# Discriminative Locality Alignment

Tianhao Zhang[1], Dacheng Tao[2,3], and Jie Yang[1]

[1] Institute of Image Processing and Pattern Recognition,
Shanghai Jiao Tong University, Shanghai, China
[2] School of Computer Engineering, Nanyang Technological University,
50 Nanyang Avenue, Singapore
[3] College of Computer Science, Zhejiang University, China
z.tianhao@gmail.com, dacheng.tao@gmail.com, jieyang@sjtu.edu.cn

**Abstract.** Fisher's linear discriminant analysis (LDA), one of the most popular dimensionality reduction algorithms for classification, has three particular problems: it fails to find the nonlinear structure hidden in the high dimensional data; it assumes all samples contribute equivalently to reduce dimension for classification; and it suffers from the matrix singularity problem. In this paper, we propose a new algorithm, termed Discriminative Locality Alignment (DLA), to deal with these problems. The algorithm operates in the following three stages: first, in part optimization, discriminative information is imposed over patches, each of which is associated with one sample and its neighbors; then, in sample weighting, each part optimization is weighted by the *margin degree*, a measure of the importance of a given sample; and finally, in whole alignment, the alignment trick is used to align all weighted part optimizations to the whole optimization. Furthermore, DLA is extended to the semi-supervised case, i.e., semi-supervised DLA (SDLA), which utilizes unlabeled samples to improve the classification performance. Thorough empirical studies on the face recognition demonstrate the effectiveness of both DLA and SDLA.

## 1 Introduction

Dimensionality reduction is the process of transforming data from a high dimensional space to a low dimensional space to reveal the intrinsic structure of the distribution of data. It plays a crucial role in the field of computer vision and pattern recognition as a way of dealing with the "curse of dimensionality". In past decades, a large number of dimensionality reduction algorithms have been proposed and studied. Among them, principal components analysis (PCA) [9] and Fisher's linear discriminant analysis (LDA) [6] are two of the most popular linear dimensionality reduction algorithms.

PCA [9] maximizes the mutual information between original high dimensional Gaussian distributed data and projected low dimensional data. PCA is optimal for reconstruction of Gaussian distributed data. However it is not optimal for classification [14] problems. LDA overcomes this shortcoming by utilizing the class label information. It finds the projection directions that maximize the trace

D. Forsyth, P. Torr, and A. Zisserman (Eds.): ECCV 2008, Part I, LNCS 5302, pp. 725–738, 2008.

of the between-class scatter matrix and minimize the trace of the within-class scatter matrix simultaneously. While LDA is a good algorithm to be applied for classification, it also has several problems as follows.

First, LDA considers only the global Euclidean structure, so it cannot discover the nonlinear structure hidden in the high dimensional non-Gaussian distributed data. Numerous manifold learning algorithms have been developed as a promising tool for analyzing the high dimensional data that lie on or near a submanifold of the observation space. Representative works include locally linear embedding (LLE) [11], Laplacian eigenmaps (LE) [2], local tangent space alignment (LTSA) [19], locality preserving projections (LPP) [8]. These algorithms, which aim to preserve the local geometry of samples, can attack the nonlinear distribution of data.

Second, LDA is in fact based on the assumption that all samples contribute equivalently for discriminative dimensionality reduction, although samples around the margins, i.e., marginal samples, are more important in classification than inner samples. A recently developed algorithm, which breaks through the assumption of equal contributions, is marginal Fisher analysis (MFA) [16]. MFA uses only marginal samples to construct the penalty graph which characterizes the interclass separability. However, it does not give these marginal samples specific weights to describe how important each is. Furthermore, MFA may lose discriminative information since it completely ignores inner samples in constructing the penalty graph.

Finally, LDA suffers from the matrix singularity problem since the between-class scatter matrix is often singular. Many algorithms have been proposed to deal with this, such as PCA plus LDA [1], direct LDA (DLDA) [18], and null-space LDA (NLDA) [5]. However, all of them may fail to consider all possible discriminative information in selecting discriminative subspace.

Most importantly, in the context of this work, almost all existing variants of LDA respond to just one or two of LDA's suite of problems, yet they may remain open to others. In order to overcome all aforementioned problems in LDA simultaneously, a new linear algorithm termed Discriminative Locality Alignment (DLA) is proposed. The algorithm operates in the following three stages: 1) the part optimization stage, 2) the sample weighting stage, and 3) the whole alignment stage. First, discriminative information is imposed over patches, each of which is associated with one sample and its neighbors; then each part optimization is weighted by *margin degree*, a measure of the importance of a given sample for classification; and finally the alignment trick [19,20,21] is used to align all of the weighted part optimizations to the whole optimization.DLA has three particular advantages: 1) because it focuses on the local patch of each sample, it can deal with the nonlinearity of the distribution of samples while preserving the discriminative information; 2) since the importance of marginal samples is enhanced for discriminative subspace selection, it learns low dimensional representations for classification properly; and 3) because it obviates the need to compute the inverse of a matrix, it has no the matrix singularity problem. In addition, we extend DLA

to the semi- supervised learning case, i.e., semi-supervised DLA (SDLA), by incorporating the part optimizations of the unlabeled samples.

The rest of the paper is organized as follows. Section 2 details the proposed DLA algorithm. Section 3 extends DLA to SDLA. Section 4 gives our experimental results. Section 5 concludes.

## 2 Discriminative Locality Alignment

Consider a set of samples $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N] \in \mathbb{R}^{m \times N}$, and each sample $\boldsymbol{x}_i$ belongs to one of the $C$ classes. The problem of linear dimensionality reduction is to find a projection matrix $U$ that maps $X \in \mathbb{R}^{m \times N}$ to $Y \in \mathbb{R}^{d \times N}$, i.e., $Y = U^T X$, where $d < m$. In this section, Discriminative Locality Alignment (DLA) is proposed to overcome problems in LDA for linear dimensionality reduction.

DLA operates in three stages. In the first stage, for each sample in the dataset, one patch is built by the given sample and its neighbors which including the samples from not only a same class but also different classes from the given sample. On each patch, an objective function is designed to preserve the local discriminative information. Since each sample can be seen as a part of the whole dataset, the stage is termed "part optimization". In the second stage, *margin degree* is defined for each sample as a measure of the sample importance in contributing classification. Then, each part optimization obtained from the first stage is weighted based on the *margin degree*. The stage termed "sample weighting". In the final stage, termed "whole alignment", all the weighted part optimizations are integrated into together to form a global coordinate according to the alignment trick [19,20,21]. The projection matrix can be obtained by solving a standard eigen-decomposition problem.

### 2.1 Part Optimization

For a given sample $\boldsymbol{x}_i$, according to the class label information, we can divide the other ones into the two groups: samples in the same class with $\boldsymbol{x}_i$ and samples from different classes with $\boldsymbol{x}_i$ . We can select $k_1$ nearest neighbors with respect to $\boldsymbol{x}_i$ from samples in the same class with $\boldsymbol{x}_i$ and term them *neighbor samples from an identical class*: $\boldsymbol{x}_{i^1}, \cdots, \boldsymbol{x}_{i^{k_1}}$. We select $k_2$ nearest neighbors with respect to $\boldsymbol{x}_i$ from samples in different classes with $\boldsymbol{x}_i$ and term them *neighbor samples from different classes*: $\boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_2}}$. The local patch for the sample $\boldsymbol{x}_i$ is constructed by putting $\boldsymbol{x}_i, \boldsymbol{x}_{i^1}, \cdots, \boldsymbol{x}_{i^{k_1}}$, and $\boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_2}}$ together as $X_i = [\boldsymbol{x}_i, \boldsymbol{x}_{i^1}, \cdots, \boldsymbol{x}_{i^{k_1}}, \boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_2}}]$.

For each patch, the corresponding output in the low dimensional space is $Y_i = [\boldsymbol{y}_i, \boldsymbol{y}_{i^1}, \cdots, \boldsymbol{y}_{i^{k_1}}, \boldsymbol{y}_{i_1}, \cdots, \boldsymbol{y}_{i_{k_2}}]$. In the low dimensional space, we expect that distances between the given sample and *neighbor samples from an identical class* are as small as possible, while distances between the given sample and *neighbor samples from different classes* are as large as possible, as illustrated in Figure 1. The left part of the figure shows the $i^{th}$ patch in the original high dimensional space and the patch consists of $\boldsymbol{x}_i$, *neighbor samples from an identical class* (i.e.,

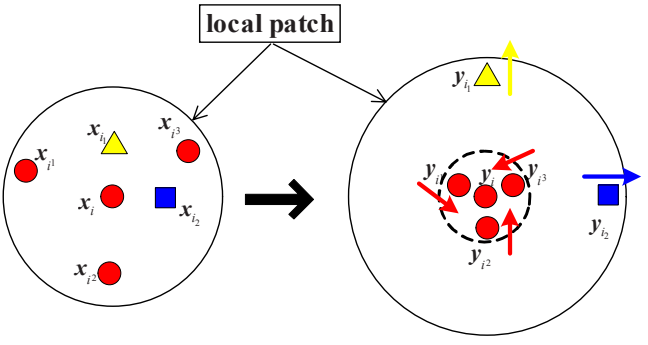


**Fig. 1.** The part optimization stage in DLA

$\boldsymbol{x}_{i^1}$, $\boldsymbol{x}_{i^2}$, and $\boldsymbol{x}_{i^3}$), and *neighbor samples from different classes* (i.e., $\boldsymbol{x}_{i_1}$ and $\boldsymbol{x}_{i_2}$). The expected results on the patch in the low dimensional space are shown as the right part of the figure. Low dimensional samples $\boldsymbol{y}_{i^1}$, $\boldsymbol{y}_{i^2}$, and $\boldsymbol{y}_{i^3}$ are as close as possible to $\boldsymbol{y}_i$, while low dimensional samples $\boldsymbol{y}_{i_1}$ and $\boldsymbol{y}_{i_2}$ are as far as possible away from $\boldsymbol{y}_i$.

For each patch in the low dimensional space, we expect that distances between $\boldsymbol{y}_i$ and *neighbor samples from an identical class* are as small as possible, so we have:

$$\arg\min_{\boldsymbol{y}_i} \sum_{j=1}^{k_1} \|\boldsymbol{y}_i - \boldsymbol{y}_{i^j}\|^2. \tag{1}$$

Meanwhile, we expect that distances between $\boldsymbol{y}_i$ and *neighbor samples from different classes* are as large as possible, so we have:

$$\arg\max_{\boldsymbol{y}_i} \sum_{p=1}^{k_2} \left\|\boldsymbol{y}_i - \boldsymbol{y}_{i_p}\right\|^2. \tag{2}$$

Since the patch built by the local neighborhood can be regarded approximately Euclidean [11], we formulate the part discriminator by using the linear manipulation:

$$\arg\min_{\boldsymbol{y}_i} \left( \sum_{j=1}^{k_1} \|\boldsymbol{y}_i - \boldsymbol{y}_{i^j}\|^2 - \beta \sum_{p=1}^{k_2} \|\boldsymbol{y}_i - \boldsymbol{y}_{i_p}\|^2 \right), \tag{3}$$

where $\beta$ is a scaling factor in $[0, 1]$ to unify the different measures of the within-class distance and the between-class distance. Define the coefficients vector

$$\boldsymbol{\omega}_i = \left[ \overbrace{1, \cdots, 1,}^{k_1} \quad \overbrace{-\beta, \cdots, -\beta}^{k_2} \right]^T, \tag{4}$$

then, Eq. (3) reduces to:

$$\begin{aligned}
&\arg\min_{\boldsymbol{y}_i}\left(\sum_{j=1}^{k_1}\|\boldsymbol{y}_i-\boldsymbol{y}_{i^j}\|^2(\boldsymbol{\omega}_i)_j+\sum_{p=1}^{k_2}\|\boldsymbol{y}_i-\boldsymbol{y}_{i_p}\|^2(\boldsymbol{\omega}_i)_{p+k_1}\right)\\
=&\arg\min_{\boldsymbol{y}_i}\left(\sum_{j=1}^{k_1+k_2}\left\|\boldsymbol{y}_{F_i\{1\}}-\boldsymbol{y}_{F_i\{j+1\}}\right\|^2(\boldsymbol{\omega}_i)_j\right)\\
=&\arg\min_{Y_i}\operatorname{tr}\left(Y_i\begin{bmatrix}-\boldsymbol{e}_{k_1+k_2}^T\\ I_{k_1+k_2}\end{bmatrix}\operatorname{diag}(\boldsymbol{\omega}_i)\begin{bmatrix}-\boldsymbol{e}_{k_1+k_2}\ I_{k_1+k_2}\end{bmatrix}Y_i^T\right)\\
=&\arg\min_{Y_i}\operatorname{tr}\left(Y_iL_iY_i^T\right),
\end{aligned}\tag{5}$$

where $F_i=\left\{i,i^1,\cdots,i^{k_1},i_1,\cdots,i_{k_2}\right\}$ is the index set for the $i^{th}$ patch; $\boldsymbol{e}_{k_1+k_2}=[1,\cdots,1]^T\in\mathbb{R}^{k_1+k_2}$; $I_{k_1+k_2}$ is the $(k_1+k_2)\times(k_1+k_2)$ identity matrix; $\operatorname{diag}(\cdot)$ is the diagonalization operator; $L_i$ encapsulates both the local geometry and the discriminative information, and it is given by

$$L_i=\begin{bmatrix}\sum_{j=1}^{k_1+k_2}(\boldsymbol{\omega}_i)_j & -\boldsymbol{\omega}_i^T\\ -\boldsymbol{\omega}_i & \operatorname{diag}(\boldsymbol{\omega}_i)\end{bmatrix}.\tag{6}$$

### 2.2 Sample Weighting

In general, samples around classification margins have a higher risk of being misclassified than samples far away from margins. As shown in Figure 2, $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$, which are lying around the nearby classification margin, are more important than $\boldsymbol{x}_3$ in seeking a subspace for classification.

To quantify the importance of a sample $\boldsymbol{x}_i$ for discriminative subspace selection, we need to find a measure, termed *margin degree* $m_i$. For a sample, its *margin degree* should be proportional to the number of samples with different class labels from the label of the sample but in the $\epsilon$-ball centered at the sample. Therefore, a possible definition of the *margin degree* $m_i$ for the $i^{th}$ sample $\boldsymbol{x}_i$ is

$$m_i=\exp\left(-\frac{1}{(n_i+\delta)t}\right)\qquad i=1,\cdots,N,\tag{7}$$

where $n_i$ is the number of samples $\boldsymbol{x}_j$ in the $\epsilon$-ball centered at $\boldsymbol{x}_i$ with labels $l(\boldsymbol{x}_j)$ different from the label of $\boldsymbol{x}_j$; $l(\boldsymbol{x})$ is the class label of the sample $\boldsymbol{x}$; $\delta$ is a regularization parameter; and $t$ is a scaling factor. In Figure 2, for a fixed $\epsilon$, $n_1=4$ because there are 4 samples with different class labels from that of $\boldsymbol{x}_1$ in the $\epsilon$-ball centered at $\boldsymbol{x}_1$ ; $n_2=1$ because there are 1 sample with different class label from that of $\boldsymbol{x}_2$ in the $\epsilon$-ball centered at $\boldsymbol{x}_2$; $n_3=0$ because there are no sample with different class label from that of $\boldsymbol{x}_3$ in the $\epsilon$-ball centered at $\boldsymbol{x}_3$. According to Eq.(7), the corresponding *margin degrees* of these three samples are ordered as $m_1>m_2>m_3$.

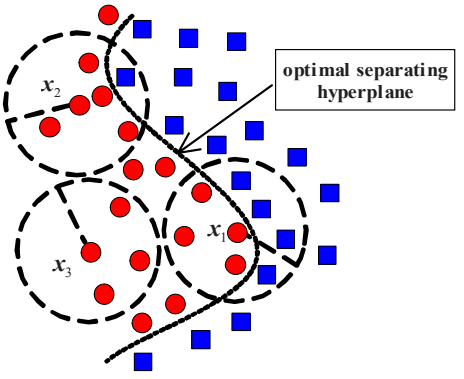


**Fig. 2.** Illustration for sample weighting

In DLA, the part optimization of the $i^{th}$ patch is weighted by the *margin degree* of the $i^{th}$ sample before the whole alignment stage, i.e.,

$$\arg\min_{Y_i} m_i \mathrm{tr}\left(Y_i L_i Y_i^T\right) = \arg\min_{Y_i} \mathrm{tr}\left(Y_i m_i L_i Y_i^T\right). \tag{8}$$

### 2.3 Whole Alignment

For each patch $X_i$, $i = 1, \cdots, N$, we have the weighted part optimizations described as Eq. (8). In this subsection, these optimizations will be unified together as a whole one by assuming that the coordinate for the $i^{th}$ patch $Y_i = \left[\boldsymbol{y}_i, \boldsymbol{y}_{i1}, \cdots, \boldsymbol{y}_{ik_1}, \boldsymbol{y}_{i_1}, \cdots, \boldsymbol{y}_{i_{k_2}}\right]$ is selected from the global coordinate $Y = [\boldsymbol{y}_1, \cdots, \boldsymbol{y}_N]$, such that

$$Y_i = Y S_i, \tag{9}$$

where $S_i \in \mathbb{R}^{N \times (k_1+k_2+1)}$ is the selection matrix and an entry is defined as:

$$(S_i)_{pq} = \begin{cases} 1 \text{ if } p = F_i\{q\} \\ 0 \quad \text{else}. \end{cases} \tag{10}$$

Then, Eq. (8) can be rewritten as:

$$\arg\min_{Y} \mathrm{tr}\left(Y S_i m_i L_i S_i^T Y^T\right). \tag{11}$$

By summing over all part optimizations described as Eq. (11) together, we can obtain the whole alignment as:

$$\begin{aligned} \arg\min_{Y} \sum_{i=1}^{N} \mathrm{tr}\left(Y S_i m_i L_i S_i^T Y^T\right) &= \arg\min_{Y} \mathrm{tr}\left(Y \left(\sum_{i=1}^{N} S_i m_i L_i S_i^T\right) Y^T\right) \\ &= \arg\min_{Y} \mathrm{tr}\left(Y L Y^T\right), \end{aligned} \tag{12}$$

where $L = \sum_{i=1}^{N} S_i m_i L_i S_i^T \in \mathbb{R}^{N \times N}$ is the alignment matrix [19]. It is obtained based on an iterative procedure:

$$L\left(F_i, F_i\right) \leftarrow L\left(F_i, F_i\right) + m_i L_i, \tag{13}$$

for $i = 1, \cdots, N$, with the initialization $L = 0$.

To obtain the linear and orthogonal projection matrix $U$, such as $Y = U^T X$, we can impose $U^T U = I_d$, where $I_d$ is the $d \times d$ identity matrix. Eq. (12) is deformed as:

$$\arg\min_{U} \operatorname{tr}\left(U^T X L X^T U\right) \text{ s.t. } U^T U = I_d. \tag{14}$$

Obviously, solutions of Eq.(14) are given by using the standard eigen-decomposition:

$$X L X^T \boldsymbol{u} = \lambda \boldsymbol{u}. \tag{15}$$

Let the column vectors $\boldsymbol{u}_1, \boldsymbol{u}_2, \cdots, \boldsymbol{u}_d$ be the solutions of Eq. (15), ordered according to eigenvalues, $\lambda_1 < \lambda_2 < \cdots < \lambda_d$. The optimal projection matrix $U$ is then given by: $U = [\boldsymbol{u}_1, \boldsymbol{u}_2 \cdots, \boldsymbol{u}_d]$.

Different from algorithms, e.g., LDA [1], LPP [8], and MFA [16], which lead to a generalized eigenvalue problem, DLA successfully avoids the matrix singularity problem since it has no inverse operation over a matrix. However, the PCA step is still recommended to reduce noise. The procedure of DLA is listed as following:

1. Use PCA to project the dataset $X$ into the subspace for eliminating the useless information. To make it clear, we still use $X$ to denote the dataset in the PCA subspace in the following steps. We denote by $U_{PCA}$ the PCA projection matrix;
2. For each sample $\boldsymbol{x}_i$ in dataset $X$, $i = 1, \cdots, N$, search $k_1$ *neighbor samples from an identical class* and $k_2$ neighbor samples from different classes, and then build the patch $X_i = [\boldsymbol{x}_i, \boldsymbol{x}_{i1}, \cdots, \boldsymbol{x}_{i^{k_1}}, \boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_2}}]$;
3. Compute $L_i$ by Eq. (6), and $m_i$ by Eq. (7). Construct the alignment matrix $L$ by the iterative procedure described by Eq. (13); and
4. Solve the standard eigen-decomposition: $X L X^T \boldsymbol{u} = \lambda \boldsymbol{u}$ to obtain the DLA projection matrix $U_{DLA} = [\boldsymbol{u}_1, \boldsymbol{u}_2 \cdots, \boldsymbol{u}_d]$, whose vectors are the eigenvectors corresponding to the $d$ smallest eigenvalues. The final projection matrix is as follows: $U = U_{PCA} U_{DLA}$.

## 3 Semi-supervised DLA

Recent researches [3,22] show that unlabeled samples may be helpful to improve the classification performance. In this section, we generalize DLA by introducing new part optimizations by taking unlabeled samples into account and then incorporating them to the whole alignment stage as semi-supervised DLA (SDLA). The unlabeled samples are attached to the original labeled samples as: $X = [\boldsymbol{x}_1, \cdots, \boldsymbol{x}_N, \boldsymbol{x}_{N+1}, \cdots, \boldsymbol{x}_{N+N_U}]$, where the first $N$ samples are labeled and the left $N_U$ ones are unlabeled. The part optimization for each labeled sample is given by Eq. (8).

Unlabeled samples are valuable to enhance the local geometry. For each unlabeled sample $\boldsymbol{x}_i,\ \ i = N+1, \cdots, N+N_U$, we search its $k_S$ nearest neighbors $\boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_S}}$ in all training samples including both labeled and unlabeled ones. Let $X_i = [\boldsymbol{x}_i, \boldsymbol{x}_{i_1}, \cdots, \boldsymbol{x}_{i_{k_S}}]$ denote the $i^{th}$ patch and the associated index set is given by $F_i^U = \{i, i_1, \cdots, i_{k_S}\}$. To capture the local geometry of the $i^{th}$ patch, we expect nearby samples remain nearby, or $\boldsymbol{y}_i \in \mathbb{R}^d$ is close to $\boldsymbol{y}_{i_1}, \cdots, \boldsymbol{y}_{i_{k_S}}$, i.e.,

$$\begin{aligned}
&\arg\min_{\boldsymbol{y}_i} \sum_{j=1}^{k_S} \left\| \boldsymbol{y}_i - \boldsymbol{y}_{i_j} \right\|^2 \\
=\ &\arg\min_{\boldsymbol{y}_i} \operatorname{tr} \left( \begin{bmatrix} (\boldsymbol{y}_i - \boldsymbol{y}_{i_1})^T \\ \vdots \\ \left(\boldsymbol{y}_i - \boldsymbol{y}_{i_{k_S}}\right)^T \end{bmatrix} \left[ \boldsymbol{y}_i - \boldsymbol{y}_{i_1}, \cdots, \boldsymbol{y}_i - \boldsymbol{y}_{i_{k_S}} \right] \right) \\
=\ &\arg\min_{Y_i} \operatorname{tr} \left( Y_i \begin{bmatrix} -\boldsymbol{e}_{k_S}^T \\ I_{k_S} \end{bmatrix} \left[ -\boldsymbol{e}_{k_S}\ I_{k_S} \right] Y_i^T \right) \\
=\ &\arg\min_{Y_i} \operatorname{tr} \left( Y_i L_i^U Y_i^T \right),
\end{aligned} \tag{16}$$

where, $\boldsymbol{e}_{k_S} = [1, \cdots, 1]^T \in \mathbb{R}^{k_S}$; $I_{k_S}$ is the $k_S \times k_S$ identity matrix; and

$$L_i^U = \begin{bmatrix} -\boldsymbol{e}_{k_S}^T \\ I_{k_S} \end{bmatrix} \left[ -\boldsymbol{e}_{k_S}\ I_{k_S} \right] = \begin{bmatrix} k_s & -\boldsymbol{e}_{k_S}^T \\ -\boldsymbol{e}_{k_S} & I_{k_S} \end{bmatrix}. \tag{17}$$

Since the unlabeled samples cannot provide the margin information, the sample weighting stage is omitted for unlabeled ones in SDLA. Putting all samples together, we have:

$$\begin{aligned}
&\arg \sum_{i=1}^{N} \min_{Y_i} \operatorname{tr} \left( Y_i m_i L_i Y_i^T \right) + \gamma \arg \sum_{i=N+1}^{N+N_U} \min_{Y_i} \operatorname{tr} \left( Y_i L_i^U Y_i^T \right) \\
=\ &\arg\min_{Y} \operatorname{tr} \left( Y \left( \sum_{i=1}^{N} S_i^L m_i L_i \left(S_i^L\right)^T + \sum_{i=N+1}^{N+N_U} S_i^U \gamma L_i^U \left(S_i^U\right)^T \right) Y^T \right) \\
=\ &\arg\min_{Y} \operatorname{tr} \left( Y L^S Y^T \right),
\end{aligned} \tag{18}$$

where $\gamma$ is a control parameter; $S_i^L \in \mathbb{R}^{(N+N_U)\times(k_1+k_2+1)}$and $S_i^U \in \mathbb{R}^{(N+N_U)\times(k_S+1)}$ are the selection matrices defined similarly as in Section 2.3; and $L^S \in \mathbb{R}^{(N+N_U)\times(N+N_U)}$ is the alignment matrix constructed by

$$\begin{cases} L^S\left(F_i, F_i\right) \leftarrow L^S\left(F_i, F_i\right) + m_i L_i, \text{ for } i = 1, \cdots, N \\ L^S\left(F_i^U, F_i^U\right) \leftarrow L^S\left(F_i^U, F_i^U\right) + \gamma L_i^U, \text{ for } i = N+1, \cdots, N+N_U, \end{cases} \tag{19}$$

with the initialization $L^S = 0$.

Similar to Section 2.3, the problem is converted to a standard eigenvalue decomposition: $XL^SX^T\boldsymbol{u} = \lambda\boldsymbol{u}$. The projection matrix $U_{SDLA}$ contains eigenvectors associated with the $d$ smallest eigenvalues. Similar to DLA, PCA is also utilized to reduce sample noise, and the final projection matrix is $U = U_{PCA}U_{SDLA}$.

## 4 Experiments

In this section, we compare the proposed DLA algorithm against representative dimensionality reduction algorithms, e.g., PCA [15], LDA [1], SLPP (LPP1 in [4]), and MFA [16]. We also study the performance of DLA by varying parameters $k_1$ (the number of *neighbor samples from an identical class*) and $k_2$ (the number of *neighbor samples from different classes*) which are crucial in building patches. Finally, the SDLA algorithm is evaluated by comparing with the original DLA. To begin with, we briefly introduce the three steps for the recognition problems.

First, we perform each of the involved algorithms on training samples to learn projection matrices. Second, each testing sample is projected to a low dimensional subspace via a projection matrix. Finally, the *nearest neighbor* (NN) classifier is used to recognize testing samples in the projected subspace.

### 4.1 Data

Three face image databases: UMIST [7], YALE [1], and FERET [10] are utilized for empirical study. The UMIST database consists of 564 face images from 20 subjects. The individuals are a mix of race, sex and appearance and are photographed in a range of poses from profile to frontal views. The YALE database contains face images collected from 15 individuals, 11 images for each individual and showing varying facial expressions and configurations. The FERET database contains 13,539 face images from 1,565 subjects. The images vary in size, pose, illumination, facial expression and age.

For UMIST and YALE, all face images are used in the experiments. For FERET, we randomly select 100 individuals, each of which has 7 images. All images from three databases are cropped with reference to the eyes and cropped images are normalized to $40 \times 40$ pixel arrays with 256 gray levels per pixel.

**Fig. 3.** Sample images. The first row comes from UMIST [7]; the second row comes from YALE [1]; and the third row comes from FERET [10].

Figure 3 shows sample images from these three databases. Each image is reshaped to one long vector by arranging its pixel values in a fixed order.

### 4.2 General Experiments

We compare the proposed DLA with two different settings, i.e., DLA1 and DLA2, to well-known related dimensionality reduction algorithms, which are PCA [15], LDA [1], SLPP (LPP1 in [4]), and MFA [16], in terms of effectiveness. For DLA1, we set $t = \infty$ in Eq. (7), while in DLA2, $t$ is determined empirically.

For all algorithms except PCA, the first step is PCA projection. In the following experiments, we project samples to the PCA subspace with $N-1$ dimensions for SLPP [4], DLA1, and DLA2. For LDA [1] and MFA [16], we retain $N - C$ dimensions in the PCA step.

For UMIST and YALE, we randomly select $p$ (= 3, 5, 7) images per individual for training, and use the remaining images for testing. For FERET, $p$ (= 3, 4, 5) images per individual are selected for training, and the remaining for testing. All trials are repeated ten times, and then the average recognition results are calculated. Figure 4 shows plots of recognition rate versus dimensionality reduction on three databases. Table 1 lists the best recognition rate for each algorithm. It also provides the optimal values of $k_1$ and $k_2$ for DLA1 and DLA2, which crucial since they have the special sense for building patches.

It is shown that both DLA1 and DLA2 outperform conventional algorithms. DLA2 performs better than DLA1 because weights over part optimizations based *margin degree* are considered to benefit the discriminative subspace selection.

It is worth emphasizing that LDA, SLPP and MFA perform poorly on FERET because face images from FERET are more complex and contain more interference for identification. One method enhance their performance is removing such useless information by using PCA projection retaining appropriate percent energies. We also conduct experiments on FERET by exploring all possible PCA

**Table 1.** Best recognition rates (%) on three databases. For PCA, LDA SLPP, and MFA, the numbers in the parentheses are the subspace dimensions. For DLA1 and DLA2, the first numbers in the parentheses are the subspace dimensions, the second and the third numbers are $k_1$ and $k_2$, respectively. Numbers in the second column denote the number of training samples per subject.

| | | PCA | LDA | SLPP | MFA | DLA1 | DLA2 |
|---|---|---|---|---|---|---|---|
| UMIST | 3 | 71.62(59) | 79.71(18) | 76.58(19) | 82.64(11) | 84.89(18,2,1) | **86.78**(18,2,1) |
| | 5 | 82.88(99) | 88.51(19) | 86.06(19) | 92.61(14) | 93.85(10,3,4) | **95.20**(10,3,4) |
| | 7 | 90.53(135) | 93.31(19) | 91.36(19) | 94.28(19) | 97.01(33,4,5) | **97.45**(33,4,5) |
| YALE | 3 | 52.33(44) | 64.08(14) | 67.00(13) | 64.33(12) | 68.50(18,2,1) | **69.67**(18,2,1) |
| | 5 | 58.33(74) | 72.78(14) | 73.44(14) | 73.44(15) | 78.11(30,3,4) | **79.89**(30,3,4) |
| | 7 | 63.33(36) | 80.80(13) | 82.33(14) | 82.67(15) | 83.83(15,3,5) | **86.50**(15,3,5) |
| FERET | 3 | 41.41(107) | 51.18(38) | 49.55(99) | 55.32(47) | 84.62(24,1,3) | **86.32**(24,1,3) |
| | 4 | 47.00(102) | 53.40(42) | 53.66(99) | 58.27(41) | 91.87(25,3,5) | **93.03**(25,3,5) |
| | 5 | 51.55(87) | 53.60(50) | 54.75(96) | 58.65(62) | 92.85(23,2,5) | **94.33**(23,2,5) |

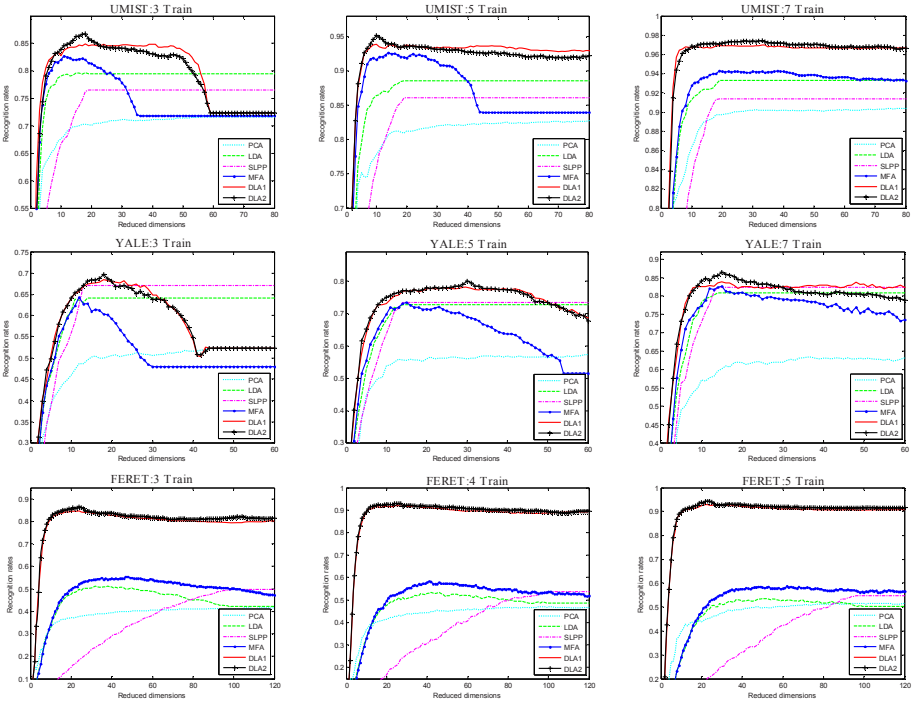


**Fig. 4.** Recognition rate vs. dimensionality reduction on three databases

**Table 2.** Best recognition rates (%) on FERET. The first numbers in the parentheses are the subspace dimensions, the second are the percent of energies retained in the PCA subspace.

| | LDA | SLPP | MFA |
|---|---|---|---|
| 3 | 78.03(17, 96%) | 78.03(17, 96%) | 78.95(21, 95%) |
| 4 | 87.17(15, 96%) | 87.17(15, 96%) | 88.40(15, 94%) |
| 5 | 91.85(21, 96%) | 91.85(21, 96%) | 92.35(19, 95%) |

subspace dimensions and selecting the best one in LDA, SLPP and MFA. As shown in Table 2, although the performances of LDA, SLPP and MFA are significantly improved, DLA1 and DLA2 are still preponderant.

### 4.3 Building Patches

In this subsection, we study effects of $k_1$ and $k_2$ in DLA by setting $t = \infty$ in Eq. (7), based on the UMIST database with $p$ (= 7) samples for each class in the training stage. The reduced dimension in experiments is fixed at 33. By varying $k_1$ from 1 to $p - 1$ (= 6) and $k_2$ from 0 to $N - p$ (= 133) simultaneously, the

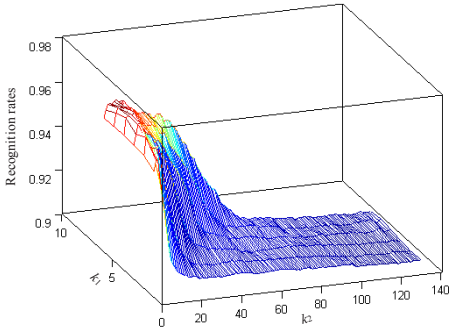


**Fig. 5.** Recognition rate vs. $k_1$ and $k_2$

recognition rate surface can be obtained as shown in Figure 5. In this figure, there is a peak which corresponds to $k_1 = 4$ and $k_2 = 5$.

In this experiment, optimal parameters $k_1$ and $k_2$ for classification can be obtained for patch building. It reveals that the local patch built by neighborhood can characterize not only the intrinsic geometry but also the discriminability better than the global structure.

### 4.4 Semi-supervised Experiments

We compare SDLA and DLA based on the UMIST database by setting $t = \infty$ in Eq. (7). The averaged recognition rates are obtained from ten different random runs. For each turn, $p\ (= 3, 5)$ samples with labels and $q\ (= 3, 5)$ samples without labels for each individual are selected randomly to train SDLA and DLA, and the left ones for each individual are used for testing. It is worth noting that $q$ samples without labels have no effects in training DLA. Table 3 shows unlabeled samples are helpful to improve recognition rates.

**Table 3.** Recognition rates (%) of DLA and SDLA on UMIST. The numbers in the parentheses are the subspace dimensions.

| | | 3 labeled | 5 labeled |
|---|---|---|---|
| 3 unlabeled | DLA | 86.15 (15) | 92.77 (11) |
| | SDLA | **87.69** (13) | **95.42** (22) |
| 5 unlabeled | DLA | 85.78 (27) | 92.53 (11) |
| | SDLA | **88.19** (11) | **95.73** (30) |

### 4.5 Discussions

Based on the experimental results reported in Section 4.2-4.4, we have the following observations:

1. DLA focuses on local patches; implements sample weighting for each part optimization; and avoids the matrix singularity problem. Therefore, it works better than PCA, LDA, SLPP, and MFA;

2. In experiments on building patches, by setting $k_1 = 6$ and $k_2 = 133$, DLA is similar to LDA because the global structure is considered. With this setting, DLA ignores the local geometry and performs poor. Thus, by setting $k_1$ and $k_2$ suitably, DLA can capture both the local geometry and the discriminative information of samples; and
3. Though analyses on SDLA, we can see that, although the unlabeled samples have no discriminative information, they are valuable to improve recognition rates by enhancing the local geometry of all samples.

## 5 Conclusions

In this paper, we have proposed a new linear dimensionality reduction algorithm, termed Discriminative Locality Alignment (DLA). The algorithm focuses on the local patch of every sample in a training set; implements the sample weighting by *margin degree*, a measure of the importance of each sample for classification; and never computes the inverse of a matrix. Advantages of DLA are that it distinguishes the contribution of each sample for discriminative subspace selection; overcomes the nonlinearity of the distribution of samples; preserves discriminative information over local patches; and avoids the matrix singularity problem. Experimental results have demonstrated the effectiveness of DLA by comparing with representative dimensionality reduction algorithms, e.g., PCA, LDA, SLPP, and MFA. An additional contribution is that we have also developed semi-supervised DLA (SDLA), which considers not only the labeled but also the unlabeled samples. Experiments have shown that SDLA performs better than DLA. It is worth emphasizing that the proposed DLA and SDLA algorithms can also be utilized to other interesting applications, e.g., pose estimation [17] emotion recognition [13], and 3D face modeling [12].

## Acknowledgements

The work was partially supported by Hong Kong Research Grants Council General Research Fund (No. 528708), National Science Foundation of China (No. 60675023 and 60703037) and China 863 High Tech. Plan (No. 2007AA01Z164).

## References

1. Belhumeur, P., Hespanha, J., Kriegman, D.: Eigenfaces vs. Fisherfaces: Recognition using Class Specific Linear Projection. IEEE Trans. Pattern Analysis and Machine Intelligence 19(7), 711–720 (1997)
2. Belkin, M., Niyogi, P.: Laplacian Eigenmaps and Spectral Techniques for Embedding and Clustering. Neural Information Processing Systems 14, 585–591
3. Belkin, M., Niyogi, P., Sindhwani, V.: On Manifold Regularization. In: Proc. Int'l Workshop on Artificial Intelligence and Statistics (2005)
4. Cai, D., He, X., Han, J.: Using Graph Model for Face Analysis. Technical report, Computer Science Department, UIUC, UIUCDCS-R-2005-2636 (2005)

5. Chen, L.F., Liao, H.Y., Ko, M.T., Lin, J.C., Yu, G.J.: A New LDA-based Face Recognition System Which Can Solve the Small Sample Size Problem. Pattern Recognition 33(10), 1713–1726 (2000)
6. Fisher, R.A.: The Use of Multiple Measurements in Taxonomic Problems. Annals of Eugenics 7, 179–188 (1936)
7. Graham, D.B., Allinson, N.M.: Characterizing Virtual Eigensignatures for General Purpose Face Recognition. In: Face Recognition: From Theory to Applications. NATO ASI Series F, Computer and Systems Science, vol. 163, pp. 446–456 (2006)
8. He, X., Niyogi, P.: Locality Preserving Projections. In: Advances in Neural Information Processing Systems, vol. 16 (2004)
9. Hotelling, H.: Analysis of A Complex of Statistical Variables into Principal Components. Journal of Educational Psychology 24, 417–441 (1933)
10. Phillips, P.J., Moon, H., Rizvi, S.A., Rauss, P.J.: The FERET Evaluation Methodology for Face-Recognition Algorithms. IEEE Trans. Pattern Analysis and Machine Intelligence 22(10), 1090–1104 (2000)
11. Roweis, S.T., Saul, L.K.: Nonlinear Dimensionality Reduction by Locally Linear Embedding. Science 290, 2323–2326 (2000)
12. Song, M., Dong, Z., Theobalt, C., Wang, H., Liu, Z., Seidel, H.-P.: A Generic Framework for Efficient 2-D and 3-D Facial Expression Analogy. IEEE Trans. Multimedia 9(7), 1384–1395 (2007)
13. Song, M., You, M., Li, N., Chen, C.: A robust multimodal approach for emotion recognition. Neurocomputing 7(10-12), 1913–1920 (2008)
14. Tao, D., Li, X., Wu, X., Maybank, S.: Geometric Mean for Subspace Selection in Multiclass Classification. IEEE Trans. Pattern Analysis and Machine Intelligence 30 (2008)
15. Turk, M., Pentland, A.: Face Recognition using Eigenfaces. In: Proc. IEEE Int'l Conf. Computer Vision and Pattern Recognition, pp. 586–591 (1991)
16. Yan, S., Xu, D., Zhang, B., Zhang, H.J., Yang, Q., Lin, S.: Graph Embedding and Extensions: A General Framework for Dimensionality Reduction. IEEE Trans. Pattern Analysis and Machine Intelligence 29(1), 40–51 (2007)
17. Yan, S., Wang, H., Fu, Y., Yan, J., Tang, X., Huang, T.: Synchronized Submanifold Embedding for Person-Independent Pose Estimation and Beyond. IEEE Trans. Image Processing (2008)
18. Yu, H., Yang, J.: A Direct LDA Algorithm for High-dimensional Data with Application to Face Recognition. Pattern Recognition 34(12), 2067–2070 (2001)
19. Zhang, Z., Zha, H.: Principal Manifolds and Nonlinear Dimension Reduction via Local Tangent Space Alignment. SIAM J. Scientific Computing 26(1), 313–338 (2005)
20. Zhao, D., Lin, Z., Tang, X.: Laplacian PCA and Its Applications. In: Proc. IEEE Int'l Conf. Computer Vision, pp. 1–8 (2007)
21. Zhang, T., Tao, D., Li, X., Yang, J.: A Unifying Framework for Spectral Analysis based Dimensionality Reduction. In: Proc. Int'l J. Conf. Neural Networks (2008)
22. Zhu, X., Ghahramani, Z., Lafferty, J.: Semi-supervised Learning using Gaussian Fields and Harmonic Functions. In: Proc. Int'l Conf. Machine Learning (2003)