vol. 71, no. 7, pp. 1730–1734, 2008.
[24] W. Yu, "Two-dimensional discriminant locality preserving projections for face recognition," *Pattern Recognit. Lett.*, vol. 30, no. 15, pp. 1378–1383, 2009.
[25] J. Lu and Y. Tan, "Improved discriminant locality preserving projections for face and palmprint recognition," *Neurocomputing*, vol. 74, no. 18, pp. 3760–3767, 2011.
[26] M. J. Black and A. Rangarajan, "On the unification of line processes, outlier rejection, and robust statistics with applications in early vision," *Int. J. Comput. Vis.*, vol. 19, no. 1, pp. 57–91, 1996.
[27] Q. Ke and T. Kanade, "Robust subspace computation using L1 norm," Dept. Math., Carnegie Mellon Univ., Pittsburgh, PA, USA, Tech. Rep. CMU-CS-03-172, Aug. 2003.
[28] A. Baccini, P. Besse, and A.D. Falguerolles, "A L1-norm PCA and a heuristic approach, ordinal and symbolic data analysis," in *Ordinal and Symbolic Data Analysis*, E. Diday, Y. Lechevalier, and P. Opitz, Eds. New York, NY, USA: Springer-Verag, 1996, pp. 359–368.
[29] H. Wang, Q. Tang, and W. Zheng, "L1-norm-based common spatial patterns," *IEEE Trans. Biomed. Eng.*, vol. 59, no. 3, pp. 653–662, Mar. 2012.
[30] Y. Pang, X. Li, and Y. Yuan, "Robust tensor analysis with L1-norm," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 20, no. 2, pp. 172–178, Feb. 2010.
[31] N. Kwak, "Principal component analysis based on L1-norm maximization," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 30, no. 9, pp. 672–1680, May 2008.
[32] X. Li, Y. Pang, and Y. Yuan, "L1-norm-based 2DPCA," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 40, no. 4, pp. 1170–1175, Aug. 2009.
[33] Y. Pang and Y. Yuan, "Outlier-resisting graph embedding," *Neurocomputing*, vol. 73, pp. 968–974, Aug. 2010.
[34] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[35] R. Jenatton, G. Obozinski, and F. Bach, "Structured sparse principal component analysis," in *Proc. 13th Int. Conf. Artif. Intell. Statist.*, Sardinia, Italy, May 2010, pp. 1–8.
[36] (2009). *Binary Alphadigits Database* [Online]. Available: http://www.cs.nyu.edu/∼roweis/data.html
[37] P. J. Phillips, H. Wechsler, J. Huang, and P. J. Rauss, "The FERET database and evaluation procedure for face recognition algorithms," *Image Vis. Comput.*, vol. 16, no. 5, pp. 295–306, Jan. 1998.
[38] D. Zhang, W. K. Kong, J. You, and M. Wong, "Online palmprint identification," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 9, pp. 1041–1050, Sep. 2003.
[39] (2006). *The PolyU Palmprint Database* [Online]. Available: http://www.comp.polyu.edu.hk/biometrics
[40] C. J. Veeman and M. J. T. Reinders, "The nearest subclass classifier: A compromise between the nearest mean and nearest neighbor classifier," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 9, pp. 1417–1429, Sep. 2005.

**Fujin Zhong** received the B.S. and M.S. degrees from the Hefei University of Technology, Hefei, China, in 2002 and 2005, respectively. He is currently pursuing the Doctoral degree in pattern recognition and biometrics from the Sichuan Province Key Laboratory of Signal and Information Processing, Southwest Jiaotong University, Chengdu, China.

He joined the School of Computer and Information Engineering, Yibin University, Yibin, China, in 2005. His current research interests include digital signal processing, pattern recognition, and biometrics and security.

**Jiashu Zhang** received the B.S. and Ph.D. degrees from the University of Electronic Science and Technology of China, Chengdu, China, in 1987 and 2001, respectively.

He joined the School of Information Science and Technology, Southwest Jiaotong University, Chengdu, in 2001, where he is a Professor and the Director of the Sichuan Province Key Laboratory of Signal and Information Processing. His current research interests include digital signal processing, information forensic and data hiding, biometrics and security, and chaos theory with application to electronic engineering.

**Defang Li** received the B.S. degree from Southwest University, Chongqing, China, in 1988.

She joined the Psychological Research and Consulting Center, Southwest Jiaotong University, Chengdu, China, in 2001. Her current research interests include cognitive psychology and facial affective recognition.