Discriminant Locally Linear Embedding With High-Order Tensor Data

Xuelong Li, *Senior Member, IEEE*, Stephen Lin, Shuicheng Yan, *Member, IEEE*, and Dong Xu, *Member, IEEE*

***Abstract*—Graph-embedding along with its linearization and kernelization provides a general framework that unifies most traditional dimensionality reduction algorithms. From this framework, we propose a new manifold learning technique called discriminant locally linear embedding (DLLE), in which the local geometric properties within each class are preserved according to the locally linear embedding (LLE) criterion, and the separability between different classes is enforced by maximizing margins between point pairs on different classes. To deal with the out-of-sample problem in visual recognition with vector input, the linear version of DLLE, i.e., linearization of DLLE (DLLE/L), is directly proposed through the graph-embedding framework. Moreover, we propose its multilinear version, i.e., tensorization of DLLE, for the out-of-sample problem with high-order tensor input. Based on DLLE, a procedure for gait recognition is described. We conduct comprehensive experiments on both gait and face recognition, and observe that: 1) DLLE along its linearization and tensorization outperforms the related versions of linear discriminant analysis, and DLLE/L demonstrates greater effectiveness than the linearization of LLE; 2) algorithms based on tensor representations are generally superior to linear algorithms when dealing with intrinsically high-order data; and 3) for human gait recognition, DLLE/L generally obtains higher accuracy than state-of-the-art gait recognition algorithms on the standard University of South Florida gait database.**

***Index Terms*—Dimensionality reduction, face recognition, human gait recognition, manifold learning, tensor representation.**

I. Introduction

FACE recognition [1], [26], [38] and human gait recognition [3], [9], [30], [34], [46] have been active areas of research because of their potential applications in human–computer interfaces, image and video content analysis, and surveillance. In face recognition and gait recognition, the captured image data often lie in a high-dimensional space. For example, the size of a face image may be 100×100 pixels, and the size of each binary silhouette image in the University of South Florida (USF) HumanID gait database is 128×88 pixels. Variations in a set of high-dimensional data, such as images of a face under different viewing conditions, often have an underlying low-dimensional structure that compactly represents the changes among these observations. For uncovering this low-dimensional structure in image data, dimensionality reduction has been an active research area in pattern recognition. Common linear algorithms for dimensionality reduction include principal components analysis (PCA) [7] which maintains the global Euclidean structure of the data in the high-dimensional space, and linear discriminant analysis (LDA) [7] which aims to preserve discriminative information between data of different classes. To avoid the often unrealistic assumption in PCA and LDA of Gaussian distributed data, nonparametric LDA [7] allows more general data distributions. In addition, methods based on kernel tricks [19], [22] have been employed to handle nonlinearities in data structure by mapping points from the original feature space into another higher dimensional one in which the structure of the data may become linear. PCA, LDA, and their variants have been previously applied to both face recognition [1], [38] and gait recognition [3], [9], [46].

Recent studies [2], [29], [37] assume that data in a complex high-dimensional data space may reside more specifically on or nearly on a low-dimensional manifold embedded within the high-dimensional space. The intent of manifold learning algorithms is to discover this intrinsic manifold structure of the data. For example, ISOMAP [37] determines a low-dimensional representation of the data set that aims to preserve geodesic distances between pairs of data points; locally linear embedding (LLE) [29] is designed to maintain the local linear reconstruction relationship among neighboring points in the low-dimensional space; and Laplacian eigenmap [2] preserves the similarities among neighboring points. These algorithms, though, only define an embedding of the training data points, and do not present a method for mapping new data points that do not exist in the training set, which is the well-known out-of-sample problem. Although there have been some attempts to deal with this out-of-sample problem [4], [10], [11], they deal only with vector-based data and hence often suffer from the curse of dimensionality. Furthermore, these algorithms are unsupervised and assume that the data all reside on a single continuous manifold, which is not the case for multiple-class classification problems. Recent attempts [28], [44] to address multiple classes only preserve the local manifold structure of each class and fail to present good criteria for characterizing the separability of samples from different classes; hence, the derived representations are not necessarily optimal for classification purposes.

These dimensionality reduction methods are based upon different embedding criteria, yet they share underlying commonalities that are captured in a general framework proposed in [49]. In [49], Yan *et al.* demonstrated that most

Manuscript received February 1, 2007; revised July 27, 2007. This paper was recommended by Associate Editor V. Murino.

X. Li is with the School of Computer Science and Information Systems, Birkbeck College, University of London, London WC1E 7HX, U.K.

S. Lin is with Microsoft Research Asia, Beijing 100080, China.

S. Yan is with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore 117576.

D. Xu is with the School of Computer Engineering, Nanyang Technological University, Singapore 639798.

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/TSMCB.2007.911536

dimensionality reduction algorithms can be unified within a proposed graph-embedding framework in which the desired statistical or geometric data properties of a given algorithm are encoded as graph relationships. Under the unified perspective of graph-embedding, operations such as linearization and kernelization can be generally applied to all embedding techniques encompassed in this framework to address the out-of-sample problem for vector input data.

Recently, it has become popular to utilize higher order tensor decomposition [13], [15], [16] for dimensionality reduction [6], [12], [31], [45], [48], [50], [53], [54], as pioneered by the work on Tensorfaces [40]. Based on image object representation, these algorithms can be roughly classified into two categories, referred to here as *image-as-vector* and *image-as-matrix*. An example of an *image-as-vector* approach is the Tensorface technique and its extensions [40]–[43]. Tensorface represents a 2-D gray-level image as a 1-D vector and organizes the image ensembles of different persons under different pose, illumination, and expression into a high-order tensor. The work in [42] applied multilinear analysis to synthesize new motions and recognize people and actions from new motions. The work in [43] is the only work to address higher order statistics within the context of higher order tensor decomposition. Other works [6], [12], [14], [27], [31]–[33], [35], [36], [48], [50], [52]–[54] employ an *image-as-matrix* representation, in which a 2-D gray-level image is directly represented as a second-order data tensor (matrix), or is filtered by Gabor functions [8], [39] to obtain a set of images that comprise a third-order data tensor.[1] Although *image-as-vector* and *image-as-matrix* techniques both apply high-order decomposition to more effectively utilize relationships among the data, their motivations are different. Tensorface aims to utilize relationships among *external* factors such as pose, illumination and expression, whereas *image-as-matrix* methods aim to characterize the variations among *internal* factors of an image object, such as image rows, columns, and Gabor features. In this paper, we adopt the *image-as-matrix* representation because it can utilize correlations among pixels within a row/column and provide a way to extend discriminant locally linear embedding (DLLE) to handle tensor data such as Gabor-filtered images. Another reason is that several different groups [6], [12], [32], [48], [50], [53] have experimentally reported that the image-as-matrix representation may lead to good classification performance for face recognition, digital number recognition, and image classification, particularly when the number of training samples is small. A possible explanation is that the smaller number of data entries along each data dimension facilitates subspace learning with little training data. We also notice that an image-as-matrix representation can be extended to effectively incorporate both internal and external factors by constructing tensors with several dimensions representing rows, columns, Gabor features, etc., and other dimensions encoding additional factors such as pose, illumination, and expression.

Previous image-as-matrix techniques have been developed for different purposes. For classifier design, Tao *et al.* [32] extended quadratic programming-based support vector machines and second-order cone programming-based minimax probability machines to their related tensor-based versions. Several works [31], [45], [48], [50], [53], [54] applied PCA or LDA criteria for dimensionality reduction but do not consider local manifold structure. To deal with local manifold structure and gain the advantages of tensor representations in manifold learning, He *et al.* [12] extended their proposed locality preserving projections (LPP) algorithm to handle second-order data tensors (matrices). This technique was further extended by Dai and Yeung [6] to tensor LPP that handles general tensor input. Dai and Yeung additionally extended neighborhood preserving embedding (NPE) [11], which is the direct linearization of LLE (LLE/L) [29], and also local discriminant embedding (LDE) [5], which is a variant of classical nonparametric LDA [7], to tensor NPE and tensor LDE, respectively.

In this paper, we formulate a new dimensionality reduction method called DLLE that enhances the visual recognition performance of LLE by utilizing both label information and local manifold structure. DLLE preserves manifold properties within each class according to the LLE criterion, and separability between different classes is enforced by maximizing margins between point pairs on different manifolds/classes. Within the graph-embedding framework, linearization of DLLE (DLLE/L) can be easily obtained. Moreover, to further enhance visual recognition, we propose a multilinear extension, i.e., the tensorization of DLLE (DLLE/T), for handling the out-of-sample problem for high-order tensor input instead of just conventional vector input. We also provide a detailed procedure on how to apply DLLE to gait recognition applications. Finally, we present a systematic evaluation of DLLE along its linearization and tensorization on face recognition and gait recognition. In this paper, we only focus on the linear and multilinear versions of DLLE; a kernel version could be obtained using the graph-embedding framework, and recent studies [51], [55] have shown that it is possible to conduct nonlinear mapping of high-order tensor input with the kernel trick. Compared with the methods in [6] and [12], DLLE is different in the following aspects: 1) DLLE aims to preserve the local manifold structure within each class in a manner similar to LLE and 2) DLLE also aims to separate different submanifolds (classes) by enlarging the distances between the margin samples.

The rest of this paper is organized as follows. We briefly review the graph-embedding framework in Section II. Then, based on this framework, we propose the linear version of DLLE, which is denoted as DLLE/L. We subsequently extend it in Section III to its multilinear version DLLE/T to handle the out-of-sample problem for high-order tensor input. In Section IV, we present a detailed procedure for applying DLLE to gait recognition. A comprehensive set of comparison experiments on face recognition and gait recognition are given in Section V, followed by concluding remarks in Section VI.

II. Graph-Embedding Framework

We briefly review the graph-embedding framework introduced in [49]. Denote the sample set as $X = [x_1, x_2, \ldots, x_N]$, $x_i \in \mathbb{R}^m$, with N indicating the total number of samples. Let

[1]Note that here the tensor is used in a context different from the mathematical definition of a tensor.

$G = \{X, S\}$ be an undirected similarity graph, which is called an intrinsic graph, with vertex set X and similarity matrix $S \in \mathbb{R}^{N\times N}$. The corresponding diagonal matrix D and the Laplacian matrix L of the graph G are defined as

$$L = D - S \qquad D_{ii} = \sum_{j\neq i} S_{ij} \; \forall i. \tag{1}$$

The task of graph-embedding is to determine a low-dimensional representation $Y = [y_1, y_2, \ldots, y_N]$ of the vertex set X, where the column vector y_i is the embedding for the vertex x_i, that preserves the similarities between pairs of data points in the original high-dimensional space. Direct graph-embedding aims to maintain similarities among vertex pairs according to the graph preserving criterion [49] as follows:

$$\begin{aligned} Y^* &= \arg \min_{\mathrm{Tr}(YBY^T)=c} \sum_{i\neq j} \|y_i - y_j\|^2 S_{ij} \\ &= \arg \min_{\mathrm{Tr}(YBY^T)=c} \mathrm{Tr}(YLY^T) \end{aligned} \tag{2}$$

where c is a constant, $\mathrm{Tr}(\cdot)$ is the trace of an arbitrary square matrix, and B is the constraint matrix. B may simply be a diagonal matrix used for scale normalization or may express more general constraints among vertices in a penalty graph G' that describes similarities between nodes that are unfavorable and should be avoided, i.e., $B = L^p = D^p - S^p$, where S^p is the similarity matrix of graph G', and B or L^p is the Laplacian matrix of G', which is defined similarly to (1).

The similarity preservation property of the graph preserving criterion works in two ways. If the similarity between samples x_i and x_j is greater (positive), then the distance between y_i and y_j should be smaller to minimize (2). On the other hand, if the similarity between x_i and x_j is lower (negative), the distance between y_i and y_j should instead be larger. Hence, the similarities and differences among vertex pairs in a graph G are preserved in the embedding.

As shown in the embedding framework [49], (2) can be resolved by converting it into the following ratio formulation:

$$Y^* = \arg\min_{Y} \frac{\mathrm{Tr}(YLY^T)}{\mathrm{Tr}(YBY^T)}. \tag{3}$$

If the constraint matrix represents only scale normalization, this ratio formulation can be directly solved by eigenvalue decomposition. For a more general constraint matrix, it can be approximately solved with generalized eigenvalue decomposition by transforming the objective function into a more tractable approximate form $\max_Y \mathrm{Tr}((YLY^T)^{-1}(YBY^T))$.

Although direct graph-embedding computes a low-dimensional representation of the vertices in X, it does not determine how new out-of-sample high-dimensional data can be mapped to the low-dimensional space. For this, linearization and kernelization, as described in [49], are needed. Taking linearization as an example, if the low-dimensional vector representations of the vertices can be obtained from linear projections as $y_i = W^T x_i$, where W is the projection matrix that we want to solve for mapping out-of-sample data, then the objective function (2) is changed to

$$W^* = \arg \min_{\mathrm{Tr}(W^T XBX^T W)=c} \mathrm{Tr}(W^T XLX^T W). \tag{4}$$

III. DLLE

We formulate a supervised manifold learning technique by extending the unsupervised LLE algorithm to the proposed DLLE. By following the graph-embedding framework, two graphs are formed: 1) the intrinsic graph G consists of a set of subgraphs that preserve LLE properties within each class and 2) the penalty graph G' connects margin vertices of different classes.

A. Construction of Intrinsic and Penalty Graphs

In constructing the intrinsic and penalty graphs of DLLE, we start with the partition of data elements into different classes. The intrinsic graph G is formed by first placing edges only between samples of the same class. The local reconstruction coefficient matrix M_{ij} can be acquired by minimizing the objective function as

$$\min \sum_{j\in N_{k1}(i)} \|x_i - M_{ij}x_j\|^2 \qquad \sum_{j\in N_{k1}(i)} M_{ij} = 1 \; \forall i \tag{5}$$

where the set $N_{k1}(i)$ contains the indices of the k_1 nearest neighbors of sample x_i in the same class. The difference from the original unsupervised LLE is that each sample is reconstructed only from other samples in its class. To incorporate the local reconstruction coefficients in the intrinsic graph, its similarity matrix is set according to

$$S_{ij} = \begin{cases} (M + M^T - M^T M)_{ij}, & \text{if } i \neq j \\ 0, & \text{otherwise} \end{cases} \tag{6}$$

as described in [49].

For the penalty graph G', its similarity matrix S^p_{ij} represents geometric or statistical properties to be avoided and is used as the constraint matrix in the graph-embedding framework. In DLLE, the penalty graph G' is constructed to promote separability of different classes. More strictly speaking, for each class indexed as l, we first compute the distances from samples in class l to samples in other classes, and then for the k_2 smallest distances, we connect the corresponding pairs of points $\{(i, j) : l_i = l, l_j \neq l\}$ with weights $S^p_{ij} = 1/k_2$ and $S^p_{ji} = 1/k_2$, where l_i is the class label for the ith sample.

In the following, we describe how to utilize the graph-embedding framework to develop algorithms for supervised manifold learning based on the design of intrinsic and penalty graphs. Unlike the formulation (3) of the original graph-embedding framework, DLLE instead optimizes the objective function in difference form, i.e.,[2]

$$Y^* = \arg\max_{Y} \mathrm{Tr}(YBY^T) - h\mathrm{Tr}(YLY^T). \tag{7}$$

[2]As shown in [7], if we want to maximize one term and at the same time minimize another term, we can use the ratio formulation as in (3) or the difference form in (7).

This objective function for minimizing $\mathrm{Tr}(YLY^T)$ while maximizing $\mathrm{Tr}(YBY^T)$ clearly has a direct solution that can be computed using the eigenvalue decomposition method. The coefficient h is used to balance the two terms of the objective function. For the elements S_{ij} of the similarity matrices, we normalize M_{ij} in (5), and also normalize values of S^p_{ij} by setting $S^p_{ij} = 1/k_2$. From normalization, the two terms in (7) are usually balanced, i.e., we can set $h = 1$ in real applications. The factor h, however, can be adjusted if balancing is needed. We will explain the benefits of expressing the objective function in difference form in Section III-C.

B. Linearization Formulation

With the two graphs G and G' in the graph-embedding framework and the objective function in (7), DLLE/L can be performed using Procedure 1.

Procedure 1: DLLE/L

1) **PCA projection:** The training samples are projected into a PCA subspace that retains a certain level of energy. Let W_{PCA} denote the transformation matrix of PCA.
2) **Graph construction:** The similarity matrix S of the intrinsic graph and S^p of the penalty graph are constructed according to the LLE manifold property within each class and separability criteria between classes.
3) **Supervised manifold learning:** To simultaneously maximize class separability and preserve the intraclass manifold property, the optimal projection matrix W^* is acquired by optimizing the following objective function:

$$W^* = \arg\max_W \mathrm{Tr}(W^T X L^p X^T W - hW^T X L X^T W) \quad (8)$$

with $L^p = D^p - S^p$ and $L = D - S$. The optimal solution of (8) can be obtained by the eigenvalue decomposition method.
4) **Output** $W = W_{\mathrm{PCA}} W^*$.

We note that maximum margin criterion (MMC) [18] presents an objective function with a similar difference form. However, MMC considers only the global Euclidean structure of the data space and ignores any nonlinear manifold structure.

C. Tensorization Formulation

Although the linearization and kernelization of the graph-embedding framework [49] give a mapping for out-of-sample data, they both assume that the data are represented as vectors. Recent studies [6], [12], [48], [50], [53], [54] have experimentally reported that dimensionality reduction algorithms with data encoded as high-order tensors usually outperform those with data represented as vectors, particularly when the number of training samples is small. The possible explanation is that the smaller number of data entries along each data dimension facilitates subspace learning from little training data. To describe the DLLE/T, we begin by defining a few tensor operations.

Definition 1—(Tensor Inner Product, Norm, and Distance): The inner product of two tensors $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_n}$ and $\mathbf{Y} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_n}$ is defined as $\langle \mathbf{X}, \mathbf{Y} \rangle = \sum_{i_1=1,\ldots,i_n=1}^{m_1,\ldots,m_n} \mathbf{X}_{i_1,\ldots,i_n} \mathbf{Y}_{i_1,\ldots,i_n}$; consequently, the norm of a tensor $\mathbf{X}$ is defined to be $\|\mathbf{X}\| = \sqrt{\langle \mathbf{X}, \mathbf{X} \rangle}$; and the tensor distance between tensors $\mathbf{X}$ and $\mathbf{Y}$ is defined as $D(\mathbf{X}, \mathbf{Y}) = \|\mathbf{X} - \mathbf{Y}\|$.

Definition 2—(Mode-k Product): The mode-k product of tensor $\mathbf{X}$ with matrix $U \in \mathbb{R}^{m'_k \times m_k}$ is defined as $\mathbf{Y} = \mathbf{X} \times_k U$, where $\mathbf{Y}_{i_1,\ldots,i_{k-1},i,i_{k+1},\ldots,i_n} = \sum_{j=1}^{m_k} \mathbf{X}_{i_1,\ldots,i_{k-1},j,i_{k+1},\ldots,i_n} U_{i,j}$, $i = 1, \ldots, m'_k$.

Definition 3—(Mode-k Flattening): The mode-k flattening of the nth-order tensor $\mathbf{X} \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_n}$ into a matrix $X^k \in \mathbb{R}^{m_k \times \prod_{i \neq k} m_i}$, i.e., $X^k \Longleftarrow_k \mathbf{X}$, is defined as $X^k_{i_k, j} = \mathbf{X}_{i_1,\ldots,i_n}$, $j = 1 + \sum_{l=1, l\neq k}^{n} (i_l - 1) \prod_{o=l+1, o\neq k}^{n} m_o$.

We denote a training set in tensor form as $\{\mathbf{X}_i \in \mathbb{R}^{m_1 \times m_2 \times \cdots \times m_n}, i = 1, \ldots, N\}$. Similar to DLLE/L, we express the low-dimensional representation of a vertex as projections of the original tensor with multiple projection matrices $\mathbf{Y}_i = \mathbf{X}_i \times_1 U^1 \times_2 U^2 \cdots \times_n U^n$ to tensors of smaller size, where $U^k \in \mathbb{R}^{m'_k \times m_k}$, $\mathbf{Y}_i \in \mathbb{R}^{m'_1 \times m'_2 \times \cdots \times m'_n}$, and m'_k denotes the lower dimensionalities after projection. Then, the objective function in (2) is converted to

$$(U^1, \ldots, U^n)^* = \arg \min_{f(U^1,\ldots,U^n)=c} \times \sum_{i \neq j} \left\| (\mathbf{X}_i - \mathbf{X}_j) \times_1 U^1 \times_2 U^2 \cdots \times_n U^n \right\|^2 S_{ij} \quad (9)$$

where S_{ij} is entry of the similarity matrix S of the intrinsic graph G, which characterizes the desired statistical or geometric properties of the dimensionality-reduced data set. For DLLE, the constraints among graph vertices are expressed by the Laplacian matrix $B = L^p = D^p - S^p$, and we have

$$f(U^1, \ldots, U^n) = \sum_{i \neq j} \left\| (\mathbf{X}_i - \mathbf{X}_j) \times_1 U^1 \times_2 U^2 \cdots \times_n U^n \right\|^2 S^p_{ij} \quad (10)$$

where S^p_{ij} is the entry of the similarity matrix of the penalty graph G'.

Generally, no closed-form optimization solution exists for (9). To address this problem, we first initialize the projection matrices $U^1, \ldots, U^n$. Each projection matrix U^k can then be solved from the other projection matrices $U^1, \ldots, U^{k-1}$, $U^{k+1}, \ldots, U^n$ by

$$U^{k^*} = \arg \min_{f(U^k)=c} \sum_{i \neq j} \left\| U^k X^k_i - U^k X^k_j \right\|^2 S_{ij} \quad (11)$$

where $\mathbf{X}^k_i = \mathbf{X}_i \times_1 U^1 \cdots \times_{k-1} U^{k-1} \times_{k+1} U^{k+1} \cdots \times_n U^n$, and X^k_i is the mode-k flattened matrix of the tensor $\mathbf{X}^k_i$. As shown within the original embedding framework [49], this objective function is approximately solved by the generalized eigenvalue decomposition method. Therefore, the objective function of (9) can be solved by iteratively optimizing different projection matrices while fixing the other projection matrices.

If we utilize the ratio formulation in (3) to solve for the objective function (9), we have observed, as shown in Section V-B, that this iterative procedure suffers from

convergence problems. This likely results from the approximation of the objective function $\min_Y \mathrm{Tr}(YLY^T)/\mathrm{Tr}(YBY^T)$ by $\max_Y \mathrm{Tr}((YLY^T)^{-1}(YBY^T))$, as mentioned in Section II. To overcome this problem, we optimize (9) with the objective function expressed in difference form as follows:

$$(U^1,\ldots,U^n)^* = \arg\max_{U^1,\ldots,U^n} \times \sum_{i\neq j} \left\|(\mathbf{X}_i-\mathbf{X}_j)\times_1 U^1\times_2 U^2\cdots\times_n U^n\right\|^2 \left(S_{ij}^p - hS_{ij}\right). \tag{12}$$

Assume that only the projection matrix U^k is unknown, then with the definitions of X_i^k and X_j^k, we have

$$\begin{aligned} U^{k^*} &= \arg\max_{U^k} \sum_{i\neq j} \left\|U^k\left(X_i^k - X_j^k\right)\right\|^2 \left(S_{ij}^p - hS_{ij}\right) \\ &= \arg\max_{U^k} \mathrm{Tr}\left[U^k \sum_{i\neq j}\left(X_i^k - X_j^k\right)\left(X_i^k - X_j^k\right)^T \times \left(S_{ij}^p - hS_{ij}\right)(U^k)^T\right] \\ &= \arg\max_{U^k} \mathrm{Tr}\left[U^k C^k (U^k)^T\right] \end{aligned} \tag{13}$$

where $C^k = \sum_{i\neq j}(X_i^k - X_j^k)(X_i^k - X_j^k)^T(S_{ij}^p - hS_{ij})$ is a symmetric matrix. Note that (13) is derived from the fact that for an arbitrary matrix A, we have $\|A\| = \mathrm{Tr}(AA^T) = \mathrm{Tr}(A^TA)$. The optimal projection matrix is composed of the leading eigenvectors of C^k.

Procedure 2: DLLE/L

1) Initialize $U_0^1 = I_{m_1}, U_0^2 = I_{m_2},\ldots,U_0^n = I_{m_n}$ or to arbitrary column orthogonal matrices
2) For $t = 1,2,\ldots,T_{\max}$ Do
   a) For $k = 1,2,\ldots,n$ Do

$$\mathbf{X}_i^k = \mathbf{X}_i \times_1 U_t^1\cdots\times_{k-1}U_t^{k-1}\times_{k+1}U_{t-1}^{k+1}\cdots\times_n U_{t-1}^n$$

$$X_i^k \Longleftarrow_k \mathbf{X}_i^k$$

$$C^k = \sum_{i\neq j}\left(X_i^k - X_j^k\right)\left(X_i^k - X_j^k\right)^T\left(S_{ij}^p - hS_{ij}\right)$$

$$C^k\left(U_t^k\right)^T = \left(U_t^k\right)^T\Lambda_k, \qquad U_t^k \in \mathbb{R}^{m_k'\times m_k}$$

   b) If $t > 2$ and $\mathrm{Tr}[U_t^k(U_{t-1}^k)^T] > (1-\epsilon)m_k'$, $k = 1,\ldots,n$, break;
   c) Output the projections $U^k = U_t^k \in \mathbb{R}^{m_k'\times m_k}$, $k = 1,\ldots,n$.

In Procedure 2, we describe in detail the procedure for learning the projection matrices in an iterative manner. Note that $T_{\max}$ signifies a predefined maximum number of iterations for Procedure 2. In the following, we analyze the computational complexity of DLLE/T, and we prove in Theorem 1 the convergence of this tensorization. The convergence property is substantiated in practice by our presented experimental results in Section V-B.

Computational Complexity Analysis: For ease of understanding, let us assume that the sample tensor is of uniform size in each dimension, i.e., $m_i = m$. In DLLE/L, eigenvalue decomposition has a complexity of $O(m^{(3n)})$. However, in DLLE/T, the complexity in computing the covariance matrices is $O(n * m^{(n+1)})$, and the complexity for eigenvalue decomposition is $O(n * m^3)$ for each loop, which is much lower than DLLE/L. Although DLLE/T has no closed-form solution and many loops are required for optimization, it is still much faster than DLLE/L due to its simplicity in each iteration loop.

Theorem 1: The iterative procedure for DLLE/T will converge to a local optimum.

Proof: Following the proof in [17], we need only to prove that the objective function is nondecreasing and has an upper bound. We rewrite the objective function in (12) as follows:

$$g\left(U_t^1,U_t^2,\ldots,U_t^n\right) = \sum_{i\neq j}\left\|(\mathbf{X}_i-\mathbf{X}_j)\times_1 U_t^1\times_2 U_t^2\cdots\times_n U_t^n\right\|^2\left(S_{ij}^p - hS_{ij}\right).$$

In each step of Procedure 2, the objective function in (12) is nondecreasing, i.e.,

$$\begin{aligned} g\left(U_t^1,U_t^2,\ldots,U_t^n\right) &\leqslant g\left(U_{t+1}^1,U_t^2,\ldots,U_t^n\right) \\ &\leqslant g\left(U_{t+1}^1,U_{t+1}^2,\ldots,U_t^n\right) \\ &\leqslant \cdots \leqslant g\left(U_{t+1}^1,U_{t+1}^2,\ldots,U_{t+1}^n\right). \end{aligned}$$

Moreover, there exists an upper bound for the objective function in (12), i.e.,

$$g\left(U_t^1,U_t^2,\ldots,U_t^n\right) \leqslant \sum_{i\neq j}\|\mathbf{X}_i-\mathbf{X}_j\|^2\left|S_{ij}^p - hS_{ij}\right|.$$

Therefore, the objective function will converge to a local optimum. ■

Remark: In 2-D LDA (2DLDA) [54] and discriminant analysis with tensor representation (DATER) [50], the objective function is as follows:

$$(U^1,U^2,\ldots,U^n) = \arg\max_{U^k\|_{k=1}^n} \frac{\sum_l n_l\left\|(\overline{\mathbf{X}}_l - \overline{\mathbf{X}})\times_1 U^1\times_2 U^2\cdots\times_n U^n\right\|^2}{\sum_i\left\|(\mathbf{X}_i - \overline{\mathbf{X}}_{l_i})\times_1 U^1\times_2 U^2\cdots\times_n U^n\right\|^2}$$

where $\overline{\mathbf{X}}$ is the average tensor of all the samples, $\overline{\mathbf{X}}_l$ is the average tensor of the samples belonging to class l, and n_l is the sample number of class l. The authors also propose to use iterative algorithms in which the optimal U^k is computed from the given $U^1,U^2,\ldots,U^{k-1},U^{k+1},\ldots,U^n$ as follows:

$$U^k = \arg\max_{U^k}\frac{\mathrm{Tr}\left[U^kS_b^k(U^k)^T\right]}{\mathrm{Tr}\left[U^kS_w^k(U^k)^T\right]} \tag{14}$$

where $S_b^k = \sum_l n_l(\overline{X}_l^k - \overline{X}^k)(\overline{X}_l^k - \overline{X}^k)^T$ and $S_w^k = \sum_i(X_i^k - \overline{X}_{l_i}^k)(X_i^k - \overline{X}_{l_i}^k)^T$, and X_i^k, X_l^k, and $\overline{X}^k$ are the

corresponding mode-k flattened matrices of tensors $\mathbf{X}_i^k$, $\overline{\mathbf{X}}_l^k$, and $\overline{\mathbf{X}}^k$.[3]

As mentioned in Section II, it is difficult to directly optimize a ratio formulation such as (3) and (14). Therefore, 2DLDA and DATER alter the objective function to a more tractable form $\arg\max_{U^k} \mathrm{Tr}[(U^k S_w^k (U^k)^T)^{-1}(U^k S_b^k (U^k)^T)]$ for which generalized eigenvalue decomposition can be applied. However, this alteration of the objective function will not guarantee that the objective function is nondecreasing as in DLLE/T. Consequently, 2DLDA and DATER suffer from convergence problems, as experimentally demonstrated in Section V-B. This highlights the importance of utilizing a difference formulation, as in (7).

IV. Application to Gait Recognition

In this section, we describe a detailed procedure for applying DLLE to gait recognition. In the literature, Sarkar *et al.* [30] proposed a baseline approach for extracting binary human silhouettes, computing the gait period for each video sequence, and recognizing people by direct matching. To facilitate subsequent research in this field, they have made the extracted binary silhouette data and gait period lengths publicly available [30]. In [20], gait recognition was conducted based on the similarity of gray-level average silhouettes over one gait cycle, which demonstrates performance that is comparable to [30] but with much faster computation speed. More recently, Han and Bhanu [9] proposed to use a two-stage PCA + LDA on a gait energy image that is similar to the gray-level average silhouettes in [20]. The efficiency of the PCA + LDA strategy has been demonstrated in face recognition [1].

In this paper, we focus on the learning algorithm for gait recognition and start our analysis from binary image sequences. As in [9] and [20], each full sequence is partitioned into several subsequences according to the gait period length N_{gait} provided by Sarkar *et al.* [30] in the USF HumanID database. The binary silhouette images within each gait cycle of a sequence are then averaged to acquire several gray-level average silhouette images as follows:

$$\mathrm{AS}_i = \frac{1}{N_{\mathrm{gait}}} \sum_{k=(i-1)N_{\mathrm{gait}}+1}^{k=iN_{\mathrm{gait}}} S_k, \quad i=1\ldots,\lfloor T/N_{\mathrm{gait}} \rfloor \qquad (15)$$

where $S = \{S(1), \ldots, S(T)\}$ denotes the set of T binary images that comprise one sequence. For the USF HumanID database, the training set is comprised of all the gray-level average silhouette images from all the sequences in the gallery set; the test set is comprised of the average silhouette images from all the sequences in the related probe sets.

Our use of the average silhouettes is based on the following observations [9], [20]: 1) it is more robust than the time-series-based silhouette representation [30], particularly since binary images from current object segmentation algorithms are quite noisy and 2) the average silhouette images do not depend on the choice of the starting stance of the gait cycle, thus the gait alignment in [30] is unnecessary. Fig. 7 shows some original binary images and the average silhouettes of three different persons in the gallery set. The average silhouettes are gray-level silhouette images, with darker regions more likely to correspond to the pedestrian. From these average silhouettes, people can be distinguished from one another.

In the testing stage, after dimensionality reduction for both the gallery and probe sets, we use the same distance measure between the gallery sequence and probe sequence, as in [20] and [30], with the median operation for more robustness to noise, as follows:

$$\mathrm{Dist}(\mathrm{LS}_P, \mathrm{LS}_G) = \mathrm{median}_{i=1}^{N_P} \left(\min_{j=1}^{N_G} \|\mathrm{LS}_P(i) - \mathrm{LS}_G(j)\|^2 \right) \qquad (16)$$

where $\mathrm{LS}_P(i)$, $i = 1, \ldots, N_P$ and $\mathrm{LS}_G(j)$, $j = 1, \ldots, N_G$ are the lower dimensional silhouette image matrices of one person from the gallery sequence and the probe sequence, respectively, N_P and N_G are the total number of images, and $\mathrm{Dist}(\mathrm{LS}_P, \mathrm{LS}_G)$ is the distance between these two sequences. Based on this distance, the nearest neighbor classifier is used for recognition.

TABLE I
Top-1 Recognition Accuracy (in Percentages) Comparison on FERET Database of DLLE/L, LDA/L, and LLE/L. For LDA/L and LLE/L, the Number in Parentheses Is the Final Dimension After Dimensionality Reduction. For DLLE/L, the Parenthesized Values Are k_1, k_2, and the Reduced Dimension

Algorithm	FERET
LDA/L	83.5 (33)
LLE/L	58.3 (319)
DLLE/L	***91.0*** (3, 200, 128)

TABLE II
Top-1 Recognition Accuracy (in Percentages) Comparison on ORL and AR Databases. For the LDA and LLE Variants, the Number in Parentheses Is the Final Dimension After Dimensionality Reduction. For the DLLE Variants, the Parenthesized Values Are k_1, k_2, and the Reduced Dimension

Algorithm	ORL	AR
LDA/L	84.6 (37)	86.1 (23)
LLE/L	85.8 (113)	84.2 (82)
DLLE/L	***90.4*** (3, 300, 113)	***87.2*** (3, 100, 82)
LDA/2D	93.3 (8x6)	78.1 (14x14)
DLLE/2D	***94.2*** (3, 100, 6x26)	***84.7*** (2, 100, 26x11)
LDA/3D	97.1 (12x5x16)	90.8 (25x7x19)
DLLE/3D	***98.3*** (2, 100, 13x2x14)	***93.9*** (3, 100, 30x10x35)

V. Experiments

We present a set of experiments to evaluate the proposed dimensionality reduction algorithms for recognition tasks. First, we compare the linearization version of DLLE with LDA on the FERET database [24], and then we compare the linearization and tensorization versions of DLLE with those of LDA on

[3]Note that $\mathbf{X}_i^k$, $\overline{\mathbf{X}}_l^k$, and $\overline{\mathbf{X}}^k$ are defined as: $\mathbf{X}_i^k = \mathbf{X}_i \times_1 U^1 \cdots \times_{k-1} U^{k-1} \times_{k+1} U^{k+1} \cdots \times_n U^n$, $\overline{\mathbf{X}}_l^k = \overline{\mathbf{X}}_l \times_1 U^1 \cdots \times_{k-1} U^{k-1} \times_{k+1} U^{k+1} \cdots \times_n U^n$, $\overline{\mathbf{X}}^k = \overline{\mathbf{X}} \times_1 U^1 \cdots \times_{k-1} U^{k-1} \times_{k+1} U^{k+1} \cdots \times_n U^n$.

TABLE III
USF HUMANID GAIT RECOGNITION EXPERIMENTS [30] (V—VIEW, S—SHOE, U—SURFACE, B—BRIEFCASE, T—TIME)

Probe Set	A	B	C	D	E	F	G	H	I	J	K	L
Size of Probe Set	122	54	54	121	60	121	60	120	60	120	33	33
Gallery/Probe Difference	V	S	VS	U	SU	SV	SUV	B	SB	VB	T	TU

TABLE IV
TOP-1 AND TOP-5 RECOGNITION ACCURACY (IN PERCENTAGES) COMPARISON OF BASELINE [30], WORK OF HAN AND BHANU [9], AND DLLE/L ON THE USF HUMANID GAIT DATABASE. FOR DLLE/L, THE PARENTHESIZED VALUES ARE k_1, k_2, AND THE REDUCED DIMENSION

Task	Top-1 Performance			Top-5 Performance		
Probe Set	Baseline [30]	Han and Bhanu [9]	DLLE/L	Baseline [30]	Han and Bhanu [9]	DLLE/L
A	73	89	***90*** (3, 100, 202)	88	93	***95*** (3, 100, 14)
B	78	87	***89*** (3, 100, 206)	93	93	***96*** (3, 100, 8)
C	48	78	***81*** (3, 100, 230)	78	89	***93*** (3, 100, 26)
D	32	36	***40*** (2, 200, 232)	66	65	***74*** (2, 100, 160)
E	22	38	***50*** (2, 200, 226)	55	60	***78*** (2, 400, 234)
F	17	20	***27*** (2, 200, 184)	42	42	***50*** (3, 100, 196)
G	17	28	***26*** (3, 100, 232)	38	45	***53*** (3, 100, 212)
H	61	62	***65*** (3, 300, 226)	85	88	***90*** (3, 100, 110)
I	57	59	***67*** (2, 400, 226)	78	79	***90*** (2, 300, 198)
J	36	59	***57*** (3, 100, 202)	62	80	***83*** (3, 200, 234)
K	3	3	***12*** (3, 100, 8)	12	6	***33*** (3, 100, 12)
L	3	6	***18*** (3, 100, 108)	15	9	***27*** (3, 100, 232)

Fig. 1. Some sample images from (top row) the FERET, (middle row) ORL, and (bottom row) AR face databases. For the FERET database, we refer to the images from left to right as ba, bf, bg, bh, be, bd, bc [25]. For the ORL database, the labels of the image are 1 to 10 from left to the right [23]. For the AR database, the first four images are labeled from 1 to 4, and the last four images are labeled from 14 to 17 [21].

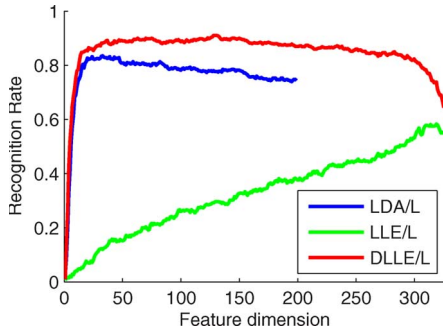


Fig. 2. Top-1 recognition rates (in percentages) with different feature dimensions on the FERET database.

the ORL [23] and AR [21] face databases. In addition, we report results from the LLE/L [11] for comparison. Second, we compare DLLE/L with recently proposed gait recognition algorithms [9], [30] on the standard USF HumanID database [30]. In all the experiments, the final classification after dimensionality reduction is determined by the nearest neighbor criterion. In Tables I–IV, *Top-1* denotes that the correct subject is ranked as the top candidate, and *Top-5* indicates that the correct subject is ranked among the top five candidates.

A. Face Recognition on FERET Database: Linearization

The proposed algorithm was tested on a subset of the FERET database [24], [25]. This subset includes 1400 images of 200 individuals (each with seven images); the images of one person are shown in Fig. 1. All the images were cropped and resized to 32×32 pixels based on the location of the two eyes, and were preprocessed by histogram equalization. Using the image labels in [25], we choose in this experiment four images per person from bf, bg, be, and bd for training, and use the remaining three images from ba, bh, and bc for testing. We focus on comparing DLLE/L, the linearization of LDA (LDA/L), and LLE/L. Table I shows the experimental results of different algorithms, and Fig. 2 shows a plot of recognition rates with respect to different feature dimensionalities. The results demonstrate the greater effectiveness of DLLE/L beyond LDA/L and LLE/L. In Fig. 2, we also note that the maximum dimension of LDA/L is constrained to be $N_c - 1$ [1], where N_c is the total number of classes, whereas LLE/L and DLLE/L do not have such constraints.

B. Face Recognition on ORL [23] and AR [21] Face Databases: Linearization and Tensorization

In this experiment, we analyze data represented as first-order tensors (vectors), second-order tensors (matrices), and

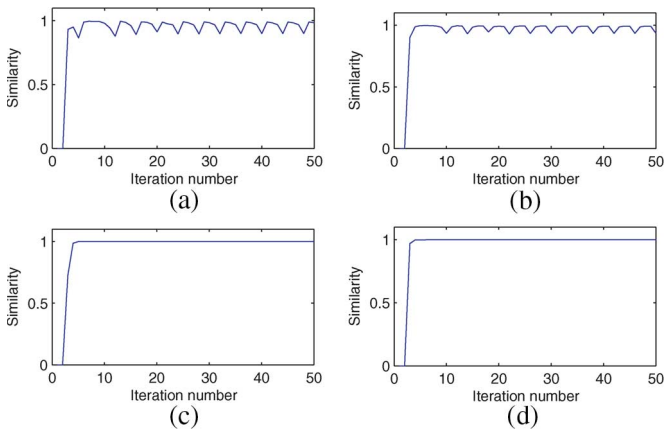


Fig. 3. Convergence characteristics of 2DLDA and DLLE/2D. The vertical axis indicates the similarity of two adjacent projection matrices, and the horizontal axis gives the number of iterations. (a) U^1 of 2DLDA. (b) U^2 of 2DLDA. (c) U^1 of DLLE/2D. (d) U^2 of DLLE/2D.

third-order tensors. The ORL database contains 400 images of 40 individuals with the images captured at different times and with variations that include expression and facial details. All the images are normalized to a resolution of 56×46. For the third-order tensor representation, we extracted 40 Gabor features of five different scales and eight different orientations at down-sampled positions [39], such that each image is encoded as a third-order tensor of size $28 \times 23 \times 40$. In this experiment, we choose the first four images of each person for training and use the remaining six images for testing.

The AR face database contains 126 people with different facial expressions under different lighting conditions. Due to incompleteness of the data, we choose 90 persons with a full set of photographs, as in [47], for experimentation. We choose four images (labeled 1 to 4) per person from the first session for training, and use another four images (labeled 14 to 17) per person from the second session for testing. Note that the two sessions occurred two weeks apart. All the images are normalized in size to 30×20. For the third-order tensor representation, again we extracted 40 Gabor features of five different scales and eight different orientations, such that each image is encoded as a third-order tensor of size $30 \times 20 \times 40$. Some sample images in the ORL and AR databases are shown in Fig. 1.

First, we take the ORL database with a second-order data tensor (matrix) representation as an example to analyze the convergence of 2DLDA [54] and DLLE/2D. Note that in 2DLDA, an iterative scheme similar to Procedure 2 was proposed to compute the projection matrices; however, it employs the ratio formulation in (3), whereas DLLE/2D utilizes the difference formulation in (7). Fig. 3 shows convergence results, in which the horizontal axis gives the number of iterations, and the vertical axis is the similarity of two adjacent projection matrices, i.e., $\mathrm{Tr}[U_t^k (U_{t-1}^k)^T]/m'_k$, where $m'_1 = m'_2 = 10$ in this figure. The convergence problem of 2DLDA is apparent, while DLLE/2D converges smoothly, demonstrating the advantage of the proposed difference formulation. We also observed similar trends for other parameter configurations of m'_1, m'_2, and on other databases.

Face recognition accuracy is listed in Table II, and Fig. 4 shows the recognition rates of different feature dimensions for 1-D vector-based, 2-D matrix-based, and 3-D tensor-based algorithms on the ORL database. On the ORL database, the performance of DLLE/2D and LDA/2D is seen to be comparable when the feature dimension is small, as shown in Fig. 4; however, the best result of DLLE/2D is more accurate than that of LDA/2D among various parameter configurations, as shown in Table II. Note that DLLE/2D and DLLE/3D are the tensorization versions of DLLE; LDA/2D is 2DLDA in [54]; LDA/3D is the DATER algorithm in [50].

C. Discussion of Face Recognition Experiments

From the preceding experiments, several observations can be made.

1) DLLE along its linearization and tensorization outperforms the corresponding versions of LDA, and DLLE/L achieves higher accuracy than LLE/L. Although LDA considers label information and LLE considers local manifold structure, *DLLE benefits from utilizing both*.
2) The recognition rate of LLE/L is better than LDA/L on the ORL database and worse than LDA on the AR and FERET databases. The results demonstrate that, when the training set can well characterize the data distribution, such as the case in ORL database, LLE/L has the potential to outperform the LDA/L, as reported in [11]; yet when the data set distribution is complex and the training data cannot well represent the data distribution, such as the cases on the AR and FERET databases, LLE/L seems to work much worse than LDA/L in these cases. All the results show that LLE/L is not better than DLLE/L in all cases.
3) Algorithms based on tensor representations are seen to be generally superior to linear algorithms, an observation that is consistent with recent studies from several other research groups [6], [12], [32], [53], [54]. We also observe one exception for the 2-D case on the AR database. A possible reason for this discrepancy is that the training data and the test data in the AR database are from different sessions and thus may not have the same distributions, which may cause some amount of performance degradation for a high-order representation. In practice, we usually collect enough training data to make the distribution of training set similar to the test set. We also note that for the 3-D case on the AR database, LDA/3D and DLLE/3D outperform the corresponding 1-D-based algorithms, which suggests that Gabor features can bring more robustness even in this case.
4) As in other previous work [6], [10]–[12], [29], [37], [53], [54], several parameters need to be decided beforehand. For the linearization-based algorithms DLLE/L, LDA/L, and LLE/L, we set the PCA dimension according to a 98% energy criterion, as in [10]. For tensorization algorithms, we set $T_{\max}$ as 5, since it is high enough for algorithm convergence, as shown in Fig. 3. For DLLE, we also need to decide the parameters h, k_1, and k_2. In Fig. 5, we plot the recognition rates with respect to different balance terms h on the FERET and ORL databases. It can be seen that the result in the case of $h = 1$ is usually

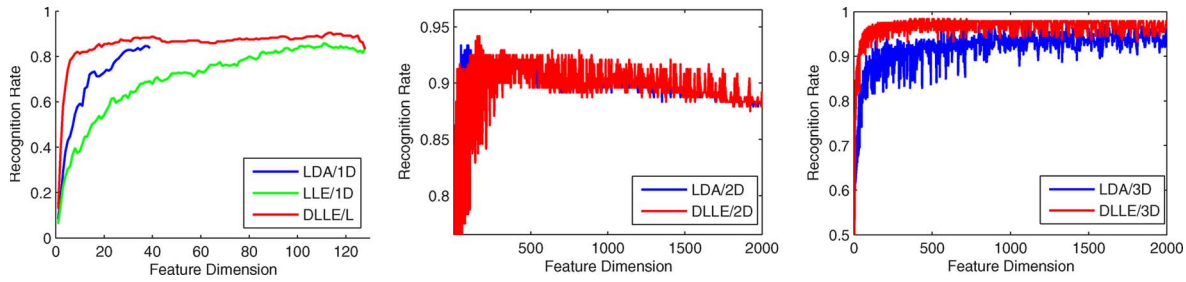


Fig. 4. Top-1 recognition rates (in percentages) with different feature dimensions on the ORL database with (left) 1-D vector input, (middle) 2-D matrix input, and (right) 3-D tensor input. For matrix and tensor-based algorithms, the feature dimension is the product of the reduced dimensions.

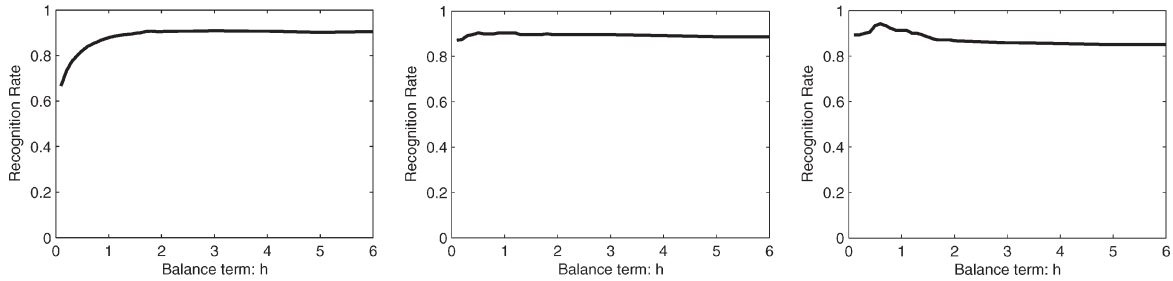


Fig. 5. Top-1 recognition rates (in percentages) with different balance terms h on the FERET database for (left) DLLE/L, (middle) ORL database for DLLE/L, and (right) DLLE/2D. The performance for $h = 1$ is generally comparable to the best result among other values of h.

comparable to the best result among other values of h. In Fig. 6, we also plot the recognition rates (in percentages) with different values of k_1 and k_2 on the FERET and ORL databases. Again, the performance is comparable among different parameter configurations. To achieve the best recognition result, we experimentally set h from 0.1 to 2 at intervals of 0.1, and from 3 to 6 at intervals of 1; k_1 is sampled at 2 and 3, and k_2 from 100 to 500 at intervals of 100. The best result among these sampled configurations is reported. Nevertheless, in Figs. 5 and 6, it can be seen that even with $h = 1$ and arbitrary k_1 and k_2, the performance is comparable to the best result. How to choose the optimal parameter configuration is an open problem for future work.

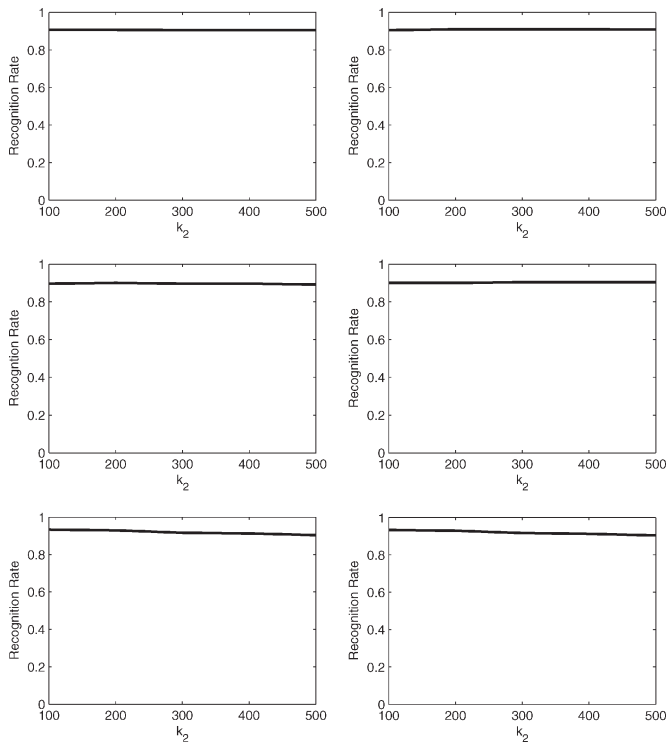


Fig. 6. Top-1 recognition rates (in percentages) with different k_1 and k_2. The left column exhibits results from $k_1 = 2$, and the second column shows results from $k_1 = 3$. The top, middle, and bottom rows are the result from DLLE/L on the FERET database, DLLE/L on the ORL database, and DLLE/2D on the ORL database. From them, it is obvious that the performance is comparable among different parameter configurations.

D. Gait Recognition on USF HumanID Gait Database

We report here our experimental results on gait recognition. Recently, Sarkar *et al.* [30] constructed the largest gait database: the USF HumanID database. This database consists of people walking in elliptical paths in front of the camera. For each person, there are up to five covariates: 1) viewpoints (left/right); 2) shoe types (A/B); 3) surface types (grass/concrete); 4) carrying conditions (with/without a briefcase); and 5) time of year (May/November; the time covariate implicitly contains the change of shoe and clothing). Therefore, there are up to 32 sequences for each person, and the full data set consists of 1870 sequences from 122 individuals. Sarkar *et al.* fixed one gallery set and created 12 probe sets to test performance under different conditions. Individuals are unique in the gallery and probe sets, and no common sequences exist among the gallery set and all the probe sets. The differences between the gallery set and the probe sets are listed in Table III, and more details about the database can be found in [30]. Some binary silhouettes and associated gray-level average silhouettes are shown in Fig. 7.

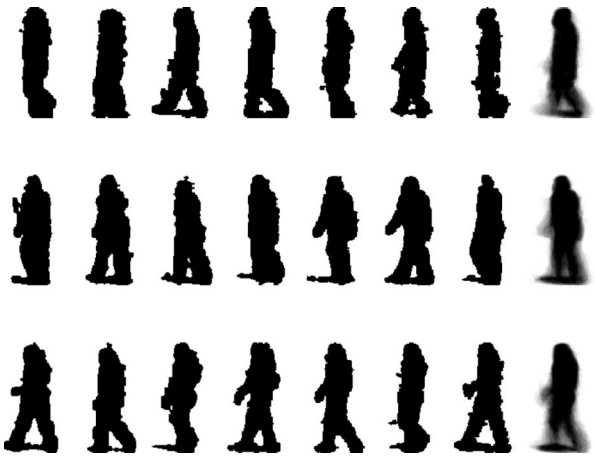


Fig. 7. Normalized and aligned binary silhouettes in the leftmost seven columns, where each row represents a different person. The rightmost column displays the corresponding gray-level average silhouettes.

Considering that recently proposed gait recognition algorithms [9], [30] are based on 1-D input, in this paper, we only focus on the comparison of DLLE/L with them. For real gait data, we list the results of the baseline algorithm [30] and the work of Han and Bhanus [9] in Table IV. For a fair comparison, we use the same binary silhouettes and gait period lengths provided in Fig. 7. In Table IV, we clearly observe that DLLE/L outperforms the state-of-the-art algorithms [9], [30] in most cases. We set the parameters h, k_1, and k_2 of DLLE/L to those we used in face recognition.

VI. Conclusion

In this paper, we present a formulation for supervised manifold learning called DLLE. In addition, we extend it to handle input data represented as high-order tensors. We conduct a comprehensive set of experiments to evaluate DLLE along with its linearization and tensorization. These comparison studies clearly demonstrate that: 1) DLLE along with its linearization and tensorization outperforms the related versions of LDA, and DLLE/L is better than LLE/L, which demonstrates the importance of utilizing both *label information* and *local manifold structure*; 2) algorithms based on tensor representations are generally superior to linear algorithms, which is consistent with recent published works [6], [12], [31], [32], [45], [53], [54]; and 3) for human gait recognition, DLLE/L is better than the state-of-the-art gait recognition algorithms [9], [30] on the standard USF gait database in most cases. In the future, we plan to extend other popular unsupervised manifold learning algorithms such as ISOMAP to variants that jointly consider label information and directly handle high-order tensor input. In addition, we plan to conduct more experiments to compare the performance of DLLE based on nonlinear mappings with kernel tricks, and to investigate automatic methods for selecting optimal parameters for h, k_1, and k_2.

References

[1] P. Belhumeur, J. Hespanha, and D. 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.

[2] M. Belkin and P. Niyogi, "Laplacian Eigenmaps and spectral techniques for embedding and clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 585–591.

[3] C. BenAbdelkader and R. Cutler, "Motion-based recognition of people in EigenGait space," in *Proc. IEEE Int. Conf. Autom. Face Gesture Recog.*, 2002, pp. 254–259.

[4] Y. Bengio, J. Paiement, and P. Vincent, "Out-of-sample extensions for LLE, isomap, MDS, eigenmaps, and spectral clustering," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 177–184.

[5] H. Chen, H. Chang, and T. Liu, "Local discriminant embedding and its variants," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 846–853.

[6] G. Dai and D. Yeung, "Tensor embedding methods," in *Proc. Nat. Conf. Artif. Intell.*, 2006, pp. 330–335.

[7] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, 2nd ed. New York: Academic, 1991.

[8] G. Guo and C. R. Dyer, "Learning from examples in the small sample case: Face expression recognition," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 35, no. 3, pp. 477–488, Jun. 2005.

[9] J. Han and B. Bhanu, "Individual recognition using gait energy image," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 2, pp. 316–322, Feb. 2006.

[10] X. He, S. Yan, Y. Hu, P. Niyogi, and H. Zhang, "Face recognition using Laplacianfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 3, pp. 328–340, Mar. 2005.

[11] X. He, D. Cai, S. Yan, and H. Zhang, "Neighborhood preserving embedding," in *Proc. IEEE Int. Conf. Comput. Vis.*, 2005, pp. 1208–1213.

[12] X. He, D. Cai, and P. Niyogi, (2005). "Tensor subspace analysis," in *Proc. Adv. Neural Inf. Process. Syst.* [Online]. Available: http://people.cs.uchicago.edu/~niyogi/papersps/HeCaiNiytensor.pdf

[13] T. Kolda, "Orthogonal tensor decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 23, no. 1, pp. 243–255, 2001.

[14] H. Kong, L. Wang, E. K. Teoh, J. Wang, and R. Venkateswarlu, "A framework of 2D Fisher discriminant analysis: Application to face recognition with small number of training samples," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 1083–1088.

[15] L. Lathauwer, B. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, Mar.–May 2000.

[16] L. Lathauwer, B. Moor, and J. Vandewalle, "On the best rank-1 and rank-(R1, R2, . . ., RN) approximation of high-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1324–1342, Mar.–May 2000.

[17] D. Lee and H. Seung, "Algorithms for non-negative matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2001, pp. 556–562.

[18] H. Li, T. Jiang, and K. Zhang, "Efficient and robust feature extraction by maximum margin criterion," in *Proc. Adv. Neural Inf. Process. Syst.*, 2003, pp. 97–104.

[19] Q. Liu, H. Lu, and S. Ma, "Improving kernel Fisher discriminant analysis for face recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 1, pp. 42–49, Jan. 2004.

[20] Z. Liu and S. Sarkar, "Simplest representation yet for gait recognition: Averaged silhouette," in *Proc. IEEE Int. Conf. Pattern Recog.*, 2004, pp. 211–214.

[21] A. Martinez and R. Benavente, "The AR face database," Comput. Vis. Center, Barcelona, Spain, CVC Tech. Rep. 24, Jun. 1998.

[22] K. Mtiller, S. Mika, G. Riitsch, K. Tsuda, and B. Scholkopf, "An introduction to kernel-based learning algorithms," *IEEE Trans. Neural Netw.*, vol. 12, no. 2, pp. 181–201, Mar. 2001.

[23] Olivetti, *Olivetti & Oracle Research Laboratory Face Database of Faces*. [Online]. Available: http://www.cam-orl.co.uk/facedatabase.html

[24] P. Phillips, H. Moon, S. Rizvi, and P. Rauss, "The FERET evaluation methodology for face-recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.

[25] P. Phillips, *The Facial Recognition Technology (FERET) Database*. [Online]. Available: http://www.itl.nist.gov/iad/humanid/feret/feret_master.html

[26] P. Phillips, P. Grother, R. Micheals, D. Blackburn, E. Tabassi, and M. Bone, *Face Recognition Vendor Test: Evaluation Report*, Mar. 2003. [Online]. Available: http://www.frvt.org

[27] A. Rangarajan, "Learning matrix space image representations," in *Proc. 3rd Int. Workshop EMMCVPR*, 2001, vol. 2134, pp. 153–168.

[28] D. Ridder, O. Kouropteva, O. Okun, M. Pietikainen, and R. Duin, "Supervised locally linear embedding," in *Proc. Artif. Neural Netw. Neural Inf. Process.*, 2003, pp. 333–341.

[29] S. Roweis and L. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *Science*, vol. 290, no. 5500, pp. 2323–2326, Dec. 2000.

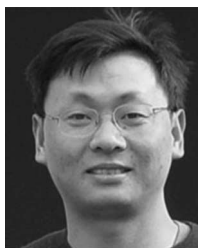
[30] S. Sarkar, P. Phillips, Z. Liu, I. Vega, P. Grother, and K. Bowyer, "The HumanID gait challenge problem: Data sets, performance, and analysis," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 2, pp. 162–177, Feb. 2005.
[31] A. Shashua and A. Levin, "Linear image coding for regression and classification using the tensor-rank principle," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2001, pp. 42–49.
[32] D. Tao, X. Li, W. Hu, S. Maybank, and X. Wu, "Supervised tensor learning," in *Proc. IEEE Int. Conf. Data Mining*, 2005, pp. 450–457.
[33] D. Tao, X. Li, X. Wu, and S. Maybank, "Human carrying status in visual surveillance," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2006, pp. 1670–1677.
[34] D. Xu, S. Yan, D. Tao, S. Lin, and H. Zhang, "Marginal Fisher analysis and its variants for human gait recognition and content based image retrieval," *IEEE Trans. Image Process.*, vol. 16, no. 11, pp. 2811–2821, Nov. 2007.
[35] D. Tao, X. Li, W. Hu, S. Maybank, and X. Wu, "Supervised tensor learning," *Knowl. Inf. Syst.*, vol. 13, no. 1, pp. 1–42, Sep. 2007.
[36] D. Tao, X. Li, X. Wu, and S. Maybank, "General tensor discriminant analysis and Gabor features for gait recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 10, pp. 1700–1715, Oct. 2007.
[37] J. Tenenbaum, V. Silva, and J. Langford, "A global geometric framework for nonlinear dimensionality reduction," *Science*, vol. 290, no. 5500, pp. 2319–2323, Dec. 2000.
[38] M. Turk and A. Pentland, "Face recognition using Eigenfaces," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 1991, pp. 586–591.
[39] L. Wiskott, J. Fellous, N. Kruger, and C. Malsburg, "Face recognition by elastic bunch graph matching," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 775–779, Jul. 1997.
[40] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear subspace analysis of image ensembles," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2003, pp. 93–99.
[41] M. A. O. Vasilescu, "Human motion signatures: Analysis, synthesis, recognition," in *Proc. IEEE Int. Conf. Pattern Recog.*, 2002, pp. 456–460.
[42] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *Proc. Eur. Conf. Comput. Vis.*, Copenhagen, Denmark, 2002, pp. 447–460.
[43] M. A. O. Vasilescu, "Multilinear independent components analysis," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 547–553.
[44] M. Vlachos, C. Domeniconi, D. Gunopulos, G. Kollios, and N. Koudas, "Non-linear dimensionality reduction techniques for classification and visualization," in *Proc. ACM Int. Conf. Knowl. Discovery Data Mining*, 2002, pp. 645–651.
[45] H. Wang and N. Ahuja, "Rank-R approximation of tensors using image-as-matrix representation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 346–353.
[46] L. Wang, H. Ning, T. Tan, and W. Hu, "Fusion of static and dynamic body biometrics for gait recognition," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 14, no. 2, pp. 149–158, Feb. 2004.
[47] X. Wang and X. Tang, "Subspace analysis using random mixture models," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 574–580.
[48] D. Xu, S. Yan, L. Zhang, H. Zhang, Z. Liu, and H. Shum, "Concurrent subspaces analysis," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 203–208.
[49] S. Yan, D. Xu, B. Zhang, and H. Zhang, "Graph-embedding: A general framework for dimensionality reduction," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 830–837.
[50] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H. Zhang, "Discriminant analysis with tensor representation," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 526–532.
[51] S. Yan, D. Xu, L. Zhang, B. Zhang, and H. Zhang, "Coupled kernel-based subspace learning," in *Proc. IEEE Int. Conf. Comput. Vis. Pattern Recog.*, 2005, pp. 645–650.
[52] J. Yang, D. Zhang, A. F. Frangi, and J. Yang, "Two-dimensional PCA: A new approach to appearance-based face representation and recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 26, no. 1, pp. 131–137, Jan. 2004.
[53] J. Ye, "Generalized low rank approximations of matrices," in *Proc. Int. Conf. Mach. Learn.*, 2004, pp. 887–894.
[54] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Proc. Adv. Neural Inf. Process. Syst.*, 2004, pp. 1569–1576.
[55] S. Zhou, "Matrix-based kernel subspace methods," SCR, Tech. Rep. SCR-05-TR-773, 2005.

Xuelong Li (M'02–SM'07) holds a permanent post in the School of Computer Science and Information Systems, Birkbeck College, University of London, London, U.K. and is a Visiting Professor with Tianjin University, Tianjin, China. His research interests include digital image/video processing, analysis, retrieval, and indexing, pattern recognition, biometrics, and visual surveillance. His research activities are partly sponsored by the Engineering and Physical Sciences Research Council, the British Council, Royal Society, etc. He has around 90 scientific publications.

Dr. Li is an Associate Editor of the IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS— PART B: CYBERNETICS, IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS, IEEE TRANSACTIONS ON IMAGE PROCESSING, and IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY. He is also an Editor of two books, an Editorial Board Member of several other journals, and a Guest Coeditor of seven special issues. He is the recipient of several best paper awards and nominations. He has served as a Chair or a Cochair of a dozen conferences, including as a Publicity Cochair of the 2008 IEEE International Conference on Systems, Man, and Cybernetics (SMC), and a Program Committee Member for more than 70 conferences, including the annual SMC conferences. He is a member of several technical committees, including the IEEE SMC Technical Committees on Computational Intelligence, Machine Learning, and Self-Organization and Cybernetics for Informatics.

Stephen Lin received the B.S.E. degree in electrical engineering from Princeton University, Princeton, NJ, and the Ph.D. degree in computer science and engineering from the University of Michigan, Ann Arbor.

He is currently a Lead Researcher with the Internet Graphics Group of Microsoft Research Asia, Beijing, China. His research interests include photometric analysis in computer vision and reflectance modeling in computer graphics.



Shuicheng Yan (M'06) received the B.S. and Ph.D. degrees from Peking University, Beijing, China, in 1999 and 2004, respectively.

He is currently an Assistant Professor with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore. His research interests include computer vision, machine learning, and data mining.

Dong Xu (M'07) received the B.Eng. and Ph.D. degrees from the University of Science and Technology of China, Hefei, in 2001 and 2005, respectively.

He is currently an Assistant Professor with the School of Computer Engineering, Nanyang Technological University, Singapore. During his Ph.D. studies, he was with Microsoft Research Asia, Beijing, China and The Chinese University of Hong Kong, Shatin. He has also spent one year at Columbia University, New York, as a Postdoctoral Research Scientist. His research interests include computer vision, pattern recognition, statistical learning, and multimedia content analysis.