Uncorrelated Multilinear Discriminant Analysis With Regularization and Aggregation for Tensor Object Recognition

Haiping Lu, *Student Member, IEEE*, Konstantinos N. Plataniotis, *Senior Member, IEEE*, and Anastasios N. Venetsanopoulos, *Life Fellow, IEEE*

***Abstract*—This paper proposes an uncorrelated multilinear discriminant analysis (UMLDA) framework for the recognition of multidimensional objects, known as tensor objects. Uncorrelated features are desirable in recognition tasks since they contain minimum redundancy and ensure independence of features. The UMLDA aims to extract uncorrelated discriminative features directly from tensorial data through solving a tensor-to-vector projection. The solution consists of sequential iterative processes based on the alternating projection method, and an adaptive regularization procedure is incorporated to enhance the performance in the small sample size (SSS) scenario. A simple nearest-neighbor classifier is employed for classification. Furthermore, exploiting the complementary information from differently initialized and regularized UMLDA recognizers, an aggregation scheme is adopted to combine them at the matching score level, resulting in enhanced generalization performance while alleviating the regularization parameter selection problem. The UMLDA-based recognition algorithm is then empirically shown on face and gait recognition tasks to outperform four multilinear subspace solutions (MPCA, DATER, GTDA, TR1DA) and four linear subspace solutions (Bayesian, LDA, ULDA, R-JD-LDA).**

***Index Terms*—Dimensionality reduction, face recognition, feature extraction, fusion, gait recognition, multilinear discriminant analysis, regularization, tensor objects.**

I. Introduction

TODAY, there are growing interests in the processing of multidimensional objects, formally known as tensor objects, in a large number of emerging applications. Tensors are considered to be the extensions of vectors and matrices. The elements of a tensor are to be addressed by a number of indices [1], where the number of indices used in the description defines the order of the tensor object and each index defines one "mode." By this definition, vectors are first-order tensors and matrices are second-order tensors. For example, gray-level images are naturally second-order tensors with the column and row modes [2], [3] and color images are 3-D objects (third-order tensors) with the column, row, and color modes [4]. Three-dimensional gray-level objects [5], such as 3-D gray-level faces [6], [7], are naturally third-order tensors with the column, row and depth modes, while the popular Gabor representation of gray-level images [8] are third-order tensors with the column, row, and Gabor modes. Many sequential (space-time) signals, such as surveillance video sequences [9], are naturally higher order tensors. Gray-level video sequences can be viewed as third-order tensors with the column, row, and time modes and color video sequences are fourth-order tensors with an additional color mode. Among the wide range of applications involving tensor objects [10], feature extraction for recognition purposes is an arguably most important one. Therefore, this paper focuses on feature extraction for tensor object recognition.

In pattern recognition applications, the tensor space where a typical tensor object is specified is often high dimensional, and recognition methods operating directly on this space suffer from the so-called curse of dimensionality [11]. On the other hand, the entries of a tensor object are often highly correlated with surrounding entries, and the samples from a particular tensor object class, such as face images, are usually highly constrained and belong to a subspace, a manifold of intrinsically low dimension [11], [12]. Feature extraction or dimensionality reduction is thus an attempt to transform a high-dimensional data set into a low-dimensional space of equivalent representation while retaining most of the underlying structure [13]. Traditional feature extraction algorithms, such as the classical principal component analysis (PCA) and linear discriminant analysis (LDA), are linear algorithms that operate on 1-D objects, i.e., first-order tensors (vectors). To apply these linear algorithms to higher order (greater than one) tensor objects, such as images and videos, these tensor objects have to be reshaped (vectorized) into vectors first. However, it is well understood that such reshaping (vectorization) breaks the natural structure and correlation in the original data, reducing redundancies and/or higher order dependencies present in the original data set, and losing potentially more compact or useful representations that can be obtained in the original tensorial forms [10]. Thus, dimensionality reduction algorithms operating directly on the tensor objects rather than their vectorized versions are desirable.

Manuscript received December 22, 2007; revised May 20, 2008; accepted August 05, 2008. First published December 12, 2008; current version published January 05, 2009. This paper was presented in part at the Biometrics Symposium, Baltimore, MD, September 11–13 2007. This work was supported in part by the Ontario Centres of Excellence through the Communications and Information Technology Ontario Partnership Program and by the Bell University Labs—at the University of Toronto.

H. Lu is with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4 Canada (e-mail: haiping@comm.utoronto.ca).

K. N. Plataniotis is with the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G4 Canada and also with the Ryerson University, Toronto, ON M5B 2K3 Canada (e-mail: kostas@comm.toronto.edu).

A. N. Venetsanopoulos is with the Ryerson University, Toronto, ON M5B 2K3 Canada (e-mail: anv@comm.toronto.edu).

Digital Object Identifier 10.1109/TNN.2008.2004625

Recently, multilinear subspace feature extraction algorithms [2], [10], [14], [15] operating directly on the tensorial representations rather than their vectorized versions are emerging, especially in the popular area of biometrics-based human recognition, e.g., face and gait recognition. The multilinear principal component analysis (MPCA) framework [10], a multilinear extension of the PCA, determines a multilinear projection that projects the original tensor objects into a lower dimensional tensor subspace while preserving the variation in the original data. Similar to PCA, MPCA is an unsupervised method as well and the feature extraction process does not make use of the class information. On the other hand, supervised multilinear feature extraction algorithms have also been developed. Since LDA is a classical algorithm that has been very successful and applied widely in various applications, there have been several variants of its multilinear extension proposed, named multilinear discriminant analysis (MLDA) in general in this paper. The 2-D LDA (2DLDA) first introduced in [16] was later extended to perform discriminant analysis on more general tensorial inputs [2]. In this so-called discriminant analysis with tensor representation (DATER)[1] approach of [2], a tensor-based scatter ratio criterion is maximized. A so-called general tensor discriminant analysis (GTDA) algorithm is proposed in [15] where a scatter difference criterion is maximized [15]. DATER and GTDA are both based on the tensor-to-tensor projection (TTP) [17]. In contrast, the tensor rank-one discriminant analysis (TR1DA) algorithm [18], [19] obtains a number of rank-one projections with the scatter difference criterion from the repeatedly calculated residues of the original tensor data, which is in fact the heuristic method in [20] for tensor approximation, and it can be viewed to be based on the tensor-to-vector projection (TVP) [17]. The multilinear extensions of linear graph-embedding algorithms were introduced similarly in [21]–[25].

In the existing MLDA variants [2], [15], [18], [19], the attention focused mainly on the objective criterion in terms of (either the ratio of or the difference between) the between-class scatter and the within-class scatter since it is well known that the classical LDA aims to maximize Fisher's discrimination criterion (FDC). However, they did not take the correlations among features into account. In other words, an important property of the classical LDA is ignored in these developments: the classical LDA derives uncorrelated features, as proved in [26] and [27], where the uncorrelated LDA (ULDA) introduced in [28] is shown to be equivalent to the classical LDA. Uncorrelated features contain minimum redundancy and ensure independence of features [27], [29]. They are highly desirable in practical recognition tasks since the subsequent classification task can be greatly simplified.

Motivated by the discussions above, this paper aims to develop a new MLDA solution that extracts uncorrelated features, named as the uncorrelated MLDA (UMLDA). The proposed UMLDA extracts discriminative features directly from tensorial data through solving a TVP so that the traditional FDC is maximized in each elementary projection, while the features extracted are constrained to be uncorrelated. The solution is iterative, based on the alternating projection method (APM), and an adaptive regularization factor is incorporated to enhance the performance in practical applications where the input dimensionality is very high but the sample size per class is often limited, such as face or gait recognition [30], [31]. The extracted features are classified through a simple classifier. Furthermore, as different initialization or regularization of UMLDA results in different features, an aggregation scheme that combines these features at the matching score level using the simple sum rule is adopted to enhance the recognition performance, with the regularization parameter selection problem alleviated at the same time.

The main contributions of this work are as follows.

1) The introduction of a UMLDA algorithm for uncorrelated discriminative feature extraction from tensors. As a multilinear extension of LDA, the algorithm not only obtains discriminative features through maximizing the traditional scatter-ratio-based criterion, but also enforces a constraint so that the features derived are uncorrelated. This contrasts to the traditional approach of linear learning algorithms [30], [32], [33], where vector rather than tensor representation is used and thus the natural structural information is destroyed. It also differs from the MLDA variants in [2], [15], [18], and [19], where there are correlations among extracted features. Another difference from the works in [2] and [15] is that a TVP rather than a TTP is used here and this work takes a systematic approach to solve such a TVP, in contrast with the heuristic approach in [18] and [19]. Furthermore, it provides a new approach with constraint enforcement in developing multilinear learning algorithms.
2) The incorporation of an adaptive regularization procedure where the within-class scatter estimation is increased through a data-independent regularization parameter. This takes into account the practical small sample size (SSS) problem, which often arises in biometrics applications, and the iterative nature of UMLDA, different from the scatter estimation without regularization in [2], [15], [18], [19].
3) The adoption of an aggregation scheme that combines several differently initialized and differently regularized UMLDA feature extractors, which produce different features, at the matching score level to achieve enhanced recognition performance while alleviating the regularization parameter selection problem faced in most regularization methods.

The rest of this paper is organized as follows. Section II introduces the notations and basic multilinear algebra operations, as well as the TVP. In Section III, the problem of UMLDA is formulated and an iterative solution is derived, with an adaptive regularization procedure introduced for better generalization in the SSS scenario. Next, the classifier employed is described and issues regarding the initialization method, projection order, termination criteria, and convergence are addressed in this section, followed by the computational aspects and a brief discussion on the connections to LDA and other MLDA variants. Section IV presents the matching score level aggregation of

[1]Here, we use the name given when the algorithm was first proposed, which is more commonly refereed to in the literature.

TABLE I
LIST OF SYMBOLS

$\mathcal{X}_m$	the mth input tensor sample, $m = 1, ..., M$
$\mathbf{u}^{(n)}$	the n-mode projection vector, $n = 1, ..., N$
$p = 1, ..., P$	the index of the EMP
P	the number of EMPs in TVP
$\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}$	the pth EMP
$\mathbf{y}_m$	the projection of $\mathcal{X}_m$ on the TVP $\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}_{p=1}^{P}$
$\mathbf{y}_m(p) = y_{m_p} = \mathbf{g}_p(m)$	the projection of the mth sample $\mathcal{X}_m$ on the pth EMP $\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}$
$S_{B_p}^{\mathbf{y}}$	the between-class scatter of the pth projected features $\{y_{m_p}, m = 1, ..., M\}$
$S_{W_p}^{\mathbf{y}}$	the within-class scatter of the pth projected features $\{y_{m_p}, m = 1, ..., M\}$
$\mathbf{g}_p$	the pth coordinate vector
$F_p^{\mathbf{y}}$	$= \frac{S_{B_p}^{\mathbf{y}}}{S_{W_p}^{\mathbf{y}}}$, the Fisher's discrimination criterion for the pth EMP
k	the iteration step index in the UMLDA algorithm
K	the maximum number of iterations in UMLDA
γ	the regularization parameter
$a = 1, ..., A$	the index of the feature extractor in aggregation
A	the number of feature extractors to be aggregated
c_m	the class label for the mth training sample
$vec(\mathcal{A})$	the vectorized representation of the tensor $\mathcal{A}$
L	the number of training samples for each class (subject)
C	the number of classes (subjects) in training

multiple UMLDA feature extractors that are differently initialized and regularized to enhance the recognition performance. In Section V, experiments on two face databases and one gait database are reported. The properties of the proposed UMLDA solutions are first illustrated, and detailed recognition results are then compared against competing linear solutions as well as multilinear solutions. Finally, Section VI draws the conclusion of this work.

II. MULTILINEAR BASICS AND THE TENSOR-TO-VECTOR PROJECTION

This section introduces the foundations that are fundamentally important for the flow of this paper. First, we review the notations and some basic multilinear operations that are necessary in presenting the proposed MLDA solution. Second, the TVP used in the proposed algorithm is described in detail. Table I summarizes the important symbols used in this paper for quick reference.

A. Notations and Basic Multilinear Algebra Concepts

The notations in this paper follow the conventions in the multilinear algebra, pattern recognition, and adaptive learning literature. In this paper, we denote vectors by lowercase boldface letters, e.g., $\mathbf{x}$; matrices by uppercase boldface letters, e.g., $\mathbf{U}$; and tensors by calligraphic letters, e.g., $\mathcal{A}$. Their elements are denoted with indices in parentheses. Indices are denoted by lowercase letters and span the range from 1 to the uppercase letter of the index, e.g., $n = 1, 2, \ldots, N$. Throughout this paper, the discussion is focused on real-valued vectors, matrices, and tensors only and the extension to complex-valued data is left for future work.

An Nth-order tensor is denoted as $\mathcal{A} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$. It is addressed by N indices i_n, $n = 1, \ldots, N$, and each i_n addresses the n-mode of $\mathcal{A}$. The n-mode product of a tensor $\mathcal{A}$ by a matrix $\mathbf{U} \in \mathbb{R}^{J_n \times I_n}$, denoted by $\mathcal{A} \times_n \mathbf{U}$, is a tensor with entries

$$(\mathcal{A} \times_n \mathbf{U})(i_1, \ldots, i_{n-1}, j_n, i_{n+1}, \ldots, i_N) = \sum_{i_n} \mathcal{A}(i_1, \ldots, i_N) \cdot \mathbf{U}(j_n, i_n). \quad (1)$$

The scalar product of two tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ is defined as

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1} \sum_{i_2} \cdots \sum_{i_N} \mathcal{A}(i_1, i_2, \ldots, i_N) \cdot \mathcal{B}(i_1, i_2, \ldots, i_N). \quad (2)$$

The "n-mode vectors" of $\mathcal{A}$ are defined as the I_n-dimensional vectors obtained from $\mathcal{A}$ by varying the index i_n while keeping all the other indices fixed. A rank-one tensor $\mathcal{A}$ equals to the outer product of N vectors: $\mathcal{A} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \cdots \circ \mathbf{u}^{(N)}$, which means that $\mathcal{A}(i_1, i_2, \ldots, i_N) = \mathbf{u}^{(1)}(i_1) \cdot \mathbf{u}^{(2)}(i_2) \cdots \mathbf{u}^{(N)}(i_N)$ for all values of indices. Unfolding $\mathcal{A}$ along the n-mode is denoted as $\mathbf{A}_{(n)} \in \mathbb{R}^{I_n \times (I_1 \times \cdots \times I_{n-1} \times I_{n+1} \times \cdots \times I_N)}$, and the column vectors of $\mathbf{A}_{(n)}$ are the n-mode vectors of $\mathcal{A}$. An example of the 1-mode vectors of a tensor $\mathcal{A} \in \mathbb{R}^{10 \times 8 \times 6}$ can be found in Fig. 1(a).

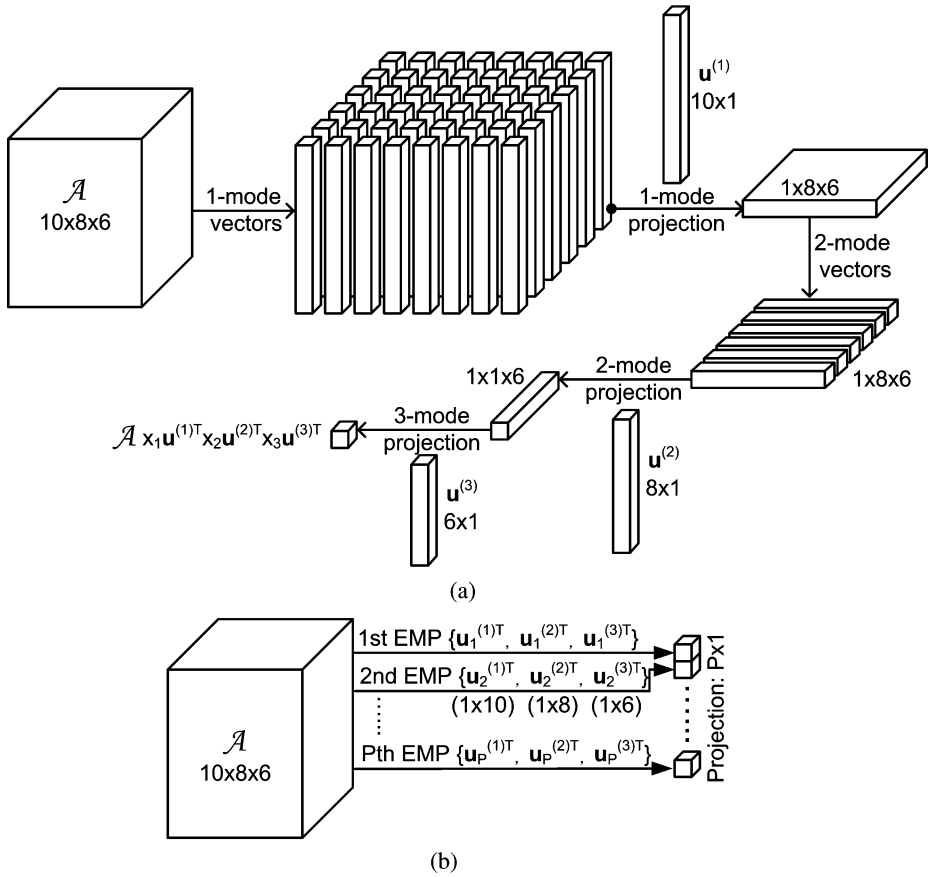

Fig. 1. (a) Elementary multilinear projection (EMP). (b) Tensor-to-vector projection (TVP).

B. Tensor-to-Vector Projection

The UMLDA framework developed in this paper takes a multilinear subspace (or tensor subspace) [21] approach of feature extraction, where tensorial data is projected into a subspace for better discrimination. As discussed in Section I, there are two general forms of multilinear projection: the TTP [2], [10], [15] and the TVP [18], [19]. Since the projections obtained by TTP can be viewed as a set of interdependent projections [17], the features extracted through TTP are likely to be correlated rather than uncorrelated. Therefore, we choose to develop the UMLDA by determining a subspace of tensor objects through TVP rather than TTP.

The TVP projects a tensor to a vector and it can be viewed as multiple projections from a tensor to a scalar, as illustrated in Fig. 1(b), where a TVP of a tensor $\mathcal{A} \in \mathbb{R}^{10\times 8\times 6}$ to a $P \times 1$ vector consists of P projections from $\mathcal{A}$ to a scalar. Thus, the projection from a tensor to a scalar is considered first. A tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_N}$ can be projected to a point y through N unit projection vectors $\{\mathbf{u}^{(1)^T}, \mathbf{u}^{(2)^T}, \ldots, \mathbf{u}^{(N)^T}\}$ as $y = \mathcal{X} \times_1 \mathbf{u}^{(1)^T} \times_2 \mathbf{u}^{(2)^T} \cdots \times_N \mathbf{u}^{(N)^T}$, $\|\mathbf{u}^{(n)}\| = 1$ for $n = 1, \ldots, N$, where $\|\cdot\|$ is the Euclidean norm for vectors. It can be written in the scalar product (2) as $y = \langle \mathcal{X}, \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \cdots \circ \mathbf{u}^{(N)} \rangle$. Denote $\mathcal{U} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \cdots \circ \mathbf{u}^{(N)}$, then we have $y = \langle \mathcal{X}, \mathcal{U} \rangle$. We name this multilinear projection $\{\mathbf{u}^{(1)^T}, \mathbf{u}^{(2)^T}, \ldots, \mathbf{u}^{(N)^T}\}$ as an elementary multilinear projection (EMP), which is the projection of a tensor on a single line (resulting a scalar) and it consists of one projection vector in each mode. Fig. 1(a) illustrates an EMP of a tensor $\mathcal{A} \in \mathbb{R}^{10\times 8\times 6}$. As pointed out in [17], an EMP can be viewed as a constrained linear projection since $\langle \mathcal{X}, \mathcal{U} \rangle = \langle \mathrm{vec}(\mathcal{X}), \mathrm{vec}(\mathcal{U}) \rangle = [\mathrm{vec}(\mathcal{U})]^T \mathrm{vec}(\mathcal{X})$, where $\mathrm{vec}(\mathcal{A})$ denotes the vectorized representation of the tensor $\mathcal{A}$ [34].

Thus, the TVP of a tensor object $\mathcal{X}$ to a vector $\mathbf{y} \in \mathbb{R}^P$ in a P-dimensional vector space consists of P EMPs $\{\mathbf{u}_p^{(1)^T}, \mathbf{u}_p^{(2)^T}, \ldots, \mathbf{u}_p^{(N)^T}\}$, $p = 1, \ldots, P$, which can be written concisely as $\{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}_{p=1}^P$. The TVP from $\mathcal{X}$ to $\mathbf{y}$ is then written as $\mathbf{y} = \mathcal{X} \times_{n=1}^N \{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}_{p=1}^P$, where the pth component of $\mathbf{y}$ is obtained from the pth EMP as $\mathbf{y}(p) = \mathcal{X} \times_1 \mathbf{u}_p^{(1)^T} \times_2 \mathbf{u}_p^{(2)^T} \cdots \times_N \mathbf{u}_p^{(N)^T}$.

III. Uncorrelated MLDA With Regularization for Tensor Object Recognition

This section proposes the UMLDA-based tensor object recognition system, which is a typical recognition system as shown in Fig. 2. The normalization step is a standard processing to ensure all input tensors having the same size [10]. At the core of this system is the UMLDA framework for feature extraction to be presented in this section: the UMLDA with regularization (R-UMLDA). The R-UMLDA produces vector features that

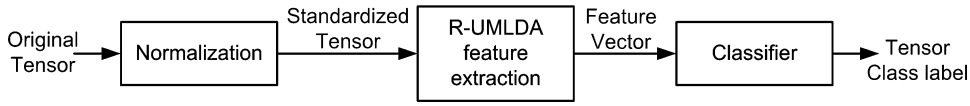

Fig. 2. UMLDA-based tensor object recognition.

can be fed into standard classifiers for classification. Besides the description of the fundamental units in Fig. 2, implementation issues, computational aspects, and the connections with other algorithms are discussed in this section as well.

A. The Uncorrelated Multilinear Discriminant Analysis With Regularization (R-UMLDA)

In the presentation, for the convenience of discussion, the training samples are assumed to be zero-mean[2] so that the constraint of uncorrelated features is the same as orthogonal features.[3] Before formally stating the objective of the UMLDA, a number of definitions are needed.

The classical FDC in LDA [32] is defined as the scatter ratio for vector samples. Here, we adapt it to scalar samples, which can be viewed as the degenerated version. The pth projected (scalar) features are $\{y_{m_p}, m = 1, \ldots, M\}$, where M is the number of training samples and y_{m_p} is the projection of the mth sample $\mathcal{X}_m$ by the pth EMP $\{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}$: $y_{m_p} = \mathcal{X}_m \times_{n=1}^{N} \{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\}$. Their corresponding between-class scatter $S_{B_p}^{\mathbf{y}}$ and the within-class scatter $S_{W_p}^{\mathbf{y}}$ are

$$S_{B_p}^{\mathbf{y}} = \sum_{c=1}^{C} N_c (\bar{y}_{c_p} - \bar{y}_p)^2 \quad S_{W_p}^{\mathbf{y}} = \sum_{m=1}^{M} (y_{m_p} - \bar{y}_{c_{m_p}})^2 \quad (3)$$

where C is the number of classes, N_c is the number of samples for class c, c_m is the class label for the mth training sample, $\bar{y}_p = (1/M)\sum_m y_{m_p} = 0$, and $\bar{y}_{c_p} = (1/N_c)\sum_{m, c_m = c} y_{m_p}$. Thus, the FDC for the pth scalar samples is $F_p^{\mathbf{y}} = S_{B_p}^{\mathbf{y}} / S_{W_p}^{\mathbf{y}}$. In addition, let $\mathbf{g}_p$ denote the pth coordinate vector, with its mth component $\mathbf{g}_p(m) = y_{m_p}$.

A formal definition of the multilinear feature extraction problem to be solved in UMLDA is then given in the following.

A set of M training tensor object samples $\{\mathcal{X}_1, \mathcal{X}_2, \ldots, \mathcal{X}_M\}$ (with zero-mean) is available for training. Each tensor object $\mathcal{X}_m \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_N}$ assumes values in the tensor space $\mathbb{R}^{I_1} \bigotimes \mathbb{R}^{I_2} \cdots \bigotimes \mathbb{R}^{I_N}$, where I_n is the n-mode dimension of the tensor and $\bigotimes$ denotes the Kronecker product. The objective of the UMLDA is to find a TVP, which consists of P EMPs $\{\mathbf{u}_p^{(n)} \in \mathbb{R}^{I_n \times 1},\ n = 1, \ldots, N\}_{p=1}^{P}$, mapping from the original tensor space $\mathbb{R}^{I_1} \bigotimes \mathbb{R}^{I_2} \cdots \bigotimes \mathbb{R}^{I_N}$ into a vector subspace $\mathbb{R}^P$ (with $P < \prod_{n=1}^{N} I_n$)

$$\mathbf{y}_m = \mathcal{X}_m \times_{n=1}^{N} \{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\}_{p=1}^{P}, \quad m = 1, \ldots, M \quad (4)$$

such that the FDC $F_p^{\mathbf{y}}$ is maximized in each EMP direction, subject to the constraint that the P coordinate vectors $\{\mathbf{g}_p \in \mathbb{R}^M, p = 1, \ldots, P\}$ are uncorrelated.

In other words, the UMLDA objective is to determine a set of P EMPs $\{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}_{p=1}^{P}$ that maximize the scatter ratio while producing features with zero correlation. Thus, the objective function for the pth EMP is

$$\{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\} = \arg\max F_p^{\mathbf{y}} \quad \text{subject to } \frac{\mathbf{g}_p^T \mathbf{g}_q}{\|\mathbf{g}_p\| \, \|\mathbf{g}_q\|} = \delta_{pq}, \qquad p, q = 1, \ldots, P \quad (5)$$

where δ_{pq} is the Kronecker delta (defined as 1 for $p = q$ and as 0 otherwise). To solve this problem, we follow the successive determination approach in the derivation of the ULDA in [28]. The P EMPs $\{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\}_{p=1}^{P}$ are determined sequentially (one by one) in P steps, with the pth step obtaining the pth EMP. This stepwise process proceeds as follows.

Step 1: Determine the first EMP $\{\mathbf{u}_1^{(n)^T},\ n = 1, \ldots, N\}$ by maximizing $F_1^{\mathbf{y}}$ without any constraint.

Step 2: Determine the second EMP $\{\mathbf{u}_2^{(n)^T},\ n = 1, \ldots, N\}$ by maximizing $F_2^{\mathbf{y}}$ subject to the constraint that $\mathbf{g}_2^T \mathbf{g}_1 = 0$.

Step 3: Determine the third EMP $\{\mathbf{u}_3^{(n)^T},\ n = 1, \ldots, N\}$ by maximizing $F_3^{\mathbf{y}}$ subject to the constraint that $\mathbf{g}_3^T \mathbf{g}_1 = 0$ and $\mathbf{g}_3^T \mathbf{g}_2 = 0$.

Step p ($p = 4, \ldots, P$): Determine the pth EMP $\{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\}$ by maximizing $F_p^{\mathbf{y}}$ subject to the constraint that $\mathbf{g}_p^T \mathbf{g}_q = 0$ for $q = 1, \ldots, p-1$.

In the following, the algorithm to compute these EMPs is presented in detail, which is summarized in the pseudocode in Fig. 3. In the figure, the stepwise process described above corresponds to the loop indexed by p.

To solve for the pth EMP $\{\mathbf{u}_p^{(n)^T},\ n = 1, \ldots, N\}$, there are N sets of parameters corresponding to N projection vectors to be determined, $\mathbf{u}_p^{(1)}, \mathbf{u}_p^{(2)}, \ldots \mathbf{u}_p^{(N)}$, one in each mode. It is desirable to determine these N sets of parameters (N projection vectors) in all modes simultaneously so that $F_p^{\mathbf{y}}$ is (globally) maximized, subject to the zero-correlation constraint. Unfortunately, this is a rather complicated nonlinear problem without an existing optimal solution, except when $N = 1$, which is the classical linear case where only one projection vector is to be solved. Therefore, we derive a suboptimal solution instead by following the principle of the alternating least square (ALS) algorithm [37]–[39], where a multilinear (least-square) optimization problem is reduced into smaller conditional subproblems that can be solved through simple established methods

[2]When the training sample mean is not zero, it can be subtracted to make the training samples to be zero-mean.

[3]Let $\mathbf{x}$ and $\mathbf{y}$ be vector observations of the variables x and y. Then, $\mathbf{x}$ and $\mathbf{y}$ are orthogonal iff $\mathbf{x}^T\mathbf{y} = 0$, and $\mathbf{x}$ and $\mathbf{y}$ are uncorrelated iff $(\mathbf{x} - \bar{x})^T(\mathbf{y} - \bar{y}) = 0$, where $\bar{x}$ and $\bar{y}$ are the means of $\mathbf{x}$ and $\mathbf{y}$, respectively [35]. Thus, two zero-mean (centered) vectors are uncorrelated when they are orthogonal [36].

Input: A set of zero-mean tensor samples $\{\mathcal{X}_m \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}, m = 1, ..., M\}$ with class labels $\mathbf{c} \in \mathbb{R}^M$, the desired feature vector length P, the regularization parameter γ, the maximum number of iterations K and a small number ϵ for testing convergence.

Output: The P EMPs $\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}_{p=1}^P$ that best separate classes in the projected space.

R-UMLDA algorithm:

For $p = 1 : P$ (step p: determine the pth EMPs)

If $p > 1$, calculate the coordinate vector $\mathbf{g}_{p-1}$: $\mathbf{g}_{p-1}(m) = \mathcal{X}_m \times_1 \mathbf{u}_{p-1}^{(1)^T} \times_2 \mathbf{u}_{p-1}^{(2)^T} ... \times_N \mathbf{u}_{p-1}^{(N)^T}$.

- For $n = 1, ..., N$, initialize $\mathbf{u}_{p_{(0)}}^{(n)} \in \mathbb{R}^{I_n}$.
- For $k = 1 : K$
  - For $n = 1 : N$
    - Calculate $\tilde{\mathbf{y}}_{m_p}^{(n)} = \mathcal{X}_m \times_1 \mathbf{u}_{p_{(k)}}^{(1)^T} ... \times_{n-1} \mathbf{u}_{p_{(k)}}^{(n-1)^T} \times_{n+1} \mathbf{u}_{p_{(k-1)}}^{(n+1)^T} ... \times_N \mathbf{u}_{p_{(k-1)}}^{(N)^T}$, for $m = 1, ..., M$.
    - Calculate $\mathbf{R}_p^{(n)}$, $\tilde{\mathbf{S}}_{B_p}^{(n)}$, and $\tilde{\mathbf{S}}_{W_p}^{(n)}$. Set $\mathbf{u}_{p_{(k)}}^{(n)}$ to be the (unit) eigenvector of $\left(\tilde{\mathbf{S}}_{W_p}^{(n)}\right)^{-1} \mathbf{R}_p^{(n)} \tilde{\mathbf{S}}_{B_p}^{(n)}$ associated with the largest eigenvalue.
  - If $k = K$ or $dist\left(\mathbf{u}_{p_{(k)}}^{(n)}, \mathbf{u}_{p_{(k-1)}}^{(n)}\right) < \epsilon$ for all n, set $\mathbf{u}_p^{(n)} = \mathbf{u}_{p_k}^{(n)}$ for all n, break.
- **Output** $\{\mathbf{u}_p^{(n)}\}$. Go the step $p + 1$ if $p < P$. Stop if $p = P$.

Fig. 3. Pseudocode implementation of the R-UMLDA algorithm for feature extraction from tensor objects.

employed in the linear case. Thus, for each EMP to be determined, the parameters of the projection vector $\mathbf{u}_p^{(n^*)}$ for each mode n^* are estimated one by one separately, conditioned on $\{\mathbf{u}_p^{(n)}, n \neq n^*\}$, the parameter values of the projection vectors for the other modes.[4] Thus, by fixing $\{\mathbf{u}_p^{(n)}, n \neq n^*\}$, a new objective function depending only on $\mathbf{u}_p^{(n^*)}$ is formulated and this conditional subproblem is linear and much simpler. The parameter estimations for each mode are obtained in this way sequentially (the n loop) and iteratively (the k loop) until a stopping criterion is met. We name this iterative method the alternating projection method (APM). It corresponds to the loop indexed by k in Fig. 3, and in each iteration k, the loop indexed by n in Fig. 3 consists of the N conditional subproblems.

To solve for $\mathbf{u}_p^{(n^*)}$ in the n^*-mode, assuming that $\{\mathbf{u}_p^{(n)}, n \neq n^*\}$ is given, the tensor samples are projected in these $(N-1)$ modes $\{n \neq n^*\}$ first to obtain (vectors)

$$\tilde{\mathbf{y}}_{m_p}^{(n^*)} = \mathcal{X}_m \times_1 \mathbf{u}_p^{(1)^T} \cdots \times_{n^*-1} \mathbf{u}_p^{(n^*-1)^T} \times_{n^*+1} \mathbf{u}_p^{(n^*+1)^T} \cdots \times_N \mathbf{u}_p^{(N)^T} \quad (6)$$

$\tilde{\mathbf{y}}_{m_p}^{(n^*)} \in \mathbb{R}^{I_{n^*}}$. This conditional subproblem then becomes to determine $\mathbf{u}_p^{(n^*)}$ that projects the vector samples $\{\tilde{\mathbf{y}}_{m_p}^{(n^*)}, m = 1, \ldots, M\}$ onto a line so that the scatter ratio is maximized, subject to the zero-correlation constraint. This is a (linear and simpler) ULDA problem with the input samples $\{\tilde{\mathbf{y}}_{m_p}^{(n^*)}, m = 1, \ldots, M\}$. The corresponding between-class scatter matrix $\tilde{\mathbf{S}}_{B_p}^{(n^*)}$ and the (regularized) within-class scatter matrix $\tilde{\mathbf{S}}_{W_p}^{(n^*)}$ are then defined as

$$\tilde{\mathbf{S}}_{B_p}^{(n^*)} = \sum_{c=1}^{C} N_c \left(\bar{\tilde{\mathbf{y}}}_{c_p}^{(n^*)} - \bar{\tilde{\mathbf{y}}}_p^{(n^*)}\right)\left(\bar{\tilde{\mathbf{y}}}_{c_p}^{(n^*)} - \bar{\tilde{\mathbf{y}}}_p^{(n^*)}\right)^T \quad (7)$$

[4]It should be noted that this conditioning implies that the estimation of the parameters in one mode is dependent on the parameters in all the other modes.

$$\tilde{\mathbf{S}}_{W_p}^{(n^*)} = \sum_{m=1}^{M} \left(\tilde{\mathbf{y}}_{m_p}^{(n^*)} - \bar{\tilde{\mathbf{y}}}_{c_{m_p}}^{(n^*)}\right)\left(\tilde{\mathbf{y}}_{m_1}^{(n^*)} - \bar{\tilde{\mathbf{y}}}_{c_{m_p}}^{(n^*)}\right)^T + \gamma \cdot \lambda_{\max}(\breve{\mathbf{S}}_W^{(n^*)}) \cdot \mathbf{I}_{I_{n^*}} \quad (8)$$

where $\bar{\tilde{\mathbf{y}}}_{c_p}^{(n^*)} = (1/N_c) \sum_{m, c_m = c} \tilde{\mathbf{y}}_{m_p}^{(n^*)}$, $\bar{\tilde{\mathbf{y}}}_p^{(n^*)} = (1/M) \sum_m \tilde{\mathbf{y}}_{m_p}^{(n^*)} = \mathbf{0}$, $\gamma \geq 0$ is a regularization parameter, $\mathbf{I}_{I_{n^*}}$ is an identity matrix of size $I_{n^*} \times I_{n^*}$, and $\lambda_{\max}(\breve{\mathbf{S}}_W^{(n^*)})$ is the maximum eigenvalue of $\breve{\mathbf{S}}_W^{(n^*)}$, which is the within-class scatter matrix for the n-mode vectors of the training samples, defined as

$$\breve{\mathbf{S}}_W^{(n^*)} = \sum_{m=1}^{M} \left(\mathbf{X}_{m(n^*)} - \bar{\mathbf{X}}_{c_m(n^*)}\right)\left(\mathbf{X}_{m(n^*)} - \bar{\mathbf{X}}_{c_m(n^*)}\right)^T \quad (9)$$

where $\bar{\mathbf{X}}_{c(n^*)}$ is the n^*-mode unfolded matrix of the class mean tensor $\bar{\mathcal{X}}_c = (1/N_c) \sum_{m, c_m = c} \mathcal{X}_m$. In the following, the motivation for introducing the regularization factor is explained.

In the targeted biometrics applications (and many other applications as well), the dimensionality of the input data is very high while at the same time, the number of training samples for each class is often too small to represent the true characteristics of their classes, resulting in the well-known SSS problem [30]. Furthermore, our empirical study of the iterative UMLDA algorithm (i.e., $\gamma = 0$) under the SSS scenario indicates that the iterations tend to minimize the within-class scatter towards zero in order to maximize the scatter ratio (since the scatter ratio reaches maximum of infinity when the within-class scatter is zero and the between-class scatter is nonzero). However, the estimated within-class scatter on the training data is usually much smaller than the real within-class scatter, due to limited number of samples for each class. Therefore, regularization [40], which has been used for combatting the singularity problem[5] in LDA-based algorithms under the SSS scenario [30], [41], is adopted

[5]While the numerical singularity problem is common for LDA-based algorithms, as pointed out in [2], this is not the case for MLDA. Therefore, the motivation of using regularization here is different from the linear case in this aspect.

here to improve the generalization capability of UMLDA under the SSS scenario, leading to the R-UMLDA. The regularization term is introduced in (8) so that during the iteration, less focus is put on shrinking the within-class scatter. Moreover, the regularization introduced is adaptive since γ is the only regularization parameter and the regularization term in the n^*-mode is scaled by $\lambda_{\max}(\breve{\mathbf{S}}_W^{(n^*)})$, which is an approximate estimate of the n^*-mode within-class scatter in the training data. The basic UMLDA is obtained by setting $\gamma = 0$.

With (7) and (8), we are ready to solve the P EMPs. For $p = 1$, the $\mathbf{u}_1^{(n^*)}$ that maximizes the FDC $\mathbf{u}_1^{(n^*)^T}\tilde{\mathbf{S}}_{B_1}^{(n^*)}\mathbf{u}_1^{(n^*)}/\mathbf{u}_1^{(n^*)^T}\tilde{\mathbf{S}}_{W_1}^{(n^*)}\mathbf{u}_1^{(n^*)}$ in the projected space is obtained as the unit eigenvector of $(\tilde{\mathbf{S}}_{W_1}^{(n^*)})^{-1}\tilde{\mathbf{S}}_{B_1}^{(n^*)}$ associated with the largest eigenvalue for a nonsingular $\tilde{\mathbf{S}}_{W_1}^{(n^*)}$. Next, we show how to determine the pth $(p > 1)$ EMP given the first $(p-1)$ EMPs. Given the first $(p-1)$ EMPs, the pth EMP aims to maximize the scatter ratio $F_p^{\mathbf{y}}$, subject to the constraint that features projected by the pth EMP is uncorrelated with those projected by the first $(p-1)$ EMPs. Let $\tilde{\mathbf{Y}}_p^{(n^*)} \in \mathbb{R}^{I_{n^*} \times M}$ be a matrix with its mth column to be $\tilde{\mathbf{y}}_{m_p}^{(n^*)}$, i.e., $\tilde{\mathbf{Y}}_p^{(n^*)} = [\tilde{\mathbf{y}}_{1_p}^{(n^*)}, \tilde{\mathbf{y}}_{2_p}^{(n^*)}, \ldots, \tilde{\mathbf{y}}_{M_p}^{(n^*)}]$, then the pth coordinate vector is obtained as $\mathbf{g}_p = \tilde{\mathbf{Y}}_p^{(n^*)^T}\mathbf{u}_p^{(n^*)}$. The constraint that $\mathbf{g}_p$ is uncorrelated with $\{\mathbf{g}_q, q = 1, \ldots, p-1\}$ can be written as

$$\mathbf{g}_p^T\mathbf{g}_q = \mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q = 0, \qquad q = 1, \ldots, p-1. \quad (10)$$

Thus, $\mathbf{u}_p^{(n^*)}$ $(p > 1)$ can be determined by solving the following constrained optimization problem:

$$\mathbf{u}_p^{(n^*)} = \arg\max \frac{\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}}{\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)}}$$
$$\text{subject to } \mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q = 0, \qquad q = 1, \ldots, p-1. \quad (11)$$

The solution is given by the following theorem for nonsingular $\tilde{\mathbf{S}}_{W_p}^{(n^*)}$.

Theorem 1: When $\tilde{\mathbf{S}}_{W_p}^{(n^*)}$ is nonsingular, the solution to the problem (11) is the (unit-length) generalized eigenvector corresponding to the largest generalized eigenvalue of the following generalized eigenvalue problem:

$$\mathbf{R}_p^{(n^*)}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u} = \lambda\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u} \quad (12)$$

where

$$\mathbf{R}_p^{(n^*)} = \mathbf{I}_{I_{n^*}} - \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1} \times \left(\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\right)^{-1} \times \mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}} \quad (13)$$

$$\mathbf{G}_{p-1} = [\mathbf{g}_1 \quad \mathbf{g}_2 \quad \cdots \mathbf{g}_{p-1}] \in \mathbb{R}^{M \times (p-1)}. \quad (14)$$

Proof: The proof of Theorem 1 is given in Appendix I. ■

By setting $\mathbf{R}_1^{(n^*)} = \mathbf{I}_{I_{n^*}}$ and from Theorem 1, we have a unified solution for R-UMLDA when $\tilde{\mathbf{S}}_{W_p}^{(n^*)}$ is nonsingular: for $p = 1, \ldots, P$, $\mathbf{u}_p^{(n^*)}$ is obtained as the unit eigenvector of $(\tilde{\mathbf{S}}_{W_p}^{(n^*)})^{-1}\mathbf{R}_p^{(n^*)}\tilde{\mathbf{S}}_{B_p}^{(n^*)}$ associated with the largest eigenvalue.

Next, we will describe the classification of R-UMLDA features and then give detailed discussions on implementation issues and connections to other algorithms.

B. Classification of R-UMLDA Features

Despite working directly on tensorial data, the proposed R-UMLDA algorithm is a feature extraction algorithm that produces feature vectors such as traditional linear algorithms (through TVP), as in traditional linear algorithms. For recognition tasks, the features extracted are to be fed into a classifier to get the class label, as shown in Fig. 2. In this work, the feature vectors obtained through R-UMLDA are fed into the nearest-neighbor classifier (NNC) with the Euclidean distance measure for classification. It should be noted at this point that, since this paper focuses on multilinear feature extraction, a simple classifier is preferred so that the recognition performance is mainly contributed by the feature extraction algorithms rather than the classifier. The classification accuracy of the proposed method (and other methods compared in Section V) is expected to improve if a more sophisticated classifier such as the support vector machine (SVM) is used instead of the NNC. However, such an experiment is out of the scope of this paper.

To classify a test sample $\mathcal{X}$ using NNC, $\mathcal{X}$ is first projected to a feature vector $\mathbf{y}$ through the TVP obtained by R-UMLDA: $\mathbf{y} = \mathcal{X} \times_{n=1}^{N} \{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}_{p=1}^{P}$. The nearest neighbor is then found as $m^* = \arg\min_m \|\mathbf{y} - \mathbf{y}_m\|$, where $\mathbf{y}_m$ is the feature vector for the mth training sample. The class label of the m^*th training sample c_{m^*} is then assigned to the test sample $\mathcal{X}$.

C. Initialization, Projection Order, Termination, and Convergence

In this section, we discuss the various implementation issues of R-UMLDA, in the order of the algorithm flow in Fig. 3: initialization, projection order, termination, and convergence.

As the determination of each EMP $\{\mathbf{u}_p^{(n)}, n = 1, \ldots, N\}$ is an iterative procedure due to the nature of R-UMLDA, like in other multilinear learning algorithms [1], [2], [15], [42]–[44], initial estimations for the projection vectors $\{\mathbf{u}_p^{(n)}\}$ are necessary. However, there is no guidance from either the algorithm or the data on the best initialization that could result in the best separation of the classes in the feature space. Thus, the determination of the optimal initialization in R-UMLDA is still an open problem, as in most iterative algorithms including other multilinear learning algorithms [2], [15], [18], [19]. In this work, we empirically study two simple and commonly used initialization methods [17]: uniform initialization and random initialization [18], [19], which do not depend on the data. In the uniform initialization, all n-mode projection vectors are initialized to have unit length and the same value along the I_n dimensions in n-mode, which is equivalent to the all ones vector $\mathbf{1}$ with proper normalization. In random initialization, each element of the n-mode projection vectors is drawn from a zero-mean uniform distribution between $[-0.5, 0.5]$ and the initialized projection vectors are normalized to have unit length. Our empirical studies in Section V-C indicate that the results of R-UMLDA are affected by initialization, and the uniform initialization gives better results.

The mode ordering (the innermost for loop in Fig. 3, indexed by n) in computing the projection vectors, named the projection order in this work, affects the solution as well. Similar to initialization, there is no way to determine the optimal projection order and it is considered to be an open problem too. Empirical studies on the effects of the projection order indicate that with all the other algorithm settings fixed, altering the projection order does result in some performance differences, but there is no guidance from either the data or the algorithm on what projection order is the best in the iteration. Therefore, we have no preference on a particular projection order and in practice, we solve the projection vectors sequentially (from 1-mode to N-mode), as in other multilinear algorithms [1], [2], [15], [19].

Remark 1: Although we are not able to determine the optimal initialization and the optimal projection order, the aggregation scheme suggested in Section IV reduces the significance of their optimal determination.

As seen from Fig. 3, the termination criterion can be simply set to a maximum number of iterations K or it can be set by examining the convergence of the projection vectors: $\mathrm{dist}(\mathbf{u}_{p(k)}^{(n)}, \mathbf{u}_{p(k-1)}^{(n)}) < \epsilon$, where ϵ is a user-defined small number threshold (e.g., $\epsilon = 10^{-3}$), and this distance is defined as

$$\mathrm{dist}\left(\mathbf{u}_{p(k)}^{(n)}, \mathbf{u}_{p(k-1)}^{(n)}\right) = \min\left(\|\mathbf{u}_{p(k)}^{(n)} + \mathbf{u}_{p(k-1)}^{(n)}\|, \|\mathbf{u}_{p(k)}^{(n)} - \mathbf{u}_{p(k-1)}^{(n)}\|\right) \quad (15)$$

since eigenvectors are unique up to sign. As to be shown in Section V-C, the recognition performance increases slowly after the first few iterations. Therefore, the iteration can be terminated by setting K in practice for convenience, especially when computational cost is a concern.

D. Computational Aspects of UMLDA

Next, the computational complexity and memory requirements of UMLDA are analyzed, in a similar way as in [10]. In the analysis, it is assumed that $I_1 = I_2 = \cdots = I_N = (\prod_{n=1}^{N} I_n)^{1/n} = I$ for simplicity.

From a computational complexity point of view, the most demanding steps involve the calculations of the projection $\tilde{\mathbf{y}}_{m_p}^{(n)}$, the computation of $\tilde{\mathbf{S}}_{B_p}^{(n)}$, $\tilde{\mathbf{S}}_{W_p}^{(n)}$, and $\mathbf{R}_p^{(n)}$, and the determination of the leading eigenvector of $(\tilde{\mathbf{S}}_{W_p}^{(n)})^{-1}\mathbf{R}_p^{(n)}\tilde{\mathbf{S}}_{B_p}^{(n)}$. The complexity of calculating $\tilde{\mathbf{y}}_{m_p}^{(n)}$ for $m = 1, \ldots, M$, $\tilde{\mathbf{S}}_{B_p}^{(n)}$, and $\tilde{\mathbf{S}}_{W_p}^{(n)}$ are in order of $O(M \cdot \sum_{n=2}^{N} I^n)$, $O(C \cdot I^2)$, and $O(M \cdot I^2)$, respectively. The computation of $\mathbf{R}_p^{(n)}$ is in order of

$$\begin{aligned} &O(I \cdot M \cdot (p-1) + I^3 + 2 \cdot (p-1) \\ &\quad \cdot I^2 + (p-1)^3 + 2 \cdot I \cdot (p-1)^2) \\ &\quad = O(I^3 + (p-1) \cdot [I \cdot M + 2 \cdot I^2 + (p-1)^2 \\ &\qquad + 2 \cdot I \cdot (p-1)]). \end{aligned} \quad (16)$$

Last, the computation of $(\tilde{\mathbf{S}}_{W_p}^{(n)})^{-1}\mathbf{R}_p^{(n)}\tilde{\mathbf{S}}_{B_p}^{(n)}$ and its eigendecomposition[6] can be completed in $O(2 \cdot I^3)$ and $O(I^3)$, respectively. Therefore, the overall computational complexity per mode n for a single iteration k during step p is

$$O\left(M \sum_{n=2}^{N} I^n + (C+M)I^2 + (p-1)\left[I \cdot M + 2I^2 + (p-1)^2 + 2I(p-1)\right] + 4I^3\right). \quad (17)$$

With respect to the memory requirement, as in MPCA [10], the respective computation can be done incrementally by reading $\mathcal{X}_m$ sequentially. Hence, the memory needed for the UMLDA algorithm can be as low as $O(I^N)$ except for $N = 1$. It should be noted at this point that sequential reading may lead to higher input/output (I/O) cost.

From the reported analysis above, it can be seen that as a sequential iterative solution, UMLDA may result in a higher computational and I/O cost. Nonetheless, since the operations in solving the UMLDA projection are usually performed offline, only during the training phase, additional computational and I/O cost due to iterations and sequential processing do not constitute a comparative disadvantage. During testing, the extraction (projection) of features from a test sample is a simple linear operation with a computational complexity similar to linear subspace projection algorithms.

E. Connections to DATER, GTDA, TR1DA, and LDA

Detailed discussions on the relationships between UMLDA, DATER, GTDA, TR1DA, and LDA are presented in [17], which are summarized in Appendix II for completeness. Briefly, the UMLDA is an MLDA variant that maximizes the scatter ratio through a TVP. LDA is the special case of UMLDA with $N = 1$. DATER is an MLDA variant also based on scatter ratio but it solves a TTP rather than TVP. GTDA is an MLDA variant that maximizes the scatter difference through a TTP. TR1DA also solves for TVP as UMLDA, however, it maximizes scatter difference and it is a heuristic approach with residue calculation, originally proposed for tensor approximation.

IV. Aggregation of R-UMLDA Recognizers

This section proposes the aggregation of a number of differently initialized and regularized UMLDA recognizers for enhanced performance, which is motivated from two properties of the basic R-UMLDA recognizer in Fig. 2. On one hand, as to be shown in Section V-C, the number of useful discriminative features that can be extracted by a single R-UMLDA is limited, partly due to the fact that EMPs to be solved in R-UMLDA correspond to very constrained situations in the linear case. On the other hand, since the R-UMLDA is affected by initialization and regularization, which cannot be optimally determined, different initialization or regularization could result in different discriminative features (also see Section V-C). From the generalization theory explaining the success of random subspace method [45], bagging and boosting [46], [47], [48], the sensitivity of the R-UMLDA to initialization and regularization suggests that R-UMLDA is not a very stable learner (feature extractor) and it is good for ensemble-based learning. Therefore, we propose the aggregation of several differently initialized and regularized UMLDA feature extractors to get the regularized UMLDA with

[6]UMLDA needs only the largest eigenvalue and the corresponding eigenvector, so more efficient computational methods may be applied in practice.

Input: A set of zero-mean tensor samples $\{\mathcal{X}_m \in \mathbb{R}^{I_1 \times I_2 \times \ldots \times I_N}, m = 1, ..., M\}$ with class labels $\mathbf{c} \in \mathbb{R}^M$, a test tensor sample $\mathcal{X}$, the desired feature vector length P, the R-UMLDA feature extractor (Fig. 3), the maximum number of iterations K, the number of R-UMLDA to be aggregated A.

Output: The class label for $\mathcal{X}$.

R-UMLDA-A algorithm:

Step 1. Feature extraction

- For $a = 1 : A$
  - Obtain the ath TVP $\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}_{p=1_{(a)}}^{P}$ from the ath R-UMLDA (Fig. 3) with the input: $\{\mathcal{X}_m\}$, P, K, γ_a, using random or uniform initialization.
  - Project $\{\mathcal{X}_m\}$ and $\mathcal{X}$ to $\{\mathbf{y}_{m_{(a)}}\}$ and $\mathbf{y}_{(a)}$, respectively, using $\{\mathbf{u}_p^{(n)^T}, n = 1, ..., N\}_{p=1_{(a)}}^{P}$.

Step 2. Aggregation at the matching score level for classification

- For $a = 1 : A$
  - For $c = 1 : C$
    - Obtain the nearest-neighbor distance $d(\mathcal{X}, c, a)$.
  - Normalize $d(\mathcal{X}, c, a)$ to $[0, 1]$ to get $\tilde{d}(\mathcal{X}, c, a)$.
- Obtain the aggregated distance $d(\mathcal{X}, c)$.
- **Output** $c^* = \arg\min_c d(\mathcal{X}, c)$ as the class label for the test sample.

Fig. 4. Pseudocode implementation of the R-UMLDA-A algorithm for tensor object recognition.

aggregation (R-UMLDA-A) recognition system so that multiple R-UMLDA recognizers can work together to achieve better performance on tensor object recognition.

Remark 2: Different projection order also could result in different features so R-UMLDA with different projection orders could be aggregated as well. However, since the effects of different projection orders are similar to those of different initializations and the number of possible projection orders (which is $N!$) is much less than the number of possible initializations (which is infinite), we fix the projection order and vary the initialization and regularization only in this work.

There are various ways to combine (or fuse) several extracted features, including the feature level fusion [49], fusion at the matching score level [50], [51], and more advanced ensemble-based learning such as boosting [46], [52], [53]. For similar argument of the choice of a simple classifier, the simple sum rule in combining matching scores is used in this work since the focus here is on the feature extraction. Although more sophisticated method such as boosting is expected to achieve better results, the investigation of alternative combination methods, such as other combination rules and feature-level fusion, are beyond the scope of this paper and will be the topic of a forthcoming paper.

Since high diversity of the learners to be combined is preferred in ensemble-based learning [52], we choose to use both uniform and random initializations in R-UMLDA-A for more diversity. Thus, although we are not able to determine the best initialization, we aggregate several R-UMLDA with different initializations to make complementary discriminative features working together to separate classes better. Furthermore, to introduce even more diversity and alleviate the problem of regularization parameter selection at the same time, we propose to sample the regularization parameter γ_a from an interval $[10^{-7}, 10^{-2}]$, which is empirically chosen to cover a wide range of γ, uniformly in log scale so that each feature extractor is differently regularized, where $a = 1, \ldots, A$ is the index of the individual R-UMLDA feature extractor and A is the number of R-UMLDA feature extractors to be aggregated.

Fig. 4 provides the pseudocode implementation for the R-UMLDA-A for tensor object recognition. The input training samples $\{\mathcal{X}_m\}$ are fed into A differently initialized and regularized UMLDA feature extractors described in Fig. 3 with parameters P, K, and γ_a to obtain a set of A TVPs $\{\mathbf{u}_p^{(n)^T}, n = 1, \ldots, N\}_{p=1_{(a)}}^{P}$, $a = 1, \ldots, A$. The training samples $\{\mathcal{X}_m\}$ are then projected to R-UMLDA feature vectors $\{\mathbf{y}_{m_{(a)}}\}$ using the obtained TVPs. To classify a test sample $\mathcal{X}$, it is projected to A feature vectors $\{\mathbf{y}_{(a)}\}$ using the A TVPs first. Next, for the ath R-UMLDA feature extractor, we calculate the nearest-neighbor distance of the test sample $\mathcal{X}$ to each candidate class c as

$$d(\mathcal{X}, c, a) = \min_{m, c_m = c} ||\mathbf{y}_{(a)} - \mathbf{y}_{m_{(a)}}||. \tag{18}$$

The range of $d(\mathcal{X}, c, a)$ is then matched to the interval $[0, 1]$ as

$$\tilde{d}(\mathcal{X}, c, a) = \frac{d(\mathcal{X}, c, a) - \min_c d(\mathcal{X}, c, a)}{\max_c d(\mathcal{X}, c, a) - \min_c d(\mathcal{X}, c, a)}. \tag{19}$$

Finally, the aggregated nearest-neighbor distance is obtained employing the simple sum rule as

$$d(\mathcal{X}, c) = \sum_{a=1}^{A} \tilde{d}(\mathcal{X}, c, a) \tag{20}$$

and the test sample $\mathcal{X}$ is assigned the label: $c^* = \arg\min_c d(\mathcal{X}, c)$.

TABLE II
CHARACTERISTICS OF THE GAIT DATA FROM THE USF GAIT CHALLENGE DATA SET

Gait data set	Gallery(GAR)	Probe A(GAL)	Probe B(GBR)	Probe C(GBL)
Number of sequences (samples)	71 (731)	71 (727)	41 (423)	41 (420)
Difference from the gallery	-	View	Shoe	Shoe, view

TABLE III
LIST OF ALGORITHMS TO BE COMPARED

Acronym	Full name	Linear/Multilinear	Reference
Bayes	Maximum likelihood version of the Bayesian solution	Linear	[56]
LDA	linear discriminant analysis	Linear	[32]
ULDA	uncorrelated linear discriminant analysis	Linear	[33]
R-JD-LDA	regularized version of the revised direct LDA	Linear	[30], [57]
MPCA	multilinear principal component analysis	Multilinear	[10]
DATER	discriminant analysis with tensor representation	Multilinear	[2]
GTDA	general tensor discriminant analysis	Multilinear	[15]
TR1DA	tensor rank-one discriminant analysis	Multilinear	[18], [19]

V. EXPERIMENTAL EVALUATION

In this section, a number of experiments are carried out on two biometric applications in support of the following two objectives:

1) investigate the various properties of the R-UMLDA algorithm;
2) evaluate the R-UMLDA and R-UMLDA-A algorithms on two tensor object recognition problems, face recognition (FR) and gait recognition (GR), by comparing their performance with that of competing multilinear learning algorithms as well as linear learning algorithms.

Before presenting the experimental results, the experimental data and algorithms to be compared are described first.

A. Experimental Data

Three popular public databases are used in the experiments: the Pose, Illumination, and Expression (PIE) database from Carnegie Mellon University (CMU) [54], the Facial Recognition Technology (FERET) database [55] and the HumanID gait challenge data set version 1.7 (V1.7) from the University of South Florida (USF).

The CMU PIE database contains 68 individuals with face images captured under varying pose, illumination, and expression. We choose the seven poses (C05, C07, C09, C27, C29, C37, C11) with at most 45° of pose variation, under the 21 illumination conditions (02 to 22). Thus, there are about 147 (7×21) samples per subject and there are a total number of 9987 face images (with nine faces missing). This face database has a large number of samples for each subject, therefore, it is used to study the properties of the proposed algorithm and the FR performance under varying number of training samples per subject, denoted by L.

The FERET database is a standard testing database for FR performance evaluation, including 14 126 images from 1199 individuals with views ranging from frontal to left and right profiles. The common practice is to use portions of the database for specific studies. Here, we select a subset composed of those subjects with each subject having at least six images with at most 45° of pose variation, resulting in 2803 face images from 335 subjects. The studies on the FR performance under varying number of subjects C are carried out on this face database since there are a large number of subjects available. Face images from the PIE and FERET databases are manually aligned, cropped, and normalized to 32×32 pixels, with 256 gray levels per pixel.

The USF gait challenge data set V1.7 consists of 452 sequences from 74 subjects walking in elliptical paths in front of the camera, with two viewpoints (left or right), two shoe types (A or B), and two surface types (grass or concrete). Here, we choose those sequences on grass surface only: the gallery set, and the probe sets A, B, and C, with detailed information listed in Table II. The capturing condition for each set is summarized in the parentheses after the set name in the table, where G, A, B, L, and R stand for grass surface, shoe type A, shoe type B, left view, and right view, respectively. Each set has only one sequence for a subject. Subjects are unique in the gallery and each probe set and there are no common sequences between the gallery set and any of the probe sets. In addition, all the probe sets are distinct. These gait data sets are employed to demonstrate the performance on third-order tensors since gait silhouette sequences are naturally 3-D data [10]. We follow the procedures in [10] to get gait samples from gait silhouette sequences and each gait sample is resized to a third-order tensor of $32 \times 22 \times 10$. The number of samples for each set is indicated in the parentheses following the number of sequences in Table II.

B. Performance Comparison Design

In the FR and GR experiments, we compare the performance of the proposed algorithms against four multilinear learning algorithms and four linear learning algorithms listed in Table III,[7] where the LDA algorithm takes the Fisherface approach [32].

[7]Note that the ULDA compared here is different from the ULDA in [28]. Therefore, it is different from the classical LDA.

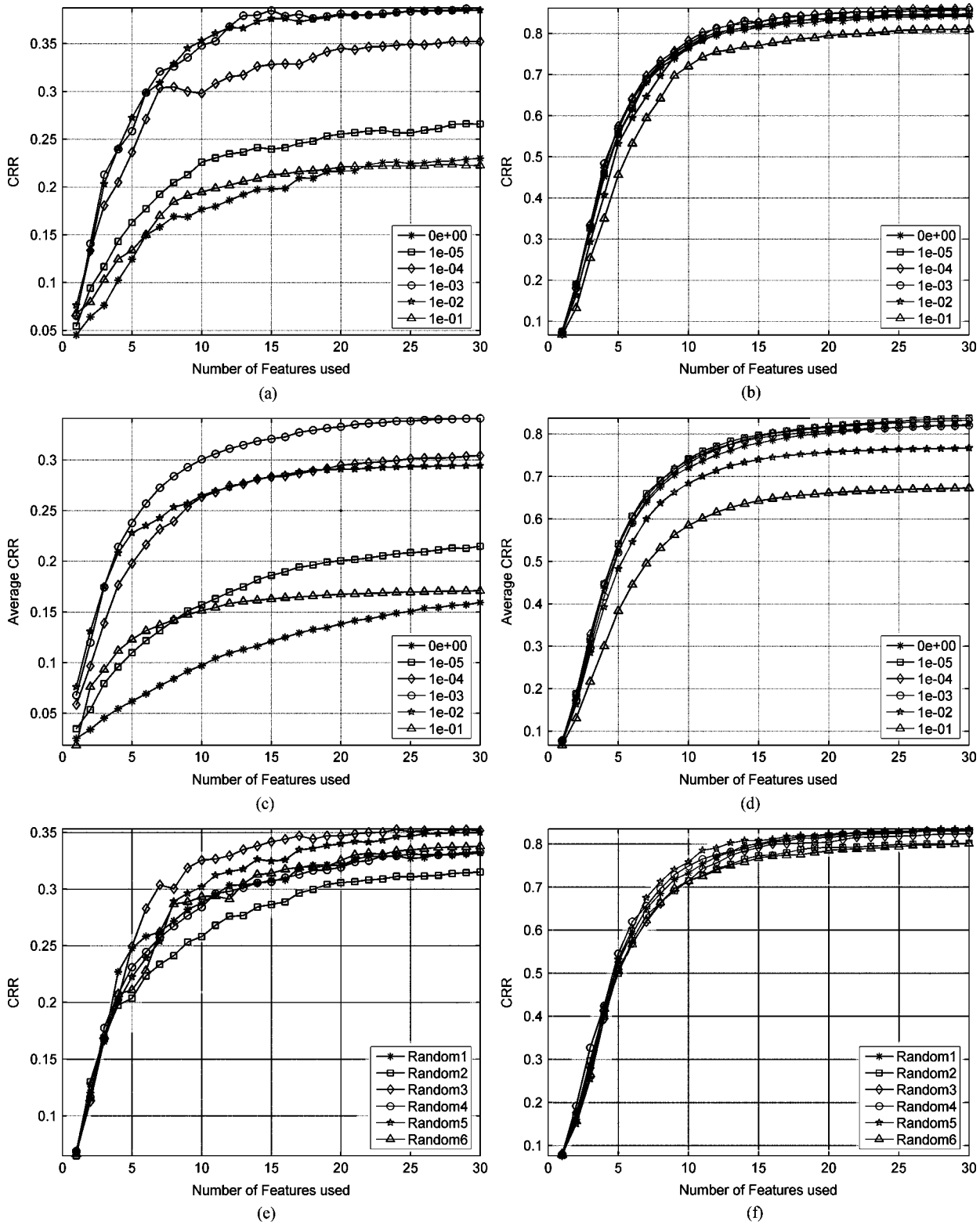

Fig. 5. Illustration of the effects of initialization and regularization on recognition performance. Uniform initialization with various γs for (a) $L = 2$ and (b) $L = 20$; random initialization with various γs averaged over 20 repetitions for (c) $L = 2$ and (d) $L = 20$; six repetitions of random initialization with $\gamma = 10^{-3}$ for (e) $L = 2$ and (f) $L = 20$.

For classification of extracted features, we use the NNC with Euclidean distance measure except for the Bayesian solution, which uses the NNC with the Mahalanobis distance [56]. The MPCA, DATER, and GTDA algorithms produce features in tensor representation, which cannot be handled directly by the selected classifier. Since from [17], the commonly used tensor distance measure, the Frobenius norm, is equivalent to the Euclidean distance between vectorized representations, the tensor

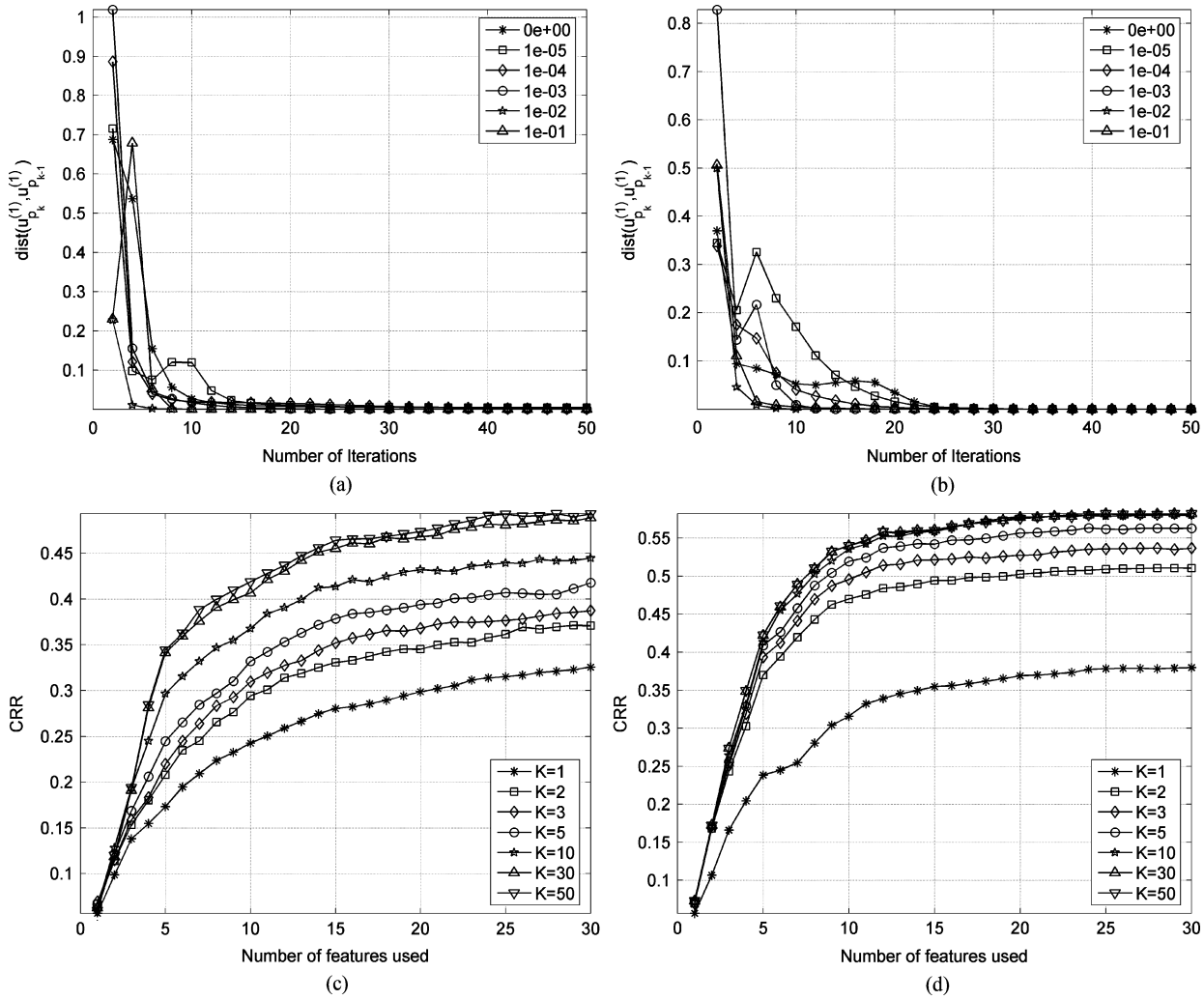

Fig. 6. Illustration of the convergence for $L = 5$: the evolution of $\mathrm{dist}(\mathbf{u}^{(1)}_{p_{(k)}}, \mathbf{u}^{(1)}_{p_{(k-1)}})$ for various γs over 50 iterations for (a) $p = 1$ and (b) $p = 8$; the CRRs for various Ks (the maximum number of iterations) for (c) $\gamma = 0$ and (d) $\gamma = 10^{-3}$.

features from MPCA, DATER, and GTDA are rearranged to vectors for direct comparison. They obtain the highest dimension projection ($P_n = I_n$ for $n = 1, \ldots, N$) first and then the TTP is viewed as $\prod_{n=1}^{N} I_n$ EMPs. The discriminability of each such EMP is calculated on the training set and the EMPs are arranged in descending discriminability so that a feature vector is obtained, as in [10].

In the experiments, for fair comparison and computational concerns, we set the number of iterations in the four MLDA algorithms to be 10, unless otherwise stated. For MPCA, DATER, GTDA, and TR1DA, up to 600 features were tested. The maximum number of features tested for LDA and ULDA is $C - 1$. For the Bayesian solution, the maximum number of features tested is $\min\{M - C, 600\}$. For the TR1DA algorithm, we tested several values of the tuning parameter ζ for each L, and the best one for each L was used: $\zeta = 2$ for $L \leq 7$, $\zeta = 0.8$ for $8 \leq L \leq 15$, $\zeta = 0.6$ for $L \geq 16$. For R-JD-LDA, the default maximum number ($\approx 0.8 \cdot (C - 1)$) of features and a regularization parameter of 0.001 originally suggested by the authors of [30] and [57] are used.

For the recognition experiments of only one R-UMLDA, uniform initialization is used and we empirically set $\gamma = 10^{-3}$, with up to 30 features tested. For R-UMLDA-A, up to 20 differently initialized and regularized versions of UMLDA feature extractors are combined with each producing up to 30 features, also resulting in a total number of 600 features. Uniform initialization is used for $a = 1, 5, 9, 13, 17$ with corresponding $\gamma_a = 10^{-2}, 10^{-3}, 10^{-4}, 10^{-5}, 10^{-6}$, and random initialization is used for the rest values of a. In computing the matrix inverse of $(\mathbf{G}_{p-1}^T \tilde{\mathbf{Y}}_p^{(n^*)^T} \tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}} \tilde{\mathbf{Y}}_p^{(n^*)} \mathbf{G}_{p-1})$ in (13), a small term $(\kappa \cdot \mathbf{I}_{p-1})$ is added, where $\kappa = 10^{-3}$, in order to get better conditioned matrix for the inverse computation.

The recognition performance is measured in correct recognition rate (CRR). The best recognition results reported are obtained by varying the number of features used and the number of R-UMLDA recognizers aggregated for the R-UMLDA-A algorithm. For all the other algorithms, the best results are obtained by varying the number of features used. For fair comparison, there is no further fine tuning of other parameters (such as the regularization parameter) for optimal performance on the testing data, including the proposed method. For better viewing, the top two recognition results in each experiment are shown in bold in tables. In the comparative evaluation of recognition performance in Figs. 8–10, for the horizontal axis which indicates the number of features used for recognition, log scale has been

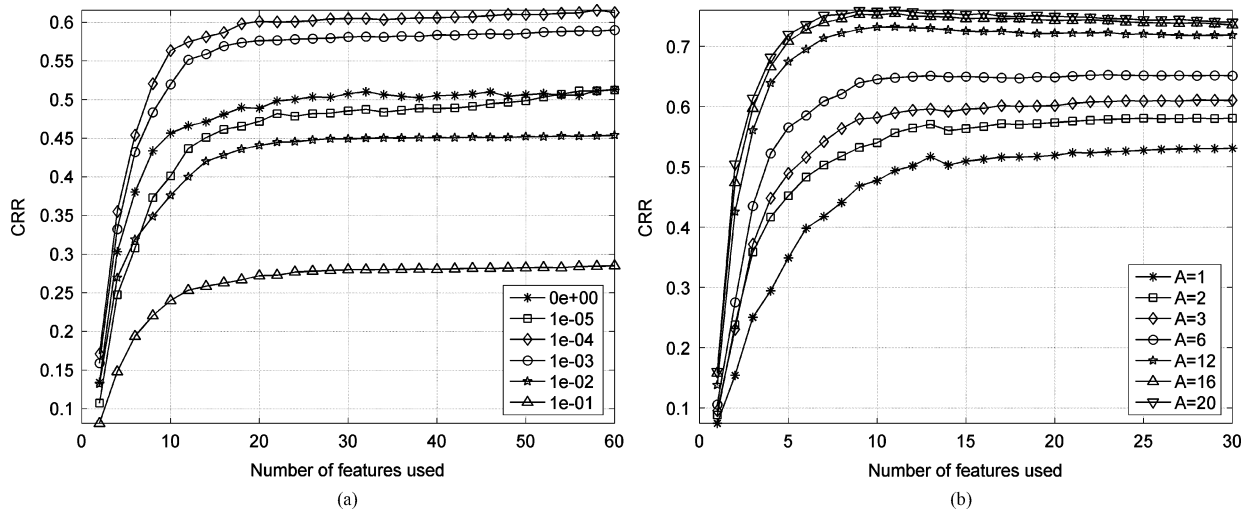

Fig. 7. Demonstration of (a) the recognition performance for $L = 5$ as P increases for various γs, and (b) the effectiveness of aggregation.

used in order to highlight the comparative performance in low-dimensional subspaces while showcasing the best overall performance achieved.

C. Empirical Studies of the R-UMLDA Properties on the PIE Database

First, the following properties of R-UMLDA are studied on the PIE database: the effects of initialization and regularization, the convergence, the number of useful features, and the effects of aggregation. These experiments are performed on one random split of the database into training and testing samples.

1) The Effects of Initialization and Regularization: Fig. 5 illustrates the effects of initialization and regularization on two FR experiments: one with $L = 2$ and one with $L = 20$, corresponding to the SSS scenario and the scenario when a large number of samples (per subject) are available for training. The CRRs for various γs are depicted in Fig. 5(a) $(L = 2)$ and Fig.5(b) $(L = 20)$ for the uniform initialization, and in Fig. 5(c) $(L = 2)$ and Fig.5(d) $(L = 20)$ for the random initialization (averaged over 20 repeated trials). Fig. 5(e) and (f) shows the plots for the CRRs from six repetitions of the random initialization with $\gamma = 10^{-3}$. They demonstrate that the recognition results are affected by initialization and different initialization results in different results. By comparing Fig. 5(f) against Fig. 5(e), it can be seen that the sensitivity to initialization is smaller for a larger L. Furthermore, by comparing Fig. 5(a) against Fig. 5(c) and Fig. 5(b) against Fig. 5(d), we observe that the uniform initialization outperforms the random initialization for both a small L and a large L. Therefore, we use the uniform initialization when only one R-UMLDA feature extractor is employed. In the following discussions, we show the convergence and the number of useful features using the uniform initialization.

In addition, the effects of regularization are also observed in Fig. 5(a)–(d). For a small L, UMLDA with a strong regularization (larger γ) can outperform that without regularization $(\gamma = 0)$, while for a large L, a too strong regularization may results in poorer performance, as observed in other regularization algorithms [30].

2) Convergence: The convergence is illustrated in Fig. 6. Fig. 6(a) and (b) depicts two examples of the evolution of $\mathrm{dist}(\mathbf{u}^{(1)}_{p(k)}, \mathbf{u}^{(1)}_{p(k-1)})$ for $p = 1$ and $p = 8$, with various γs, up to 50 iterations. As seen in the figure, in the worst scenarios, the projection vector converges around $k = 15$ for $p = 1$ and around $k = 30$ for $p = 8$. In addition, a stronger regularization (larger γ) is more likely to result in faster convergence. Furthermore, the recognition performance is examined for various Ks, as shown in Fig. 6(c) and (d) with $L = 5$ for $\gamma = 0$ and $\gamma = 10^{-3}$, respectively. It indicates that the first few iterations improve the recognition performance the most, and more iterations afterwards give slow improvement in the recognition rate, especially for a larger γ. Therefore, we set K to a fixed number $K = 10$ to terminate the iteration in practice. When computational efficiency is important, K can be further reduced to improve processing speed, while sacrificing some recognition performance.

3) The Number of Useful Features and the Effects of Aggregation: The R-UMLDA is limited in the number of extracted features (P) useful for recognition, as depicted in Fig. 7(a), where the CRRs are shown for up to 60 features for $L = 5$ and with various γs. In particular, the first few features are very powerful, while beyond a certain number (e.g., 20), the performance varies very slowly with an increased P. Fortunately, from the study of the effects of initialization and regularization, we find that different initialization or regularization produces different results (Fig. 5). Thus, the proposed aggregation scheme makes use of this property and combines differently initialized and regularized R-UMLDA recognizers to achieve enhanced results. At the same time, the problem of regularization parameter selection is alleviated. The results of aggregation are shown in Fig. 7(b) for $L = 5$ and up to 20 R-UMLDA recognizers to be combined, by the R-UMLDA-A described in Section V-B. The figure demonstrates that the aggregation is an effective procedure and there are indeed complementary discriminative information from differently initialized and regularized R-UMLDA recognizers.

D. Face Recognition Results

In FR experiments, face images are input directly as second-order tensors to the multilinear algorithms, while for the linear algorithms, they are vectorized to 1024×1 vectors as input. For each subject in an FR experiment, L samples are randomly selected for training and the rest are used for testing. We report

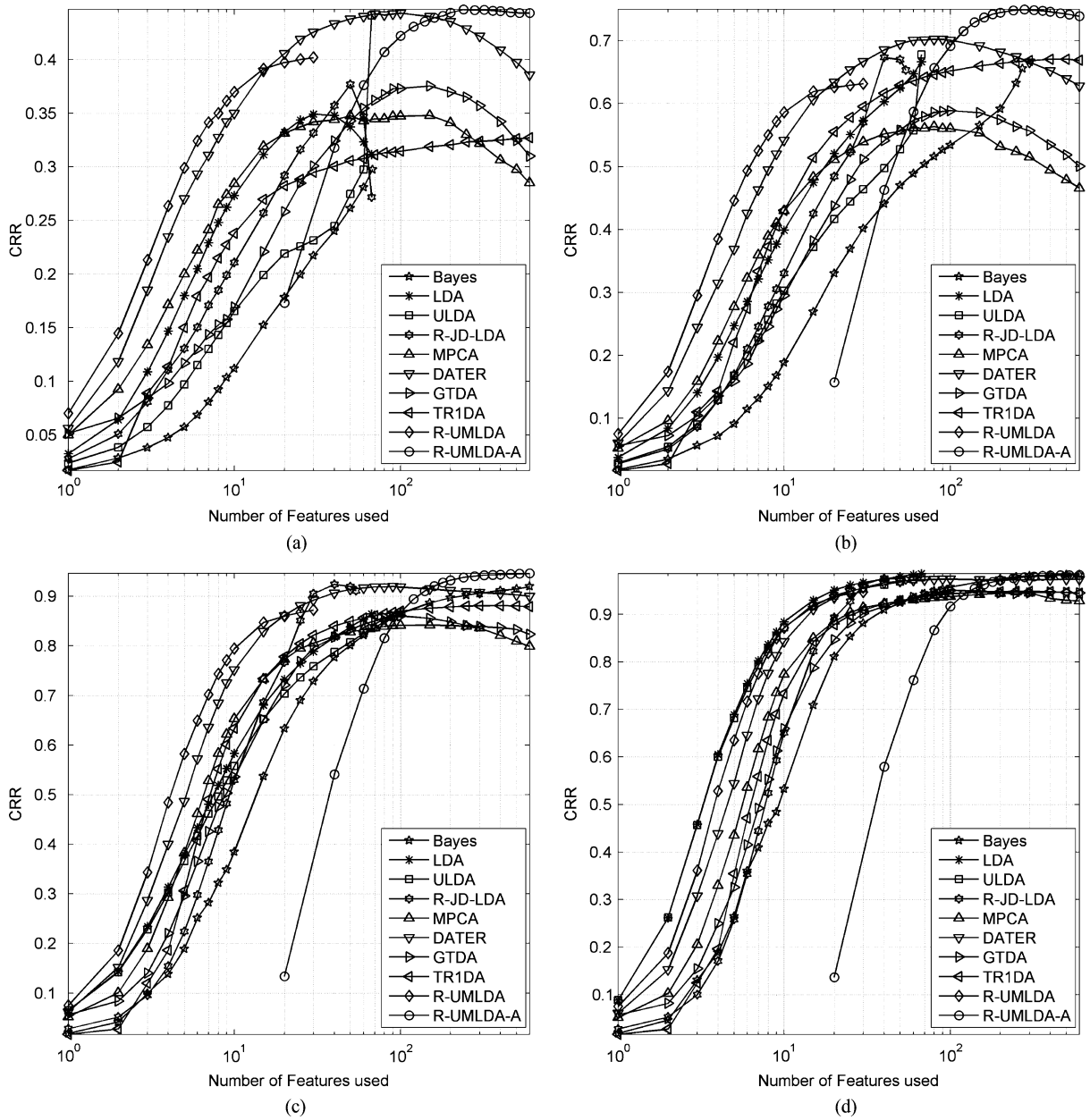

Fig. 8. Face recognition results on the PIE database: correct recognition rate against the number of features used for (a) $L = 2$, (b) $L = 5$, (c) $L = 20$, and (d)$L = 40$.

the results, including the mean and standard deviation (Std), averaged over 20 random splits.

1) FR Results on the PIE Database: In order to study the recognition performance with different Ls, we perform nine FR experiments on the PIE database with $L = 2, 3, 4, 5, 6, 8, 10, 20, 40$. The top CRRs are listed in Table IV, where our R-UMLDA-A performs the best except for $L = 40$. The detailed results for $L = 2, 5, 20$, and 40 are depicted in Fig. 8. From the figure, the first few features (around ten) extracted by the R-UMLDA are the most powerful features in recognition in all cases except when $L = 40$, where the LDA and ULDA features are the most discriminative ones.

In addition, it should be noted that in each experiment, if we tune the regularization parameter for R-JD-LDA and R-UMLDA, and the range of γ for R-UMLDA-A, improved performance can be obtained since stronger regularization results in better performance for a small L and weaker regularization is better for a larger L [30]. Nonetheless, with fixed range of γ, our R-UMLDA-A still outperforms all the other algorithms for L ranging from 2 to 20.

2) FR Results on the FERET Database: It is argued in [52] that the learning capacity of any LDA-like algorithm is directly proportional to L, and reciprocally proportional to C. Thus, to evaluate the recognition performance with different Cs, we carry out four experiments on the FERET database with $C = 80, 160, 240, 320$ and fix $L = 4$ so that no more than half of the face images are used for training. The numbers of training and testing faces for each experiment are detailed in Table V. Table VI lists the top CRRs, where R-UMLDA-A outperforms all the other methods in all cases. Moreover, we can observe that

TABLE IV
FACE RECOGNITION RESULTS ON PIE DATABASE: THE TOP CRRS (MEAN ± STD%)

L	Bayes	LDA	ULDA	R-JD-LDA	MPCA	DATER	GTDA	TR1DA	R-UMLDA	R-UMLDA-A
2	29.7±1.2	35.0±1.6	44.1±1.3	38.0±1.9	34.9±4.3	**44.3**±2.2	37.8±2.5	32.7±2.6	40.2±1.8	**44.7**±1.8
3	45.4±1.1	47.6±1.8	55.7±1.3	53.5±1.4	46.0±2.4	**58.3**±1.8	48.0±2.2	51.1±1.3	50.8±1.7	**58.4**±1.7
4	57.3±1.4	59.0±2.0	63.4±1.4	62.0±1.6	52.2±1.7	**65.9**±1.8	54.4± 1.7	61.9±1.9	58.2±1.3	**69.0**±2.1
5	65.6±1.1	66.7±1.4	67.8±1.1	67.9±1.3	56.4±1.8	**70.3**±1.4	59.0± 1.9	67.1±1.6	63.2±1.5	**74.9**±1.5
6	71.6±1.1	71.2±1.3	70.7±0.9	72.0±1.3	59.9±1.6	**74.3**±1.3	62.7± 1.3	69.7±1.5	67.0±1.1	**78.7**±1.4
8	79.0±1.2	76.6±1.1	74.5±1.0	78.5±1.1	66.6±1.0	**79.9**±1.0	69.2± 1.4	74.6±1.1	72.5±1.0	**84.7**±1.0
10	83.3±1.0	79.3±1.2	75.2±1.1	82.8±1.2	71.3±0.9	**83.5**±0.9	74.0±1.0	79.4±1.4	76.7±1.4	**87.8**±1.2
20	92.0±0.8	86.4±1.1	83.6±1.0	**92.5**±0.5	84.2±0.7	92.0±0.4	86.2±0.9	88.1±0.7	87.2±0.7	**94.6**±0.5
40	**98.4**±0.3	**98.6**±0.2	97.7±0.3	97.7±0.3	94.8±0.5	97.4±0.3	95.2±0.4	94.8±0.4	94.7±0.4	98.3±0.3

TABLE V
DETAILS OF THE FOUR EXPERIMENTS ON THE FERET DATABASE

C	number of training faces	number of testing faces
80	320	825
160	640	1113
240	960	1273
320	1280	1433

the recognition performance of R-UMLDA-A and R-JD-LDA are just slightly affected by C. In contrast, LDA and ULDA are affected significantly by C. Detailed recognition results are shown in Fig. 9, where in all cases, the first few (around ten) features extracted by R-UMLDA are the most discriminative ones again.

E. Gait Recognition Results

After two sets of experiments to evaluate the performance on 2-D face images under varying L and C, we test the performance on 3-D gait data. In GR experiments, gait samples are input directly as third-order tensors to the multilinear algorithms, while for the linear algorithms, they are vectorized to 7040×1 vectors as input. We follow the standard testing procedures in GR, where the gallery set is used for training and the probe sets (A, B, and C) are used for testing. Since R-UMLDA-A involves random initialization, we report the results (mean and standard deviation) averaged over 20 repeated experiments. In addition to the CRRs of the gait samples, the CRRs of gait sequences are also reported. The calculation of matching scores between two gait sequences follows that in [10].

Tables VII and VIII present the top CRRs for individual gait samples and gait sequences, respectively, for probes A, B, and C and their average. R-UMLDA-A and R-JD-LDA achieve the best performance on recognizing individual gait samples, indicating that regularization is indeed effective for the challenging GR problem as well. Regarding the performance on recognizing gait sequences, R-UMLDA-A achieves the best results on average, showing that the proposed multilinear solution is indeed more powerful than other solutions in recognizing tensorial signals. Fig. 10(a) and (b) plot the detailed CRRs averaged over the three probes for individual gait samples and gait sequences, respectively. Fig. 10(a) shows that the first few features (around 20) produced by R-UMLDA are again the most discriminative ones in classifying gait samples, while Fig. 10(b) illustrates that R-UMLDA still has the best performance when recognizing gait sequences using 3–20 features only. It is also noted that the Bayesian solution, which is originally proposed for FR, is not suitable for GR and its recognition performance deteriorates when more than 70 features are used.

F. Discussions

We have performed a large number of experiments on face and gait recognition to evaluate our proposed algorithms. From the results presented above, the following observations are made.

1) The first few features extracted by the proposed R-UMLDA, obtained with fixed parameters, are the most discriminative ones in various scenarios, except in the FR experiment on the PIE database with $L = 40$. This demonstrates that extracting uncorrelated features directly from tensorial signals using R-UMLDA is indeed beneficial for recognition, especially in the low-dimensional space under the SSS scenario, although the overall recognition performance is limited by the number of useful features.
2) The proposed R-UMLDA-A overcomes the limitation of R-UMLDA in the number of useful features through aggregating several differently initialized and regularized UMLDA feature extractors. This scheme has not only enhanced the recognition performance but also alleviated the regularization parameter selection problem. Without tuning the regularization parameter, R-UMLDA-A achieves the best overall performance in all the FR and GR experiments except in FR with $L = 40$ on the PIE database, where it is slightly outperformed by LDA and the Bayesian solution. This comparative evaluation demonstrates that R-UMLDA-A is a robust and effective recognition algorithm for tensor objects.
3) Although random initialization is employed in R-UMLDA-A, from the standard deviations based on 20 repeated trials reported in Tables IV and VI–VIII, the recognition results obtained by R-UMLDA-A have low variances, showing that it is competitive in terms of stability as well.

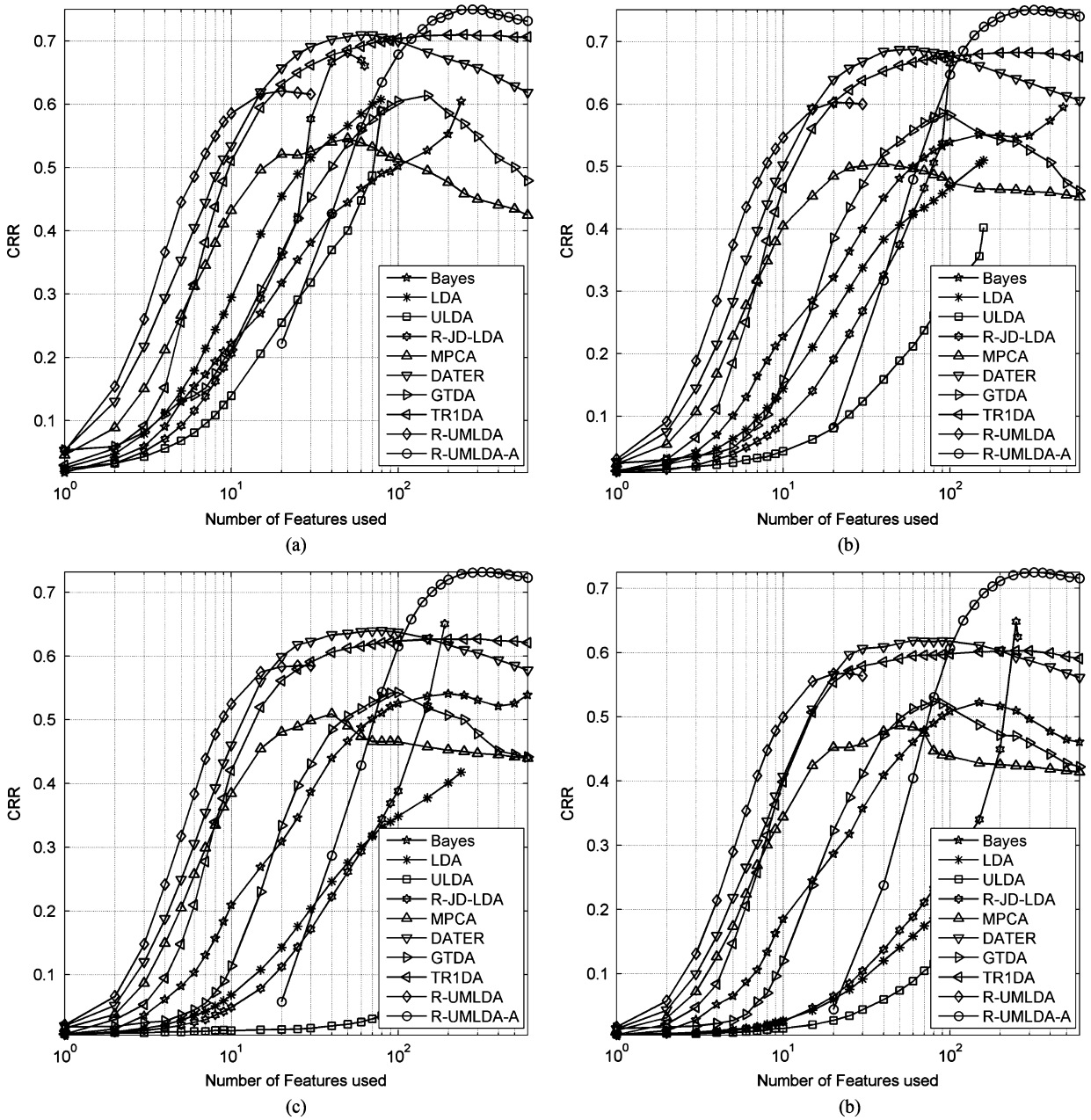

Fig. 9. Face recognition results on the FERET database: correct recognition rate against the number of features used for (a) $C = 80$, (b) $C = 160$, (c) $C = 240$ and (d) $C = 320$.

4) Moreover, the best recognition performance of R-UMLDA-A is reached with approximately the same number of features in all the FR and GR experiments. The addition of extra features only marginally alters its recognition performance. It should be noted at this point that unlike R-UMLDA-A, other algorithms exhibit poorer stability characteristics in the sense that variation in the number of features used may lead to large performance variation. As case in point, note that the performance of the Bayesian solution starts to decrease significantly when more than 150 features are used in the FR experiments on the FERET database with $C = 320$, and when more than 70 features are used in the GR experiments on the USF database. LDA and ULDA significantly underperform the other algorithms on the FERET database with $C = 320$. On the PIE database with $L = 2$ and on the FERET database with $C = 320$, which are both very difficult learning problems [52], R-JD-LDA shows abrupt decrease in recognition performance when the last few features are included. The recognition rates of MPCA, DATER, and GTDA start to decrease when more than 100 features are used in the FR experiments on the PIE database with $L = 2$ and $L = 5$. The investigation of the rather dramatic performance variations for some algorithms observed when the number of features used changes will be interesting but it is beyond the scope of this paper.

Finally, in order to provide some insights into the R-UMLDA algorithm, Fig. 11(a)–(c) depicts, as gray-level images, the first

TABLE VI
FACE RECOGNITION RESULTS ON THE FERET DATABASE: THE TOP CRRS (MEAN ± STD%)

C	Bayes	LDA	ULDA	R-JD-LDA	MPCA	DATER	GTDA	TR1DA	R-UMLDA	R-UMLDA-A
80	60.5±1.7	60.8±2.0	58.9±1.5	68.1±2.0	54.7±3.4	**71.0**±1.6	61.8±2.3	**71.0**±1.6	62.1±1.4	**75.1**±1.8
160	59.7±1.7	51.0±1.8	40.2±1.3	**70.5**±1.6	50.7±3.4	68.8±1.6	58.6±2.3	68.3±2.0	60.3±1.6	**75.1**±1.7
240	54.2±1.1	41.8±1.4	10.8±0.8	**68.7**±1.0	51.2±2.6	64.0±1.3	54.5±2.0	62.7±1.3	58.6±1.3	**73.2**±1.0
320	52.4±1.0	29.1±1.1	20.9±0.9	**66.4**±1.1	48.9±1.6	62.0±1.6	52.6±1.8	60.4±1.6	56.9±1.4	**72.5**±1.6

TABLE VII
GAIT RECOGNITION RESULTS ON THE USF GAIT CHALLENGE DATA SETS: THE TOP CRRS (IN PERCENT) FOR INDIVIDUAL GAIT SAMPLES

Probe	Bayes	LDA	ULDA	R-JD-LDA	MPCA	DATER	GTDA	TR1DA	R-UMLDA	R-UMLDA-A
A	46.2	65.7	60.8	**71.0**	54.7	61.2	58.7	63.5	51.9	**69.8**±1.3
B	44.2	49.2	43.3	**55.8**	50.4	51.8	54.1	50.8	45.9	**59.4**±1.0
C	26.0	31.7	30.0	**40.0**	34.3	33.6	**38.8**	35.7	25.2	36.7±1.0
Average	39.5	51.8	47.6	**58.3**	46.9	49.7	51.7	52.5	43.0	**57.9**±0.8

TABLE VIII
GAIT RECOGNITION RESULTS ON THE USF GAIT CHALLENGE DATA SETS: THE TOP CRRS (IN PERCENT) FOR GAIT SEQUENCES

Probe	Bayes	LDA	ULDA	R-JD-LDA	MPCA	DATER	GTDA	TR1DA	R-UMLDA	R-UMLDA-A
A	70.4	87.3	85.9	**88.7**	84.5	87.3	85.9	83.1	85.9	**95.6**±1.5
B	75.6	63.4	61.0	68.3	**80.5**	65.9	**78.0**	73.2	68.3	77.1±1.8
C	43.9	51.2	48.8	56.1	**61.0**	58.5	**63.4**	**61.0**	43.9	58.0±1.9
Average	62.8	70.6	66.6	73.2	75.2	72.5	**76.5**	74.5	69.9	**80.3**±1.0

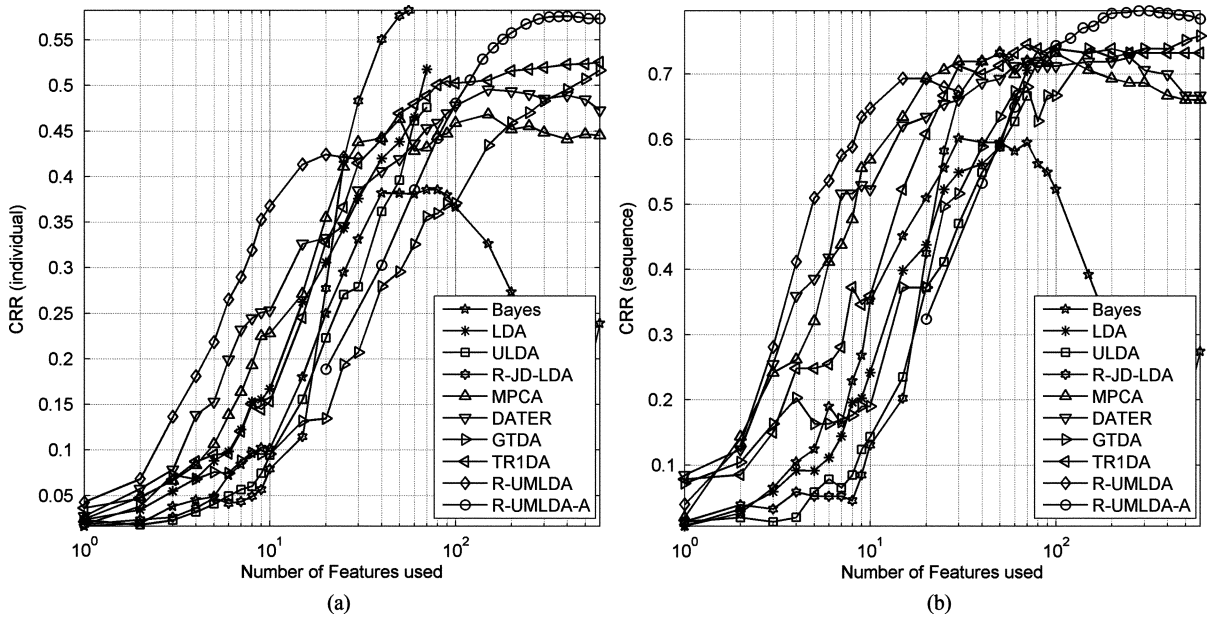

Fig. 10. Gait recognition results on the USF gait challenge data sets (average over probes A, B, and C): correct recognition rate against the number of features used for (a) individual gait samples, and (b) gait sequences.

three EMPs obtained by R-UMLDA using the PIE database with $L = 20$, the FERET database with $C = 320$, and the USF gait gallery set, respectively. From the EMPs for the face data, it can be seen that there is strong presence of structured information due to the multilinear nature, which is different from the information conveyed by the bases produced by linear algorithms such as eigenface [58] or fisherface [32]. It is interesting to note that the eye area is more evident in the second and third EMPs in Fig. 11(b). Fig. 11(a) and (b) indicates that the EMPs obtained by R-UMLDA encode discriminative information found in a large portion of the face, with each EMP capturing a particular pattern. In the EMPs for the gait data, which are third-order tensors displayed as their 1-mode unfolded matrices in Fig. 11(c), structure is again observed across the three modes (column, row, and time). The first gait EMP indicates that the most discriminative information encoded by R-UMLDA comes from the foot

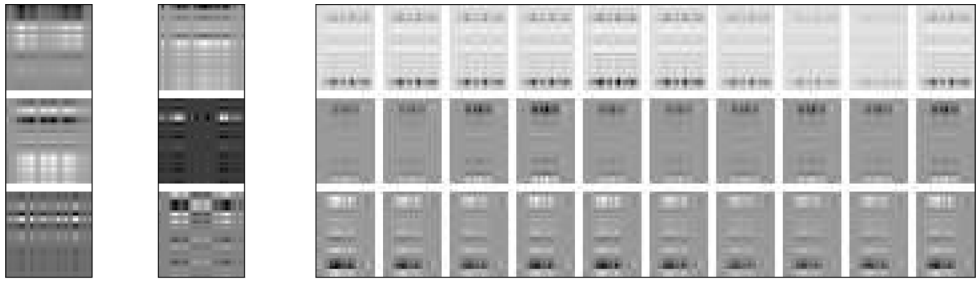

Fig. 11. Illustration of the first three EMPs (in top-to-down order) obtained by R-UMLDA from (a) the PIE database with $L = 20$, (b) the FERET database with $C = 320$, and (c) the USF gait gallery sequences (1-mode unfolding is shown).

area (bottom), which is expected based on the physiology of the human walking cycle. The second gait EMP demonstrates that in addition to the foot area, the head area (top) also provides important discriminative information. The third gait EMP encodes discriminative information from other parts of the body such as the lower legs and the hip. These observations provide insights into the nature of the features encoded by R-UMLDA and offer a better understanding of this algorithm's performance when applied to certain data sets.

VI. Conclusion

In this paper, a novel UMLDA algorithm is proposed to extract uncorrelated discriminative features directly from tensorial data using the TVP of tensor objects, and a regularization term is incorporated to tackle the SSS problem, resulting in the regularized UMLDA. Furthermore, since the recognition performance is affected by initialization and regularization, an aggregation scheme is proposed to combine several differently initialized and regularized UMLDA feature extractors to achieve better recognition performance while mitigating the problem of regularization parameter selection. Experiments on biometric applications, tested on the PIE and FERET face databases and the USF gait database, demonstrate that the UMLDA-based recognizer achieves the best overall results compared with recognizers based on other multilinear subspace solutions as well as linear subspace algorithms.

Appendix I
Proof of Theorem 1

Proof: For a nonsingular $\tilde{\mathbf{S}}_{W_p}^{(n^*)}$, any $\mathbf{u}_p^{(n^*)}$ can be normalized such that $\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} = 1$ and the ratio $\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}/\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)}$ keeps unchanged. Therefore, the maximization of this ratio is equivalent to the maximization of $\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}$ with the constraint that $\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} = 1$ and Lagrange multipliers can be used to transform the problem (11) to the following to include all the constraints:

$$F(\mathbf{u}_p^{(n^*)}) = \mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \nu\left(\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - 1\right) - \sum_{q=1}^{p-1}\mu_q\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q \quad (21)$$

where ν and $\{\mu_q, q = 1, \ldots, p-1\}$ are Lagrange multipliers.

The optimization is performed by setting the partial derivative of $F(\mathbf{u}_p^{(n^*)})$ with respect to $\mathbf{u}_p^{(n^*)}$ to zero

$$\frac{\partial F(\mathbf{u}_p^{(n^*)})}{\partial \mathbf{u}_p^{(n^*)}} = 2\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - 2\nu\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \sum_{q=1}^{p-1}\mu_q\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q = 0. \quad (22)$$

Multiplying (22) by $\mathbf{u}_p^{(n^*)^T}$ results in

$$2\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - 2\nu\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} = 0 \Rightarrow \nu = \frac{\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}}{\mathbf{u}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)}} \quad (23)$$

which indicates that ν is exactly the criterion to be maximized.

Next, a set of $(p-1)$ equations are obtained by multiplying (22) by $\mathbf{g}_q^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}$, $q = 1, \ldots, p-1$, respectively

$$2\mathbf{g}_q^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \sum_{q=1}^{p-1}\mu_q\mathbf{g}_q^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\cdot\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q = 0. \quad (24)$$

Let

$$\boldsymbol{\mu}_{p-1} = [\mu_1\ \mu_2\ \cdots\ \mu_{p-1}]^T \quad (25)$$

and use (14), then the $(p-1)$ equations of (24) can be represented in a single matrix equation as following:

$$2\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\cdot\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\boldsymbol{\mu}_{p-1} = 0. \quad (26)$$

Thus

$$\boldsymbol{\mu}_{p-1} = 2\left(\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\right)^{-1}\cdot\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}. \quad (27)$$

Since from (14) and (25)

$$\sum_{q=1}^{p-1}\mu_q\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{g}_q = \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\boldsymbol{\mu}_{p-1} \quad (28)$$

$$2\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - 2\nu\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\boldsymbol{\mu}_{p-1} = 0 \Rightarrow \nu\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}_p^{(n^*)} = \tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\frac{\boldsymbol{\mu}_{p-1}}{2}$$

$$= \tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)} - \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\left(\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\right)^{-1}\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}$$

$$= \left[\mathbf{I}_{I_{n^*}} - \tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\left(\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\tilde{\mathbf{Y}}_p^{(n^*)}\mathbf{G}_{p-1}\right)^{-1}\mathbf{G}_{p-1}^T\tilde{\mathbf{Y}}_p^{(n^*)^T}\tilde{\mathbf{S}}_{W_p}^{(n^*)^{-1}}\right]\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u}_p^{(n^*)}.$$

(22) can be written as shown in the equation at the top of the page. Using the definition in (13), a generalized eigenvalue problem is obtained as $\mathbf{R}_p^{(n^*)}\tilde{\mathbf{S}}_{B_p}^{(n^*)}\mathbf{u} = \nu\tilde{\mathbf{S}}_{W_p}^{(n^*)}\mathbf{u}$. Since ν is the criterion to be maximized, the maximization is achieved by setting $\mathbf{u}_p^{(n)^*}$ to be the (unit) generalized eigenvector corresponding to the largest generalized eigenvalue of (12). ■

Appendix II
Connections Among Various MLDA Algorithms and LDA

The traditional linear projection is a vector-to-vector projection (VVP). The two multilinear projections are the TTP and the TVP. Thus, LDA is based on VVP. DATER and GTDA are MLDA through TTP (MLDA-TTP), and TR1DA and UMLDA are MLDA through TVP (MLDA-TVP). The DATER algorithm [2] is a specific realization of the MLDA-TTP, with the objective of maximizing the scatter ratio. The GTDA algorithm [15] is an MLDA-TTP variant maximizing the scatter difference, The TR1DA algorithm [18], [19] is an MLDA-TVP variant maximizing the scatter difference, with a heuristic residue-based approach. The UMLDA algorithm proposed in this paper is an MLDA-TVP variant maximizing the scatter ratio, while pursuing uncorrelated features.

The relationships between the LDA, MLDA-TTP, and MLDA-TVP are revealed by examining the relationships between VVP, TTP, and TVP first. Obviously, VVP is the special case of TTP and TVP with $N = 1$. Each projected element in TTP can be viewed as the projection of an EMP formed by taking one column from each projection matrix so the projected tensor by TTP is obtained effectively through $\prod_{n=1}^{N} I_n$ interdependent EMPs, while in TVP, the P EMPs obtained successively are not interdependent in general. Moreover, the projection using an EMP $\{\mathbf{u}^{(1)^T}, \mathbf{u}^{(2)^T}, \ldots, \mathbf{u}^{(N)^T}\}$ can be written as $y = \langle \mathcal{X}, \mathcal{U} \rangle = \langle \mathrm{vec}(\mathcal{X}), \mathrm{vec}(\mathcal{U}) \rangle = [\mathrm{vec}(\mathcal{U})]^T \mathrm{vec}(\mathcal{X})$. Therefore, an EMP is equivalent to a linear projection of $\mathrm{vec}(\mathcal{X})$ on a vector $\mathrm{vec}(\mathcal{U})$. Since $\mathcal{U} = \mathbf{u}^{(1)} \circ \mathbf{u}^{(2)} \circ \cdots \circ \mathbf{u}^{(N)}$, the EMP is in effect a linear projection with constraint on the projection vector such that it is the vectorized representation of a rank-one tensor. Compared with a projection vector of size $I \times 1$ in VVP specified by I parameters ($I = \prod_{n=1}^{N} I_n$ for an Nth-order tensor), an EMP in TVP can be specified by $\sum_{n=1}^{N} I_n$ parameters. Hence, to project a tensor of size $\prod_{n=1}^{N} I_n$ to a vector of size $P \times 1$, the TVP needs to estimate only $P \cdot \sum_{n=1}^{N} I_n$ parameters, while the VVP needs to estimate $P \cdot \prod_{n=1}^{N} I_n$ parameters. The implication in pattern recognition problem is that the TVP has fewer parameters to estimate while being more constrained on the solutions, and the VVP has less constraint on the solutions sought while having more parameters to estimate. Summing up, LDA is a special case of MLDA-TTP and MLDA-TVP when $N = 1$, with the scatter ratio as the separation criterion. On the other hand, the MLDA-TTP is looking for interdependent EMPs while the EMPs sought successively in the MLDA-TVP are not interdependent generally. Furthermore, for the same projected vector size, the MLDA-TVP has fewer parameters to estimate while the projection to be solved is more constrained, and LDA has more parameters to estimate while the projection is less constrained.

Acknowledgment

The authors would like to thank the anonymous reviewers for their insightful and constructive comments. They would also like to thank Dr. J. Wang for her help in the experimental design, and the providers of the PIE, FERET, and USF gait databases.

References

[1] L. D. Lathauwer, B. D. Moor, and J. Vandewalle, "On the best rank-1 and rank-($R_1, R_2, \ldots, R_N$) approximation of higher-order tensors," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1324–1342, 2000.

[2] S. Yan, D. Xu, Q. Yang, L. Zhang, X. Tang, and H. Zhang, "Multilinear discriminant analysis for face recognition," *IEEE Trans. Image Process.*, vol. 16, no. 1, pp. 212–220, Jan. 2007.

[3] Z. Lei, R. Chu, R. He, S. Liao, and S. Z. Li, "Face recognition by discriminant analysis with Gabor tensor representation," in *Proc. Int. Conf. Biometrics*, Aug. 2007, pp. 87–95.

[4] Y.-D. Kim and S. Choi, "Color face tensor factorization and slicing for illumination-robust recognition," in *Proc. Int. Conf. Biometrics*, Aug. 2007, pp. 19–28.

[5] C. M. Cyr and B. B. Kimia, "3D object recognition using shape similarity-based aspect graph," in *Proc. IEEE Conf. Comput. Vis.*, Jul. 2001, vol. 1, pp. 254–261.

[6] K. W. Bowyer, K. Chang, and P. Flynn, "A survey of approaches and challenges in 3D and multi-modal 3D + 2D face recognition," *Comput. Vis. Image Understand.*, vol. 101, no. 1, pp. 1–15, Jan. 2006.

[7] S. Z. Li, C. Zhao, M. Ao, and Z. Lei, "Learning to fuse 3D + 2D based face recognition at both feature and decision levels," in *Proc. IEEE Int. Workshop Anal. Model. Faces Gestures*, Oct. 2005, pp. 43–53.

[8] C. Liu and H. Wechsler, "Independent component analysis of Gabor features for face recognition," *IEEE Trans. Neural Netw.*, vol. 14, no. 4, pp. 919–928, Jul. 2003.

[9] R. Chellappa, A. Roy-Chowdhury, and S. Zhou, *Recognition of Humans and Their Activities Using Video*. San Rafael, CA: Morgan & Claypool, 2005.

[10] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "MPCA: Multilinear principal component analysis of tensor objects," *IEEE Trans. Neural Netw.*, vol. 19, no. 1, pp. 18–39, Jan. 2008.

[11] G. Shakhnarovich and B. Moghaddam, "Face recognition in subspaces," in *Handbook of Face Recognition*, S. Z. Li and A. K. Jain, Eds. New York: Springer-Verlag, 2004, pp. 141–168.

[12] J. Zhang, S. Z. Li, and J. Wang, "Manifold learning and applications in recognition," in *Intelligent Multimedia Processing With Soft Computing*, Y. P. Tan, K. H. Yap, and L. Wang, Eds. New York: Springer-Verlag, 2004, pp. 281–300.

[13] M. H. C. Law and A. K. Jain, "Incremental nonlinear dimensionality reduction by manifold learning," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 3, pp. 377–391, Mar. 2006.

[14] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Gait recognition through MPCA plus LDA," in *Proc. Biometrics Symp.*, Sep. 2006, pp. 1–6.
[15] D. Tao, X. Li, X. Wu, and S. J. 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.
[16] J. Ye, R. Janardan, and Q. Li, "Two-dimensional linear discriminant analysis," in *Advances in Neural Information Processing Systems (NIPS)*. Cambridge, MA: MIT Press, 2004, pp. 1569–1576.
[17] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "A taxonomy of emerging multilinear discriminant analysis solutions for biometric signal recognition," in *Biometrics: Theory, Methods, and Applications*, N. V. Boulgouris, K. Plataniotis, and E. Micheli-Tzanakou, Eds. New York: Wiley/IEEE, 2008.
[18] Y. Wang and S. Gong, "Tensor discriminant analysis for view-based object recognition," in *Proc. Int. Conf. Pattern Recognit.*, Aug. 2006, vol. 3, pp. 33–36.
[19] D. Tao, X. Li, X. Wu, and S. J. Maybank, "Elapsed time in human gait recognition: A new approach," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Apr. 2006, vol. 2, pp. 177–180.
[20] T. G. Kolda, "Orthogonal tensor decompositions," *SIAM J. Matrix Anal. Appl.*, vol. 23, no. 1, pp. 243–255, 2001.
[21] X. He, D. Cai, and P. Niyogi, "Tensor subspace analysis," in *Advances in Neural Information Processing Systems 18 (NIPS)*. Cambridge, MA: MIT Press, 2005.
[22] G. Dai and D. Y. Yeung, "Tensor embedding methods," in *Proc. 21st Nat. Conf. Artif. Intell.*, Jul. 2006, pp. 330–335.
[23] S. Yan, D. Xu, B. Zhang, H. J. Zhang, Q. Yang, and S. Lin, "Graph embedding and extensions: A general framework for dimensionality reduction," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 1, pp. 40–51, Jan. 2007.
[24] D. Xu, S. Lin, S. Yan, and X. Tang, "Rank-one projections with adaptive margins for face recognition," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 37, no. 5, pp. 1226–1236, Oct. 2007.
[25] G. Hua, P. A. Viola, and S. M. Drucker, "Face recognition using discriminatively trained orthogonal rank one tensor projections," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, Jun. 2007, pp. 1–8.
[26] Z. Jin, J. Y. Yang, Z. M. Tang, and Z. S. Hu, "A theorem on the uncorrelated optimal discriminant vectors," *Pattern Recognit.*, vol. 34, no. 10, pp. 2041–2047, Oct. 2001.
[27] J. Ye, R. Janardan, Q. Li, and H. Park, "Feature reduction via generalized uncorrelated linear discriminant analysis," *IEEE Trans. Knowl. Data Eng.*, vol. 18, no. 10, pp. 1312–1322, Oct. 2006.
[28] Z. Jin, J. Y. Yang, Z. S. Hu, and Z. Lou, "Face recognition based on the uncorrelated discriminant transformation," *Pattern Recognit.*, vol. 34, pp. 1405–1416, 2001.
[29] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Uncorrelated multilinear principal component analysis through successive variance maximization," in *Proc. Int. Conf. Mach. Learn.*, Jul. 2008, pp. 616–623.
[30] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Regularization studies of linear discriminant analysis in small sample size scenarios with application to face recognition," *Pattern Recognit. Lett.*, vol. 26, no. 2, pp. 181–191, 2005.
[31] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Uncorrelated multilinear discriminant analysis with regularization for gait recognition," in *Proc. Biometrics Symp.*, Sep. 2007, pp. 1–6.
[32] P. N. Belhumeur, J. P. Hespanha, and D. J. Kriegman, "Eigenfaces vs. fisherfaces: Recognition using class specific linear projection," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 19, no. 7, pp. 711–720, Jul. 1997.
[33] J. Ye, "Characterization of a family of algorithms for generalized discriminant analysis on undersampled problems," *J. Mach. Learn. Res.*, vol. 6, pp. 483–502, Apr. 2005.
[34] T. K. Moon and W. C. Stirling, *Mathematical Methods and Algorithms for Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 2000.
[35] J. L. Rodgers, W. A. Nicewander, and L. Toothaker, "Linearly independent, orthogonal and uncorrelated variables," *Amer. Statist.*, vol. 38, no. 2, pp. 133–134, May 1984.
[36] Y. Koren and L. Carmel, "Robust linear dimensionality reduction," *IEEE Trans. Vis. Comput. Graphics*, vol. 10, no. 4, pp. 459–470, Jul. 2004.
[37] J. D. Carroll and J. J. Chang, "Analysis of individual differences in multidimensional scaling via an n-way generalization of "Eckart-Young" decomposition," *Psychometrika*, vol. 35, pp. 283–319, 1970.
[38] R. A. Harshman, "Foundations of the Parafac procedure: Models and conditions for an "explanatory" multi-modal factor analysis," *UCLA Working Papers in Phonetics*, vol. 16, pp. 1–84, 1970.
[39] P. Kroonenberg and J. Leeuw, "Principal component analysis of three-mode data by means of alternating least squares algorithms," *Psychometrika*, vol. 45, no. 1, pp. 69–97, 1980.
[40] J. H. Friedman, "Regularized discriminant analysis," *J. Amer. Statist. Assoc.*, vol. 84, no. 405, pp. 165–175, Mar. 1989.
[41] C. Liu, "Capitalize on dimensionality increasing techniques for improving face recognition grand challenge performance," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 5, pp. 725–737, May 2006.
[42] M. A. O. Vasilescu and D. Terzopoulos, "Multilinear analysis of image ensembles: Tensorfaces," in *Proc. 7th Eur. Conf. Comput. Vis.*, May 2002, pp. 447–460.
[43] S. W. Park and M. Savvides, "Individual kernel tensor-subspaces for robust face recognition: A computationally efficient tensor framework without requiring mode factorization," *IEEE Trans. Syst. Man Cybern. B, Cybern.*, vol. 37, no. 5, pp. 1156–1166, Oct. 2007.
[44] K. Jia and S. Gong, "Multi-modal tensor face for simultaneous super-resolution and recognition," in *Proc. Int. Conf. Comput. Vis.*, Oct. 2005, vol. 2, pp. 1683–1690.
[45] T. K. Ho, "The random subspace method for constructing decision forests," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 8, pp. 832–844, Aug. 1998.
[46] R. E. Schapire, "The boosting approach to machine learning: An overview," in *Nonlinear Estimation and Classification*, D. D. Denison, M. H. Hansen, C. Holmes, B. Mallick, and B. Yu, Eds. New York: Springer, 2003.
[47] R. P. W. D. M. Skurichina, "Bagging, boosting and the random subspace method for linear classifiers," *Pattern Anal. Appl.*, vol. 5, no. 2, pp. 121–135, 2002.
[48] L. Breiman, "Arcing classifiers," *Ann. Statist.*, vol. 26, no. 3, pp. 801–849, 1998.
[49] A. Ross and R. Govindarajan, "Feature level fusion of hand and face biometrics," in *Proc. SPIE Conf. Biometric Technol. Human Identification II*, Mar. 2005, pp. 196–204.
[50] A. K. J. A. Ross and J. Z. Qian, "Information fusion in biometrics," *Pattern Recognit. Lett.*, vol. 24, pp. 2115–2125, 2003.
[51] J. Kittler, M. Hatef, R. P. W. Duin, and J. Matas, "On combining classifiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 20, no. 3, pp. 226–239, Mar. 1998.
[52] J. Lu, K. N. Plataniotis, A. N. Venetsanopoulos, and S. Z. Li, "Ensemble-based discriminant learning with boosting for face recognition," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 166–178, Jan. 2006.
[53] H. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Boosting LDA with regularization on MPCA features for gait recognition," in *Proc. Biometrics Symp.*, Sep. 2007, pp. 1–6.
[54] T. Sim, S. Baker, and M. Bsat, "The CMU pose, illumination, and expression database," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no. 12, pp. 1615–1618, Dec. 2003.
[55] P. J. Phillips, H. Moon, S. A. Rizvi, and P. Rauss, "The FERET evaluation method for face recognition algorithms," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 10, pp. 1090–1104, Oct. 2000.
[56] B. Moghaddam, T. Jebara, and A. Pentland, "Bayesian face recognition," *Pattern Recognit.*, vol. 33, no. 11, pp. 1771–1782, 2000.
[57] J. Lu, K. N. Plataniotis, and A. N. Venetsanopoulos, "Face recognition using LDA based algorithms," *IEEE Trans. Neural Netw.*, vol. 14, no. 1, pp. 195–200, Jan. 2003.
[58] M. Turk and A. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

Haiping Lu (S'02) received the B.Eng. and M.Eng degrees in electrical and electronic engineering from Nanyang Technological University, Singapore, in 2001 and 2004, respectively, and the Ph.D. degree in electrical and computer engineering from University of Toronto, Toronto, ON, Canada, in 2008.

Currently, he is a Postdoctoral Fellow in the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto. His current research interests include statistical pattern recognition, machine learning, multilinear algebra, tensor object processing, biometric encryption, and data mining.

Konstantinos N. (Kostas) Plataniotis (S'90–M'92–SM'03) received the B. Eng. degree in computer engineering from University of Patras, Patras, Greece, in 1988 and the M.S. and Ph.D. degrees in electrical engineering from Florida Institute of Technology, Melbourne, FL, in 1992 and 1994, respectively.

Currently, he is a Professor at the Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, and an Adjunct Professor at the School of Computer Science, Ryerson University, Toronto, ON, Canada. He is a founding member and the inaugural Associate Director-Research of the Identity, Privacy and Security Initiative (IPSI) at the University of Toronto. He servers on the Executive Committee of Knowledge Media Design Institute (KMDI) at the University of Toronto. He is a contributor to ten books, coauthor of *Color Image Processing and Applications* (New York: Springer Verlag, 2000, ISBN-3–540-66953–1), coeditor of *Color Imaging: Methods and Applications* (Boca Raton, FL: CRC Press, 2006, ISBN 084939774X).

Dr. Plataniotis was the Guest Editor for the March 2005 IEEE SIGNAL PROCESSING MAGAZINE special issue on Surveillance Networks and Services. He is an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS and the IEEE SIGNAL PROCESSING LETTERS. He is a member of the 2008 IEEE Educational Activity Board and the Chair of the IEEE EAB Continuing Professional Education Committee. He is the 2008 representative of the Computational Intelligence Society to the IEEE Biometrics Council and a member of the Steering Committee for the IEEE TRANSACTIONS ON MOBILE COMPUTING. He will serve as the Editor-in-Chief for the IEEE SIGNAL PROCESSING LETTERS from January 1, 2009 to December 31, 2011. He is a registered professional engineer in the province of Ontario, and a member of the Technical Chamber of Greece. In 2005, he became the recipient of the IEEE Canada Engineering Educator Award "for contributions to engineering education and inspirational guidance of graduate students." He is the joint recipient of the "2006 IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award" for the paper "Face recognition using kernel direct discriminant analysis algorithms" published in 2003. He is the Managing Guest Editor of the IEEE/Wiley Press volume on *Biometrics: Theory, Method and Applications* to be published in 2009.

Anastasios N. Venetsanopoulos (S'66–M'69–SM'79–F'88–LF'07) received the B.S. degree in electrical and mechanical engineering from National Technical University of Athens (NTU), Athens, Greece, in 1965 and the M.S., M.Phil., and Ph.D. degrees in electrical engineering from Yale University, New Haven, CT, in 1966, 1968, and 1969, respectively.

He joined the Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, in September 1968 as a Lecturer and he was promoted to Assistant Professor in 1970, Associate Professor in 1973, and Professor in 1981. He has served as Chair of the Communications Group and Associate Chair of the Department of Electrical Engineering. Between July 1997 and June 2001, he was Associate Chair (graduate studies) in the Department of Electrical and Computer Engineering and was acting Chair during the spring term of 1998–1999. He served as the Inaugural Bell Canada Chair in Multimedia between 1999 and 2005. Between 2001 and 2006, he served as the 12th Dean of the Faculty of Applied Science and Engineering, University of Toronto and on October 1, 2006, he accepted the position of founding Vice-President research and innovation at Ryerson University. He was on research leave at the Imperial College of Science and Technology, the National Technical University of Athens, the Swiss Federal Institute of Technology, the University of Florence, and the Federal University of Rio de Janeiro, and has also served as Adjunct Professor at Concordia University. He has served as lecturer in 138 short courses to industry and continuing education programs and as consultant to numerous organizations; he is a contributor to 35 books and has published over 800 papers in refereed journals and conference proceedings on digital signal and image processing and digital communications.

Prof. Venetsanopoulos was elected IEEE Fellow "for contributions to digital signal and image processing"; he is also Fellow of the Engineering Institute of Canada (EIC) and was awarded an honorary doctorate from the National Technical University of Athens in October 1994. In October 1996, he was awarded the "Excellence in Innovation Award" of the Information Technology Research Centre of Ontario and Royal Bank of Canada, "for innovative work in color image processing and its industrial applications." In November 2003, he was the recipient of the "Millennium Medal of IEEE." In April 2001, he became Fellow of the Canadian Academy of Engineering. In 2003, he was the recipient of the highest award of the Canadian IEEE, the MacNaughton Award, and in 2003 he served as the Chair of the Council of Deans of Engineering of the Province of Ontario (CODE). In 2006, he was selected as the joint recipient of the 2003 IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award.